

# Digital Quantum Simulation of Cavity Quantum Electrodynamics: Insights from Superconducting and Trapped Ion Quantum Testbeds

Alex H. Rubin,<sup>1,2</sup> Brian Marinelli,<sup>3,4</sup> Victoria A. Norman,<sup>1,2</sup> Zainab Rizvi,<sup>5</sup> Ashlyn D. Burch,<sup>6</sup>

Ravi K. Naik,<sup>3,4</sup> John Mark Kreikebaum,<sup>3,7</sup> Matthew N. H. Chow,<sup>6</sup> Daniel S. Lobser,<sup>6</sup>

Melissa C. Revelle,<sup>6</sup> Christopher G. Yale,<sup>6</sup> Megan Ivory,<sup>6</sup> David I. Santiago,<sup>3,4</sup> Christopher Spitzer,<sup>3,4</sup> Marina Krstic-Marinkovic,<sup>8</sup> Susan M. Clark,<sup>6</sup> Irfan Siddiqi,<sup>3,4</sup> and Marina Radulaski<sup>1</sup>

<sup>1</sup>Department of Electrical and Computer Engineering, University of California, Davis, CA 95616

<sup>2</sup>Department of Physics, University of California, Davis, CA 95616, USA

<sup>3</sup>Quantum Nanoelectronics Laboratory, Department of Physics, University of California at Berkeley, Berkeley, CA 94720, USA

<sup>4</sup>Applied Mathematics and Computational Research Division, Lawrence

Berkeley National Laboratory, Berkeley, California 94720, USA

<sup>5</sup>Department of Computer Science, University of California, Davis, CA 95616

<sup>6</sup>Sandia National Laboratories, Albuquerque, New Mexico 87123, United States

<sup>7</sup>Materials Science Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720, USA

<sup>8</sup>Institute for Theoretical Physics, ETH Zurich, 8093 Zurich, Switzerland

(Dated: December 24, 2024)

We explore the potential for hybrid development of quantum hardware where currently available quantum computers simulate open Cavity Quantum Electrodynamical (CQED) systems for applications in optical quantum communication, simulation and computing. Our simulations make use of a recent quantum algorithm that maps the dynamics of a singly excited open Tavis-Cummings model containing  $N$  atoms coupled to a lossy cavity. We report the results of executing this algorithm on two noisy intermediate-scale quantum computers: a superconducting processor and a trapped ion processor, to simulate the population dynamics of an open CQED system featuring  $N = 3$  atoms. By applying technology-specific transpilation and error mitigation techniques, we minimize the impact of gate errors, noise, and decoherence in each hardware platform, obtaining results which agree closely with the exact solution of the system. These results can be used as a recipe for efficient and platform-specific quantum simulation of cavity-emitter systems on contemporary and future quantum computers.

## I. INTRODUCTION

Since quantum computers were first conceived of in the 1980s, a major motivation for their construction has been the possibility of efficient simulation of quantum physics [1]. Indeed, efficient methods of implementing the unitary time evolution operator  $e^{-iHt}$  are known for broad classes of physically relevant Hamiltonians  $H$ , requiring a quantum circuit whose size is polynomial in the degrees of freedom of  $H$  [2–4]. Meanwhile, classical solutions of the same problem are superpolynomial at best [5]. It is hoped that this efficiency can be used to solve physical chemistry problems related to drug development and battery manufacturing, among other areas [6, 7].

Another field in which the classical intractability of large quantum simulations is problematic is the design of many-body cavity quantum electrodynamical devices. These are systems which feature multiple quantum emitters (e.g. quantum dots, color centers, or atoms) coupled to an optical cavity or a coupled cavity array, and can form the building blocks of technologies such as quantum repeaters, simulators, and computers [8, 9]. The use of optics provides natural advantages in certain areas of application; for example, the dominance of fiber optics in long-haul telecommunications makes CQED physics essential in designing repeater nodes for a quantum internet [10]. Crucially, these devices must be modeled as open quantum systems in order to capture their opti-

cal emission and absorption and to understand the impact of the various forms of decoherence on their performance. Therefore the well-studied techniques of Hamiltonian simulation are not sufficient, and we must instead apply methods which can account for the non-unitarity inherent in open quantum systems.

Fortunately, several methods for implementing efficient simulations of Lindbladian dynamics have been developed in recent years, with applicability to quite general systems with at most polynomial overhead [11–16]. Given the limited sizes and coherence times of today’s quantum hardware, even this modest overhead is still generally too much for practical application of these methods to technologically interesting CQED systems, a challenge that is being tackled by the development of model-specific algorithms and timely growth of quantum computational capabilities. However, the practical aspects of executing these algorithms on specific quantum platforms need to be investigated to maximize the near term utility.

In this work, we bridge the gap between algorithm requirements and hardware capabilities by exploring methods to optimize the simulation of an open CQED system on two leading quantum computing platforms: one based on superconducting qubits and the other using trapped ions. The algorithm at hand is a mapping of the singly excited Tavis-Cummings system of  $N$  two-level atoms coupled to a lossy cavity [17] onto a quantum cir-

cuit. By reducing the Hilbert space of each component in the system to two dimensions, this mapping reduces hardware requirements to circuit depths suitable for implementation on currently available quantum processing units (QPUs). Through the use of optimized compilation and error mitigation strategies (such as mirror SWAPs, randomized compiling, and noiseless output extrapolation), we raise the performance of the system so that accurate simulation results can be obtained.

## II. TAVIS-CUMMINGS ALGORITHM

The Tavis-Cummings model [18] describes  $N$  two-level quantum emitters coupled to a single optical mode (see Fig. 1(a)) with the Hamiltonian

$$H_{\text{TC}} = \omega_c a^\dagger a + \sum_{i=1}^N \left\{ \frac{1}{2} \omega_i \sigma_i^z + g_i (\sigma_i^+ a + a^\dagger \sigma_i^-) \right\}, \quad (1)$$

where  $\omega_c$ ,  $\omega_i$  are the frequencies of the cavity mode and the  $i^{\text{th}}$  emitter, respectively;  $\sigma_i^z$ ,  $\sigma_i^\pm$  are the Pauli  $Z$  and raising/lowering operators for the  $i^{\text{th}}$  emitter, respectively;  $a$  ( $a^\dagger$ ) is the annihilation (creation) operator for the cavity mode; and  $g_i$  is the coupling of the  $i^{\text{th}}$  emitter to the cavity mode. Here and throughout this discussion we use natural units ( $\hbar = 1$ ).

Including Markovian interactions with the environment in the form of optical loss from the cavity mode, time evolution of the density matrix  $\rho$  for the system obeys a master equation which takes the following Lindblad form

$$\dot{\rho} = \mathcal{L}(\rho) \equiv -i[H_{\text{TC}}, \rho] + \frac{\kappa}{2} \mathcal{D}_a(\rho), \quad (2)$$

where  $\mathcal{D}_a(\rho) = 2a\rho a^\dagger - \{a^\dagger a, \rho\}$  is the dissipator generated by  $a$ , representing loss from the cavity at rate  $\kappa$ .

Note that this master equation neglects blackbody excitation of the cavity, because the average thermal occupation of modes in the technologically relevant wavelength range ( $\sim 1 \mu\text{m}$ ) are extremely low at room temperature and below. For simplicity, we also neglect other decoherence processes such as nonradiative relaxation and phase decay of the emitters. As described in the foregoing section, numerically solving (2) for a particular value of time  $t$  as

$$\rho(t) = e^{\mathcal{L}t}(\rho(t=0)) \quad (3)$$

using classical methods incurs  $\mathcal{O}(2^N)$  memory and run-time costs in general. We therefore seek efficient quantum algorithms for this purpose.

The recently developed Q-MARINA algorithm maps this special case of the open Tavis-Cummings system onto a quantum circuit with depth  $2N + 1$  gates acting on  $N + 1$  qubits [17], achieving  $\mathcal{O}(N)$  space and time cost. It is important to note that this mapping works only in the single-excitation regime, which is efficiently

solvable by classical methods as well. Nevertheless, its low resource requirements make this method practical with current quantum processors, and therefore presents an opportunity for exploring the feasibility of achieving accurate simulations of CQED physics on today's noisy intermediate-scale quantum (NISQ) hardware. Because we consider only a single excitation, the combined population of the cavity/environment can be represented by a single qubit. By bringing the environment into the simulation explicitly, the dissipative dynamics of the system can be implemented with the unitary operations available on a quantum processor. The  $N$  two-level emitter populations are likewise mapped onto  $N$  qubits.

Starting with an initial state where one emitter is excited and the rest of the system is in the ground state, the simulation circuit calculates the expectation value of the populations of each emitter ( $\langle \sigma_i^\dagger \sigma_i \rangle$ ) and the cavity/environment at some arbitrary time  $t$  in the future. The population transfer between the cavity/environment qubit and each emitter qubit is computed with a controlled-Y rotation followed by a CNOT acting on those two qubits. The angle of the controlled-Y rotation is a function of the Tavis-Cummings system parameters ( $g$ ,  $\kappa$ ), the propagation time  $t$ , and the rotation angles of each cavity-emitter interaction calculated previously in the circuit:  $\theta_i = \theta_i(g, \kappa, t, \theta_1, \dots, \theta_{i-1})$ . The overall structure of a Q-MARINA circuit (see Fig. 1(b)) is as follows:

1. Prepare the initial state with an X gate on one emitter qubit, representing an excitation.
2. The excited emitter qubit interacts with the cavity/environment qubit through a controlled-Y rotation and a CNOT.
3. Each other emitter qubit interacts with the cavity/environment qubit (via a controlled-Y rotation and a CNOT) in turn.

The structure of this algorithm as a direct mapping between physical system and quantum circuit stands in contrast to other algorithms for simulating general open quantum systems ([11–16]) which rely on either a Trotter decomposition or the use of carefully chosen measurements to implement the dissipative dynamics. As a result, these methods have a tradeoff between the number of Trotter steps or measurements, and the final accuracy. Q-MARINA directly implements the exact dynamics by including the environment in the simulation, so its accuracy boils down to the errors inherent in the quantum processor it is implemented on.

We use the Q-MARINA algorithm to simulate an  $N = 3$  open Tavis-Cummings system on two NISQ devices: a trapped ion processor, and a superconducting processor. We benchmark the results from the two quantum processors against the numerical solution to the master equation (2) obtained using QuTiP (Quantum Toolbox in Python), a software package for classical simulations of quantum systems [19].

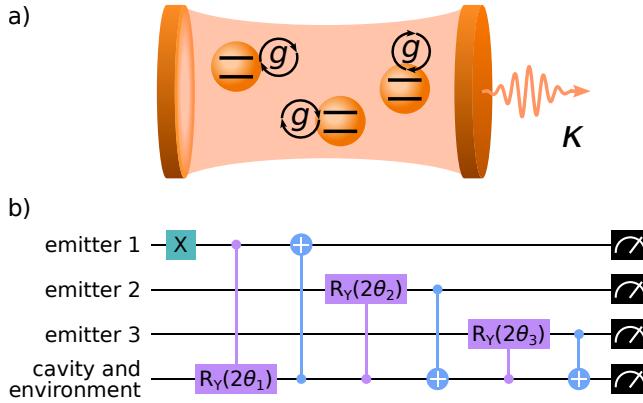


FIG. 1. (a) Diagram of an open Tavis-Cummings CQED system consisting of three two-level quantum emitters coupled to a lossy optical cavity mode. (b) The Q-MARINA circuit for an  $N = 3$  emitter Tavis-Cummings system used in the simulations presented in this work. The initial state is emitter 1 in the excited state, and all other parts of the system are in their ground states.

### III. QUANTUM ERROR MITIGATION

Since the 1990s, work by Shor and others has shown that it's possible in principle to detect and correct errors in a quantum processor provided the physical error rates remain below certain thresholds [20, 21]. Quantum computing hardware has made great strides in the decades since then, and recently some groups have demonstrated the first logical qubits operating below threshold [22, 23]. However these error thresholds remain quite challenging to meet, and systems featuring error-corrected qubits are still in their infancy.

In light of these challenges, researchers have developed a suite of techniques known as quantum error mitigation (QEM) that can partially compensate for errors without requiring the significant quantum resources needed for full error correction [24]. QEM techniques typically take advantage of features of a quantum processor's noise profile, implementation details of its gates, or structure implicit to a given quantum circuit, and use this to guide a circuit's compilation or postprocessing of measurement results in order to minimize the impact of noise. Therefore QEM is most effectively used by taking into account the details of the circuit and hardware platform. For example, randomized compiling (RC) acts to convert coherent errors into stochastic ones by compiling many different (but logically equivalent) versions of a circuit; platforms in which coherent errors dominate will thus be best suited to using RC.

#### A. Model-specific postselection and averaging

A particularly simple form of error mitigation we employed in our simulations is the post-processing applied

to the raw measurement results, which is guided by the physics of our simulations. First, we postselected our data by excluding any measurements which violated conservation of energy in the simulated system. As previously described, the initial state of the system represents a single excitation in one of the emitters; since the environment is included in the simulation, in the absence of noise the total population in the system should remain fixed. We therefore discard any measurement results which fall outside the single-excitation subspace, as they represent errors. Unlike other postselection schemes where a fixed fraction of events must be discarded due to inherent probabilistic effects, our method's discard rate depends only on the quality of the experimental implementation. This means that as hardware improves, the postselection overhead could approach zero, allowing the technique to scale to larger simulations. In our simulations the fraction of discarded shots ranged from 8-29% depending on the specific implementation.

As a second postprocessing step, we note that the simulated populations of emitters two and three should be the same at all time steps since they are treated identically in the master equation (2) and have the same initial state (the ground state). Any discrepancy between their populations represents the effect of noise, so after postselection we average these two emitters' populations and consider that the population of both. This is essentially equivalent to doubling the number of shots on this portion of the simulation, reducing sampling error by a factor of  $1/\sqrt{2}$ , or  $1/\sqrt{N-1}$  for systems with  $N$  emitters. All data presented in this work has been postselected and averaged in the manner just described unless otherwise stated.

### IV. IMPLEMENTATION ON A TRAPPED ION QUANTUM TESTBED

#### A. The QSCOUT trapped ion quantum processor

The first implementation of the algorithm took place on a trapped ion quantum processor, the Quantum Scientific Computing Open User Testbed (QSCOUT) at Sandia National Laboratories [25]. The platform consists of four ytterbium ( $^{171}\text{Yb}^+$ ) ion qubits in a linear trap. All-to-all connectivity is provided by coupling to the vibrational modes of the ion chain. This high connectivity lends itself well to the algorithm, which requires one-to-all interactions.

On the hardware, all Raman transitions are realized with combinations of individual addressing beams and a global beam counter-propagating to the individual addressing beams' direction. The single-qubit gates are realized by placing both tones of the Raman transition on an individual beam (co-propagating), while the two-qubit gates are realized by placing the necessary tones across both the individual and global beams (counter-propagating). We use QSCOUT's single qubit X gate and

two-qubit arbitrary-angle Mølmer–Sørensen gate, which can be expressed on the hardware in either the XX basis ( $MS_{XX}$ ), or a ZZ basis variation. While the bare MS interaction is XX-type, both versions require a series of wrapper single-qubit gates in order to mitigate phase instabilities when mixing gates of differing propagation [26] and to reduce phase-dependent crosstalk [27]. The fidelities of the MS and single-qubit gates are given in Appendix A. A high-level programming language, `Jaql`, is provided for interfacing with the hardware.

We executed the Q-MARINA algorithm on this platform to simulate a CQED system with the following parameters:  $N = 3$ ,  $g = 4$  GHz,  $\kappa = 2$  GHz; the system was evolved in time from  $t = 0$  ns to  $t_{\max} = 3$  ns. We benchmark our quantum simulations against the results of a classical simulation obtained with the python package QuTiP.

## B. Circuit compilations

We first manually compiled the algorithm to QSCOUT’s native gates by hand and implemented it in the programming language `Jaql`. Each controlled-R<sub>Y</sub> operation was compiled into a pair of CNOT gates and unconditional R<sub>Y</sub> rotations [28] and further decomposed each CNOT into  $MS_{XX}$  and single-qubit rotations [29], yielding the circuit seen in Fig. 3(a).

While preserving the  $\mathcal{O}(N)$  circuit depth, this manual compilation increased the overall number of two-qubit gates in each circuit from  $2N$  to  $3N$ , an overhead which would be expected to impact performance given that two-qubit gates are typically the most challenging operations in a quantum computer. The results of this hand-compiled version of the simulation can be seen in Fig. 2.

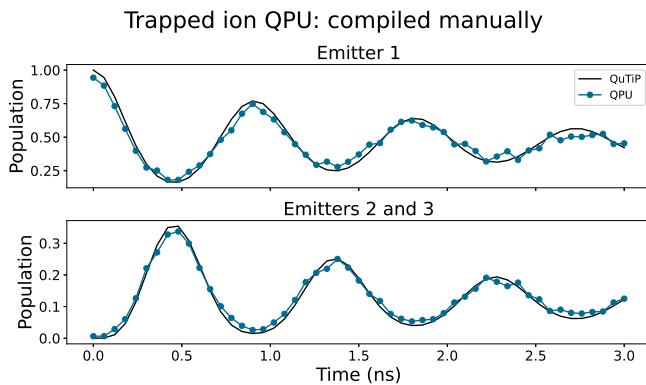


FIG. 2. Results from the QSCOUT trapped ion QPU using the manually compiled version of the algorithm. Each data point represents 2,000 shots; approximately 27% of shots were discarded in postselection using the technique described in Sec. III A. Wall-clock runtime of the simulation was about eight minutes. Black line indicates the exact solution of the system.

We improved upon the use of fixed MS entangling

gates in the hand-compiled version of our algorithm by switching to ZZ( $\theta$ ) entangling gates. This was done using `Superstaq`, an optimizing compiler for quantum programs [30, 31]. Starting with an implementation of our algorithm written in `qiskit` (see Fig. 1(b)) `Superstaq` compiled it to QSCOUT’s native gates, with ZZ( $\theta$ ) as the two-qubit entangling gate. This compilation method produced native circuits containing  $2N$  two-qubit gates (see Fig. 3(b)), matching the two-qubit gate complexity of the original abstract circuit (Fig. 1(b)). Results of executing this `Superstaq`-compiled version of Q-MARINA are presented in Fig. 4.

## C. Error mitigation: SWAP mirroring

To attempt to further improve the simulation accuracy, we applied an error mitigation technique known as SWAP mirroring [30], which is implemented as a feature within the `Superstaq` quantum compiler. SWAP mirroring is based on the observation that there are instances when appending two SWAP gates to an arbitrary two-qubit unitary can allow it to be compiled to a more efficient gate sequence than the original unitary itself, since the final SWAP can be achieved by virtual qubit relabeling. The `Superstaq` compiler compares versions of the circuit compiled with and without appended SWAPs, and selects the implementation which minimizes the total MS rotation angle. Smaller MS gate angles require lower laser amplitude to implement, and are thus subject to reduced amplitude noise. The total MS gate angle across the simulation was reduced by about 2%; the MS gates which were improved by SWAP mirroring had their rotation angles reduced by about 17% on average. The results of executing our algorithm compiled by `Superstaq` with ZZ( $\theta$ ) entangling gates and SWAP mirroring is seen in Fig. 5.

## D. Error mitigation: randomized compiling

We also applied error mitigation by using randomized compiling [32, 33]. This technique adds randomly selected single-qubit gates around each two-qubit gate cycle, chosen in such a way that the overall logical operation of the circuit is unchanged. Many equivalent randomized versions of the circuit are produced, each is run for some number of shots, and an average is taken over all the results. The effect of this randomization is to convert coherent and non-Markovian errors into uncorrelated stochastic errors; a larger number of randomizations more strongly tailors the noise. Since coherent errors accumulate quadratically in circuit depth and stochastic errors accumulate linearly, randomized compiling provides an advantage by mitigating the growth of errors. The results of the simulation compiled with RC into 10 randomizations are presented in Fig. 6.

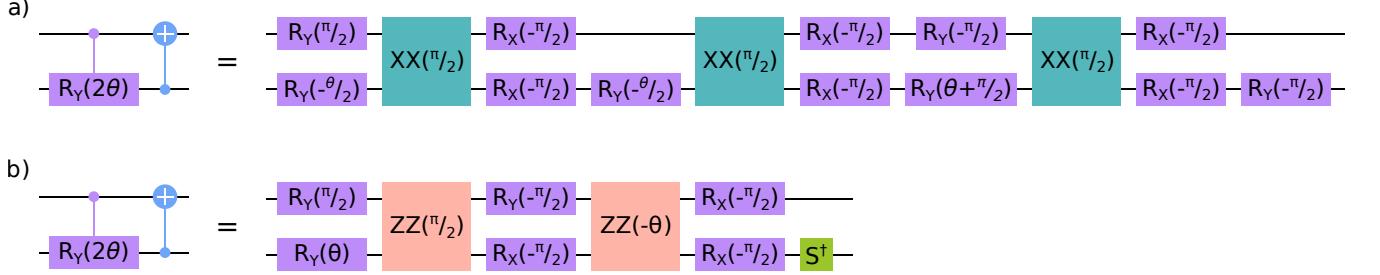


FIG. 3. Different compilations of the gates implementing an interaction between the emitter qubit and the environment qubit, using QSCOUT native gates. (a) The manually compiled version derived as described in the main text. (b) An example of a compilation using ZZ entangling gates, produced by *Superstaq*. Note that while the compiled circuit always contains a  $ZZ(\pi/2)$  and  $ZZ(-\theta)$ , the optimized single-qubit wrapper gates vary depending on the value of  $\theta$ , so this exact compilation was not used in every time step of the simulation, and is presented only as an example.

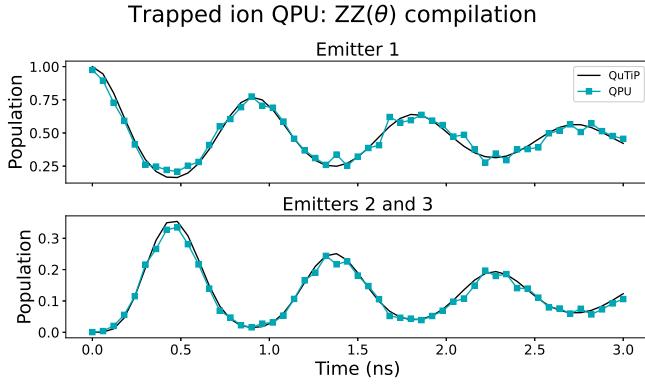


FIG. 4. Results from the QSCOUT trapped ion QPU using the version of the simulation circuit compiled with  $ZZ(\theta)$  entangling gates by *Superstaq*. Each data point represents a simulation with 2,000 shots; approximately 10% of shots were discarded in postselection using the technique described in Sec. III A. Wall-clock runtime of the simulation was about three minutes. Black line indicates the exact solution of the system.

### E. Analysis

The behaviors of the cavity and emitter populations over time are fundamental to characterizing a CQED system’s behavior. Accurately modeling these dynamics can not only shed light on the basic physics involved (for example, Rabi oscillations, decoherence, and in some cases, collective phenomena such as super- and subradiance) but is also of central importance in designing such a system for technological applications – for instance, optimizing quantum state transduction in a CQED-based quantum repeater node. To quantitatively assess the accuracy of our quantum simulations we employ the Hellinger distance, a metric of the difference between two probability distributions. At each time step, the simulation yields the populations of the constituent parts of the system, which sum to one, so the Hellinger distance between the simulated distribution and the exact distribution at time

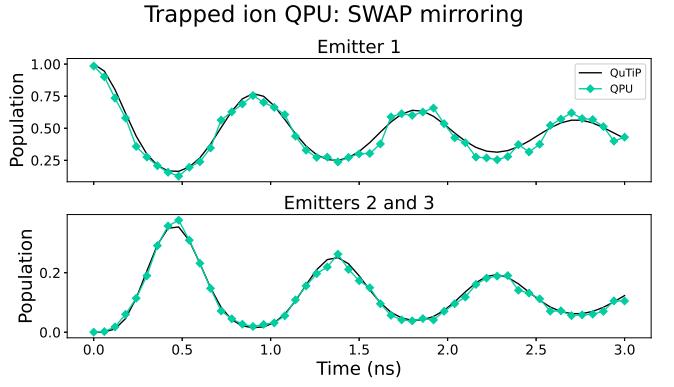


FIG. 5. Results from the QSCOUT trapped ion QPU using the algorithm compiled with  $ZZ(\theta)$  entangling gates and SWAP mirroring by *Superstaq*. Each data point represents a simulation with 2,000 shots; approximately 10% of shots were discarded in postselection using the technique described in Sec. III A. Wall-clock runtime of the simulation was about three minutes. Black line indicates the exact solution of the system.

$t$  is

$$H(t) = \frac{1}{\sqrt{2}} \sqrt{\sum_i \left( \sqrt{s_i(t)} - \sqrt{e_i(t)} \right)^2}. \quad (4)$$

where  $s_j[t]$  and  $e_j[t]$  are the simulated and exact probabilities for state  $j$  at time  $t$ . We then take the mean of  $H(t)$  across the full timespan of the simulation:

$$H = \frac{1}{t_{\max}} \sum_{t=0}^{t_{\max}} H(t), \quad (5)$$

which we refer to as Mean Hellinger Distance (MHD). This can be taken as a mean of the error of the entire simulation; a Mean Hellinger Distance  $H = 0$  would indicate perfect agreement between the results of the quantum processor and the exact numerical solution.

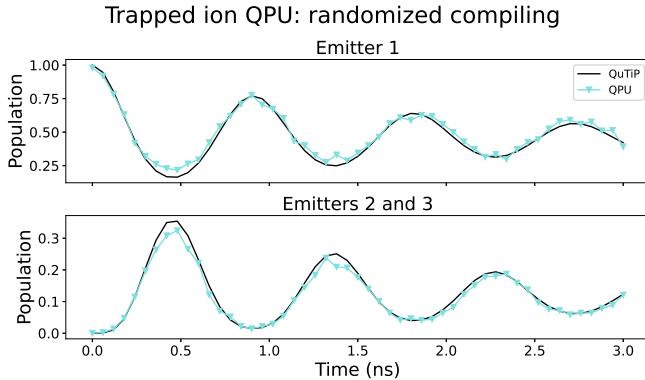


FIG. 6. Results from the QSCOUT trapped ion QPU using the algorithm compiled with  $ZZ(\theta)$  entangling gates, averaged over 10 randomized compilations. Each data point represents a simulation with 2,000 shots; approximately 15% of shots were discarded in postselection using the technique described in Sec. III A. Wall-clock runtime of the simulation was about one hour. The black line indicates the exact solution of the system.

In Fig. 7, we compare the MHD for the four versions of the simulation run on the QSCOUT trapped ion processor. When postselection is not applied, all three of the error mitigation techniques ( $ZZ(\theta)$  compilation, mirror SWAP, and RC) all provide a similar level of reduction in MHD relative to the manually compiled version of the simulation with fixed MS gates, which performs the worst. The randomized compiling version of the algorithm has a somewhat higher MHD than either the  $ZZ(\theta)$  or mirror SWAP versions, likely due to the deeper circuits produced by RC. While ion traps benefit from high gate fidelities and coherence times, their slower gate speeds mean deeper circuits take longer to execute, during which time experimental parameters like laser intensities and trap fields can drift, potentially introducing cumulative errors that are difficult to mitigate through randomization techniques.

The most significant improvement in MHD is provided by postselection itself, which reduces the simulation error by as much as a factor of 10. When postselection is applied, all four versions of the algorithm exhibit roughly comparable performance. This likely indicates that the errors affecting the simulation are predominantly stochastic, as the three error mitigation techniques act by either reducing coherent errors ( $ZZ(\theta)$  and mirror SWAP) or tailoring them into stochastic errors for better averaging (RC).

The effect of the error mitigation strategies can also be seen in the fraction of shots that were discarded by the postselection technique (Fig. 8). As the accuracy of the simulation improves, fewer shots of the simulation fall outside the single-excitation subspace and are excluded by postselection. These results broadly follow the trend of the non-postselected data in Fig. 7.

We also examined the accuracy of the simulations in

frequency space. A central feature of a strongly coupled CQED system's behavior is Rabi oscillations: exchange of excitations between the emitters and the cavity. Understanding the rate of Rabi oscillations is crucial for engineering control pulses needed to operate the system as a quantum networking device or component of a quantum computer. To assess the accuracy of the simulated Rabi oscillations, we compared the Fourier transforms of the simulation results with those of the numerical solution. These results for QSCOUT are presented in Fig. 9 for all three versions of the simulation. Since the simulated system consists of identical emitters resonant with the cavity, the theoretical Rabi rate is simply  $\Omega = \sqrt{N}g$ . We find that the quantum simulations accurately obtain the Rabi rate of the system, even without optimized compilation techniques or error mitigation, which is partly a reflection of the robustness of Fourier analysis to noise. Errors are largely uncorrelated between time steps in the simulation, so they appear as low-amplitude white noise in the frequency domain. This allows the fundamental Rabi frequency to remain clearly identifiable in the Fourier transform despite the presence of amplitude noise in the time-domain data.

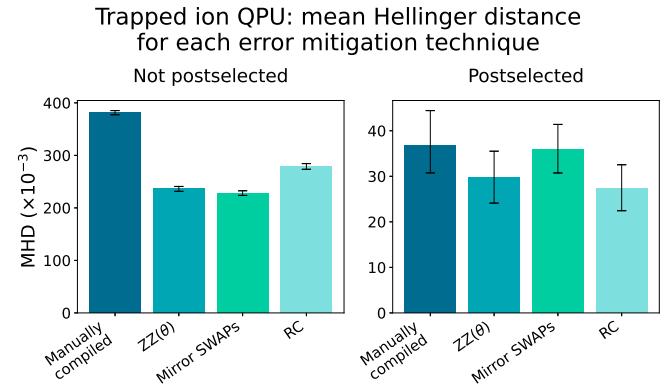


FIG. 7. Mean Hellinger Distance (see Eq. (5)) between the simulated distribution and the exact numerical solution obtained with QuTiP for each version of the algorithm executed on the QSCOUT trapped ion QPU, with and without postselection applied. Each simulation was executed with 51 timesteps, with 2,000 shots per timestep. Error bars represent 95% confidence intervals computed via bootstrap resampling with 1,000 replicates.

## V. IMPLEMENTATION ON A SUPERCONDUCTING QUANTUM TESTBED

### A. The AQT superconducting quantum processor

To test our algorithm on a superconducting system, we used a processor built and operated by the Advanced Quantum Testbed (AQT) at Lawrence Berkeley National Laboratory. We use four of the superconducting transmon qubits on the device, arranged in a linear chain,

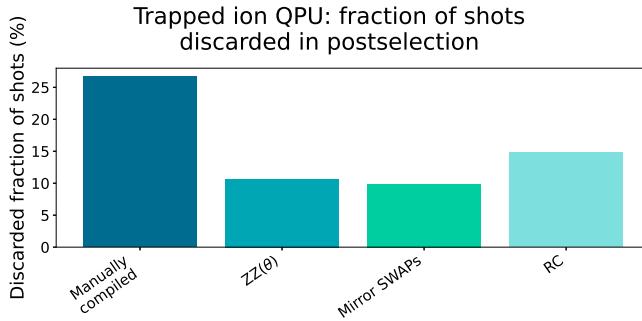


FIG. 8. Fraction of simulation shots discarded by postselection (see Sec. III A) for each version of the algorithm executed on the QSCOUT trapped ion QPU.

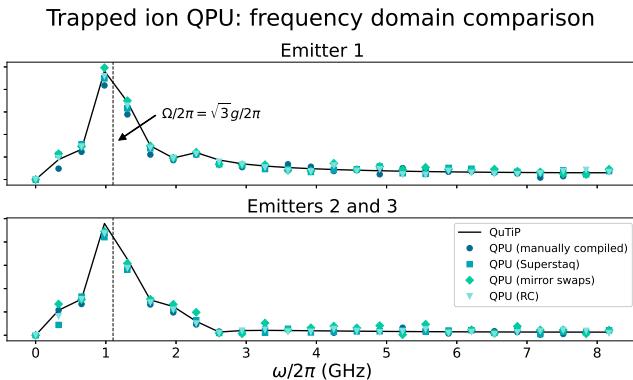


FIG. 9. Fourier transforms of the simulation results from the QSCOUT trapped ion quantum processor. The mean population has been subtracted, to eliminate the zero-frequency component. Each data point represents a simulation with 2,000 shots. The data has been postselected as described in Sec. III A. The dashed line indicates the theoretical Rabi frequency for the simulated system. The black line indicates the Fourier transform of the numerical solution obtained with QuTiP.

such that each qubit can interact with one or two adjacent qubits. Arbitrary single qubit rotations can be performed with resonant Rabi driving. The native two-qubit gate is a controlled phase (CZ) gate described in Ref. [34]. The fidelities for the CZ and single-qubit gates on this device are given in [35, Supplementary Note 4].

As before, we simulate an open Tavis-Cummings system with parameters  $N = 3$ ,  $g = 4$  GHz,  $\kappa = 2$  GHz, and the system was evolved in time from  $t = 0$  ns to  $t_{\max} = 3$  ns. All results from the AQT processor were postselected according to the procedure described in Sec. III A; the fraction of shots discarded in postselection was about 8–29% depending on the implementation.

## B. Circuit compilations

In order to adapt the simulation algorithm’s one-to-all interaction topology to the connectivity of the linear AQT processor, we added a SWAP gate to shuttle the cavity/environment qubit along the chain, allowing it to interact with each of the emitter qubits (see Fig. 10). This is analogous to the “star-to-line” routing optimization implemented by Superstaq [30]. In principle, this could extend to any length of linear qubit chain with only  $\mathcal{O}(N)$  SWAP gates, allowing for the simulation of larger systems while preserving the linear circuit depth and runtime. However it is worth noting that, in practice, even a constant-factor increase in circuit depth can markedly affect the performance.

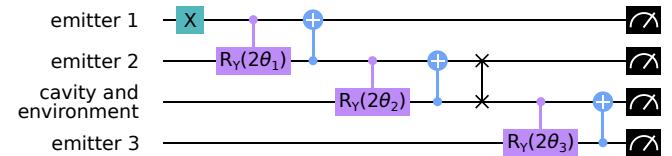


FIG. 10. The modified simulation circuit executed on AQT’s 4-qubit superconducting processor, showing the use of the SWAP to shuttle the cavity/environment qubit along the chain. The qubit labels indicate the role of the qubits at the end of the circuit, after the SWAP has interchanged the emitter 2 and cavity/environment qubits.

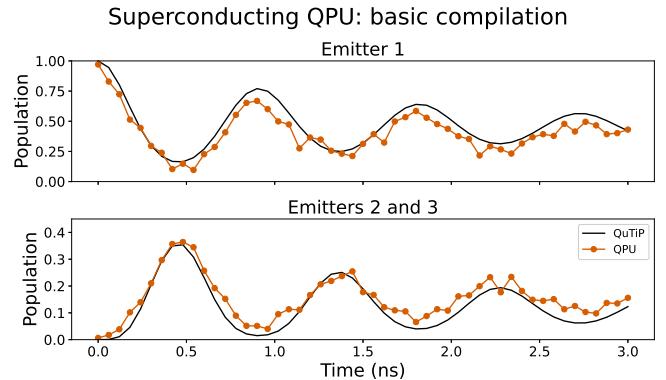


FIG. 11. Results from the AQT superconducting QPU with no error mitigation compiled into the circuit. Each data point represents a simulation with 20,000 shots; approximately 29% of shots were discarded in postselection using the technique described in Sec. III A. Wall-clock runtime of the simulation was about eight minutes. The black line indicates the exact solution of the system.

We initially executed a version of the algorithm compiled to AQT’s native gates without error mitigation. The result of running this version of the circuit (postselected as described in Sec. III A) are presented in Fig. 11, already shows clearly visible Rabi oscillations and rough agreement with the numerical solution.

### C. Error mitigation: randomized compiling (RC)

We applied randomized compiling (described in Sec. IV D) to generate the circuits executed on AQT’s superconducting hardware, using 40 and 80 random compilations; the results are presented in Fig. 12. Already at 40 randomizations the observed Rabi oscillations are markedly smoother than without error mitigation (Fig. 11), though increasing to 80 randomizations does not significantly change the results.

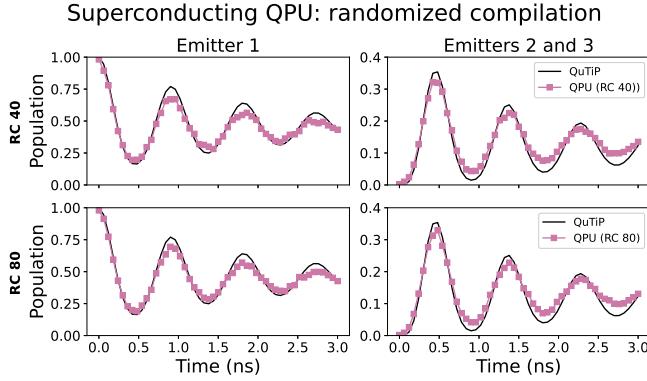


FIG. 12. Results from the AQT superconducting QPU with randomized compilation applied, using 40 (top row) and 80 (bottom row) random compilations. Each data point represents a simulation with 20,000 shots; approximately 25% of shots were discarded in postselection using the method described in Sec. III A. Wall-clock runtime of the simulation was about 17 minutes for 40 randomizations and 24 minutes for 80 randomizations. The black line represents the exact solution of the system.

### D. Error mitigation: noiseless output extrapolation (NOX)

NOX is an error mitigation technique that aims to create an unbiased estimate of what a circuit’s output would be if it were operating with zero error [36]. It achieves this by deliberately increasing the amount of error in the circuit, then extrapolating from these results back to zero error. To achieve this, each gate cycle  $\mathcal{H}$ , which is affected by some noise  $\mathcal{D}$ , is modified so that the noise is selectively amplified:  $\mathcal{D}_{\mathcal{H}}\mathcal{H} \rightarrow \mathcal{D}_{\mathcal{H}}^{\alpha}\mathcal{H}$  for some  $\alpha > 1$ . The typical noise amplification technique is known as “identity insertion”, which works by replacing  $\mathcal{H}$  by  $\mathcal{H}(\mathcal{H}\mathcal{H}^{-1})^{\alpha}$ . Provided that the  $\mathcal{H}$  and its inverse suffer the same noise, and commute with it, identity insertion efficiently amplifies the gate noise without changing the logical operation.

NOX requires errors to be stochastic in order to accurately estimate the zero-noise condition, making it a natural addition to RC which tailors coherent errors into stochastic ones. We applied NOX to the simulation circuits generated by random compilation (with 40 random-

izations) in order to further suppress noise; the results from executing this version of the algorithm are seen in Fig. 13.

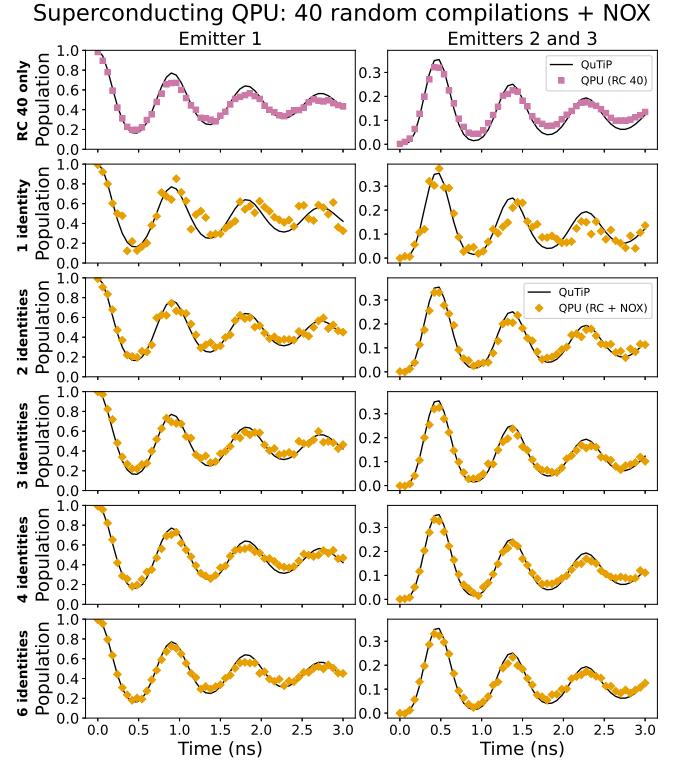


FIG. 13. Results from the AQT superconducting QPU using RC (40 random compilations) and NOX error mitigation with an increasing number of inserted identities. Each data point represents a simulation with 20,000 shots; in the NOX data, approximately 8% of shots were discarded in postselection using the technique described in Sec. III A. Wall-clock runtime of the simulation was about four hours for each iteration of NOX. The black line represents the exact solution of the system.

### E. Analysis

As described in Section IV E, we use the Mean Hellinger Distance to quantify the deviation between the simulation results from the quantum processor and the exact solution obtained numerically with QuTiP. The MHD for the three methods of error mitigation used on the AQT processor are presented in Fig. 14. When randomized compiling is applied, coherent errors in the simulation are tailored into stochastic noise whose average better represents the error-free result, curtailing the acute, random deviations from the exact solution which characterize the result without RC. In general, more random compilations will reduce coherent errors to a greater degree; this effect appears to be saturated already with 40 compilations, as evidenced by the lack of improvement when increasing to 80 random compilations.

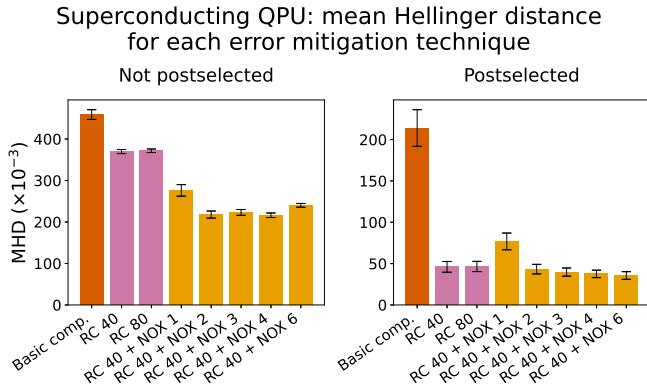


FIG. 14. Mean Hellinger Distance (see Eq. (5)) between the simulated distribution and the exact solution obtained with QuTiP for each version of the algorithm executed on the AQT superconducting QPU. Each version of the simulation algorithm propagated the initial state forward by 3 ns in 51 time steps, with 20,000 shots per time step. All data has been postselected in the manner described in Sec. III A. Error bars represent 95% confidence intervals computed via bootstrap resampling with 1,000 replicates.

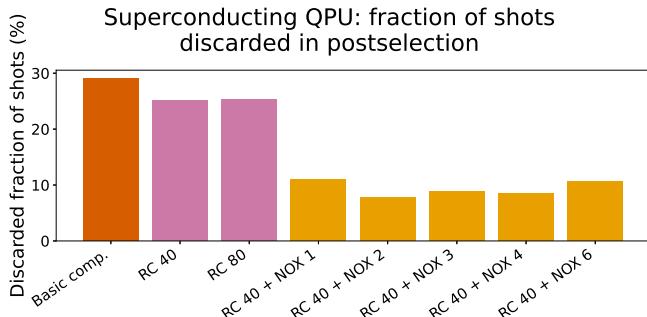


FIG. 15. Fraction of simulation shots discarded by postselection (see Sec. III A) for each version of the algorithm executed on the QSCOUT trapped ion QPU.

When NOX is used, we sample the circuit’s output at each level of amplified noise. The noise amplification needs to be large relative to the sampling error of the original circuit or else the variance of the estimate of the zero noise circuit operation will be large, leading to a larger MHD. We observe a crossover at  $\alpha = 4$  identities where the MHD of the zero noise extrapolated estimate falls below that obtained with randomized compiling alone.

We can also see the strength of NOX in mitigating error in Fig. 15 which shows the fraction of shots across the entire simulation discarded in the postselection process. This fraction remains above 20% even when RC is applied, but NOX reduces it below 10%.

As in Sec. IV E, we compare the Fourier transforms of the simulation results with those of the numerical solution obtained with QuTiP. These Fourier transforms are presented in Fig. 16 for all three versions of the algorithm executed on the AQT superconducting processor.

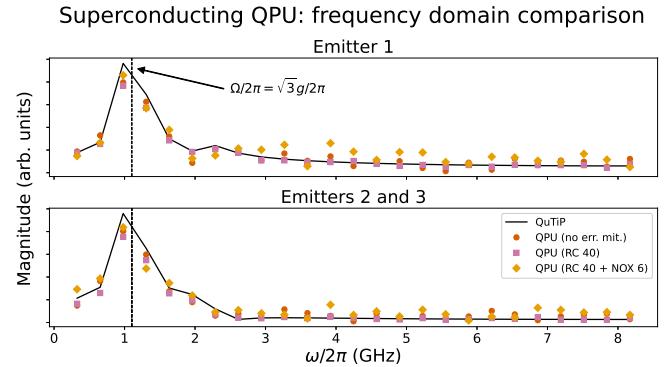


FIG. 16. Fourier transforms of the simulation results from the AQT superconducting QPU. The mean population has been subtracted, to eliminate the zero-frequency component. The underlying population data has been postselected according to the procedure described in Sec. III A. The dashed line indicates the expected Rabi frequency for the simulated system. The black line indicates the Fourier transform of the numerical solution of the system obtained with the Python package QuTiP.

As with the results from the QSCOUT processor, we find that the quantum simulations accurately obtain the Rabi rate of the system even without error mitigation.

## VI. DISCUSSION

These results show the potential for today’s NISQ-era quantum computers to accurately simulate open, many-body cavity QED physics, and the vital role that error mitigation plays in reducing noise in these simulations. The contrast between the two platforms used in this work highlights important trade-offs in quantum computing architectures. Each platform offers unique advantages: trapped ions provide native all-to-all connectivity ideal for CQED simulations, while superconducting qubits offer faster gate times and greater system stability. The nature of the noise affecting each system is also significantly different, as a consequence of arising from different physical mechanisms. As we have seen, these distinct architectural and noise characteristics have strong implications for which error mitigation techniques are most effective.

We have shown that error mitigation techniques such as RC, NOX, mirror SWAPs and optimized compilation techniques, selected on the basis of the platform at hand, can be effective at suppressing noise enough to obtain accurate results in the simulation of open CQED systems. The effectiveness of different error mitigation strategies depends strongly on the characteristic noise of each platform. Superconducting qubits primarily suffer from coherent errors that are relatively stable over time, arising from miscalibrated gate pulses or slow variation in flux voltages. This makes them well-suited to randomized compiling, which converts these coherent errors into

stochastic ones. Such stochastic errors are then effectively suppressed by NOX, since the errors present in a given gate cycle will also be present in the additional copies of that gate (and its inverse) used for identity insertion [36]. The stability of superconducting systems and their fast gate times allows them to support the relatively deep circuits generated by RC and to take a larger number of shots per simulation run, which improves statistical precision.

By contrast, ion traps exhibit noise of a more strongly stochastic nature due to trap heating and fluctuations in trap frequency. This fundamental difference in noise character makes RC less effective for trapped ions, and NOX particularly challenging to implement [37]. On the other hand, our postselection technique is particularly powerful in this situation, even in the absence of other error mitigation techniques. This can be seen by comparing the most basic version of the algorithm on each platform: postselection improves the MHD by roughly a factor of 10 for the hand-compiled version of the algorithm on the trapped ion platform, but only a factor of about 2 for the basic compilation of the algorithm on the superconducting platform. Ion trap processors also benefit from optimization strategies not focused on the statistics of the noise. Compilation with variable-angle ZZ( $\theta$ ) entangling gates and SWAP mirroring both reduced the MHD of the simulations, both with and without postselection applied. Indeed, these techniques are particularly appropriate for our algorithm, which contains many controlled rotations and thus many optimization opportunities for both compilation techniques to capitalize upon.

While the simulations presented in this work represent small systems which are a special case of the open Tavis-Cummings model, improved simulation algorithms and larger quantum computers could enable simulation of more general open quantum optical systems beyond the scale easily simulated on classical computers. The algorithm we have applied relies on a relatively simple mapping between elements of an open CQED system and computational qubits, and only applies in the special case of identical, lossless quantum emitters identically coupled to a lossy cavity. Most existing quantum algorithms for simulating open quantum systems are far too demanding in terms of qubit count to be feasible on today's NISQ machines. Yet, these algorithms are also highly general, and there may be opportunities for cost savings by considering the aspects of the problem unique to CQED.

Many-body open CQED holds great promise as a basis for future quantum technologies. Subradiant states exhibited by groups of inhomogeneous emitters coupled to a cavity may form the basis of quantum optical memories

[38]. Engineered of dissipation in these systems could be applied to quantum state preparation [39]. Many of these phenomena are challenging to study experimentally, and can be costly to simulate classically above small scales.

By applying emerging techniques of quantum simulation to open CQED systems, we can hope to achieve a virtuous cycle of technological development, in which better simulations help design better quantum devices, which in turn enable better simulations. However, starting this engine is difficult, as current quantum computers are generally too noisy to support simulations beyond what is achievable with classical computers. Until and unless large, error-corrected quantum computers become available, error mitigation will be vital to obtaining accurate simulation results. This work stands as a platform-specific guide to the best use of error mitigation in quantum simulation of open cavity quantum electrodynamical systems.

## VII. ACKNOWLEDGMENTS

Authors acknowledge support by Noyce Foundation, National Science Foundation CAREER program (Award 2047564), the UC Multicampus Research Programs and Initiatives of the University of California (Grant Number M23PL5936), Google Research Scholar Fellowship, and the Pauli Center for Theoretical Study. This material was funded in part by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research Quantum Testbed Program under contracts DE-AC02-05CH11231 and DE-NA0003525. Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525. This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

## VIII. DATA AVAILABILITY

All data from the quantum simulations, as well as the code used to solve the model with QuTiP and produce the figures in this work can be found at [github.com/radulaski/CQED\\_quantum\\_simulation](https://github.com/radulaski/CQED_quantum_simulation).

---

[1] Richard P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6):467–488, Jun 1982.

[2] Jesse Patton, Victoria A. Norman, Eliana C. Mann, Brinda Puri, Richard T. Scalettar, and Marina Radulaski. Polariton creation in coupled cavity arrays with spectrally disordered emitters, 2023.

[3] Dorit Aharonov and Amnon Ta-Shma. Adiabatic quantum state generation. *SIAM Journal on Computing*, 37(1):47–82, 2007.

[4] Dominic W. Berry, Graeme Ahokas, Richard Cleve, and Barry C. Sanders. Efficient quantum algorithms for simulating sparse hamiltonians. *Communications in Mathematical Physics*, 270(2):359–371, Mar 2007.

[5] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.

[6] Andrew J. Daley, Immanuel Bloch, Christian Kokail, Stuart Flannigan, Natalie Pearson, Matthias Troyer, and Peter Zoller. Practical quantum advantage in quantum simulation. *Nature*, 607(7920):667–676, Jul 2022.

[7] Isaac H. Kim, Ye-Hua Liu, Sam Pallister, William Pol, Sam Roberts, and Eunseok Lee. Fault-tolerant resource estimate for quantum chemical simulations: Case study on li-ion battery electrolyte molecules. *Phys. Rev. Res.*, 4:023019, Apr 2022.

[8] Sridhar Majety, Pranta Saha, Victoria A. Norman, and Marina Radulaski. Quantum information processing with integrated silicon carbide photonics. *Journal of Applied Physics*, 131(13):130901, 04 2022.

[9] Galan Moody, Volker J Sorger, Daniel J Blumenthal, Paul W Juodawlkis, William Loh, Cheryl Sorace-Agaskar, Alex E Jones, Krishna C Balram, Jonathan C F Matthews, Anthony Laing, Marcelo Davanco, Lin Chang, John E Bowers, Niels Quack, Christophe Galland, Igor Aharonovich, Martin A Wolff, Carsten Schuck, Neil Sinclair, Marko Lončar, Tin Komljenovic, David Weld, Shayan Mookherjea, Sonia Buckley, Marina Radulaski, Stephan Reitzenstein, Benjamin Pingault, Bartholomeus Machielse, Debsuvra Mukhopadhyay, Alexey Akimov, Aleksei Zheltikov, Girish S Agarwal, Kartik Srinivasan, Juanjuan Lu, Hong X Tang, Wentao Jiang, Timothy P McKenna, Amir H Safavi-Naeini, Stephan Steinhauer, Ali W Elshaari, Val Zwiller, Paul S Davids, Nicholas Martinez, Michael Gehl, John Chiaverini, Karan K Mehta, Jacquiline Romero, Navin B Lingaraju, Andrew M Weiner, Daniel Peace, Robert Cernansky, Mirko Lobino, Eleni Diamanti, Luis Trigo Vidarte, and Ryan M Camacho. 2022 roadmap on integrated quantum photonics. *Journal of Physics: Photonics*, 4(1):012501, jan 2022.

[10] H. J. Kimble. The quantum internet. *Nature*, 453(7198):1023–1030, Jun 2008.

[11] M. Kliesch, T. Barthel, C. Gogolin, M. Kastoryano, and J. Eisert. Dissipative quantum church-turing theorem. *Phys. Rev. Lett.*, 107:120501, Sep 2011.

[12] Richard Cleve and Chunhao Wang. Efficient quantum algorithms for simulating lindblad evolution, 2019.

[13] Andrew M. Childs and Tongyang Li. Efficient simulation of sparse markovian quantum dynamics. *Quantum Information and Computation*, 17(11 & 12), September 2017.

[14] R. Di Candia, J. S. Pedernales, A. del Campo, E. Solano, and J. Casanova. Quantum simulation of dissipative processes without reservoir engineering. *Scientific Reports*, 5(1):9981, May 2015.

[15] Dhrumil Patel and Mark M. Wilde. Wave matrix lindbladization i: Quantum programs for simulating markovian dynamics, 2023.

[16] Dhrumil Patel and Mark M. Wilde. Wave matrix lindbladization ii: General lindbladians, linear combinations, and polynomials, 2023.

[17] Marina Krstic Marinkovic and Marina Radulaski. Singly-excited resonant open quantum system tavis-cummings model with quantum circuit mapping. *Scientific Reports*, 13(1):19435, Nov 2023.

[18] Michael Tavis and Frederick W. Cummings. Exact solution for an  $n$ -molecule—radiation-field hamiltonian. *Phys. Rev.*, 170:379–384, Jun 1968.

[19] J.R. Johansson, P.D. Nation, and Franco Nori. Qutip 2: A python framework for the dynamics of open quantum systems. *Computer Physics Communications*, 184(4):1234–1240, 2013.

[20] A. R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54:1098–1105, Aug 1996.

[21] Emanuel Knill, Raymond Laflamme, and Wojciech H. Zurek. Resilient quantum computation. *Science*, 279(5349):342–345, 1998.

[22] Rajeev Acharya, Laleh Aghababaie-Beni, Igor Aleiner, Trond I. Andersen, Markus Ansmann, Frank Arute, Kunal Arya, Abraham Asfaw, Nikita Astrakhantsev, Juan Atalaya, Ryan Babbush, Dave Bacon, Brian Balfour, Joseph C. Bardin, Johannes Bausch, Andreas Bengtsson, Alexander Bilmes, Sam Blackwell, Sergio Boixo, Gina Bortoli, Alexandre Bourassa, Jenna Bovaird, Leon Brill, Michael Broughton, David A. Browne, Brett Bucchea, Bob B. Buckley, David A. Buell, Tim Burger, Brian Burkett, Nicholas Bushnell, Anthony Cabrera, Juan Campero, Hung-Shen Chang, Yu Chen, Zijun Chen, Ben Chiaro, Desmond Chik, Charina Chou, Jahan Claes, Agnetta Y. Cleland, Josh Cogan, Roberto Collins, Paul Conner, William Courtney, Alexander L. Crook, Ben Curtin, Sayan Das, Alex Davies, Laura De Lorenzo, Dripto M. Debroy, Sean Demura, Michel Devoret, Agustin Di Paolo, Paul Donohoe, Ilya Drozdov, Andrew Dunsworth, Clint Earle, Thomas Edlich, Alec Eickbusch, Aviv Moshe Elbag, Mahmoud Elzouka, Catherine Erickson, Lara Faoro, Edward Farhi, Vinius S. Ferreira, Leslie Flores Burgos, Ebrahim Forati, Austin G. Fowler, Brooks Foxen, Suhas Ganjam, Gonzalo Garcia, Robert Gasca, Élie Genois, William Giang, Craig Gidney, Dar Gilboa, Raja Gosula, Alejandro Grajales Dau, Dietrich Graumann, Alex Greene, Jonathan A. Gross, Steve Habegger, John Hall, Michael C. Hamilton, Monica Hansen, Matthew P. Harrigan, Sean D. Harrington, Francisco J. H. Heras, Stephen Heslin, Paula Heu, Oscar Higgott, Gordon Hill, Jeremy Hilton, George Holland, Sabrina Hong, Hsin-Yuan Huang, Ashley Huff, William J. Huggins, Lev B. Ioffe, Sergei V. Isakov, Justin Iveland, Evan Jeffrey, Zhang Jiang, Cody Jones, Stephen Jordan, Chaitali Joshi, Pavol Juhas, Dvir Kafri, Hui Kang, Amir H. Karamlou, Kostyantyn Kechedzhi, Julian Kelly, Trupti Khaire, Tanuj Khattar, Mostafa Khezri, Seon Kim, Paul V. Klimov, Andrey R. Klots, Bryce Kobrin, Pushmeet Kohli, Alexander N. Korotkov, Fedor Kostritsa, Robin Kothari, Borislav Kozlovskii, John Mark Kreikebaum, Vladislav D. Kurilovich, Nathan Lacroix, David Landhuis, Tiano Lange-Dei, Brandon W. Langley, Pavel Laptev, Kim-Ming Lau, Loïck Le Guevel, Justin Ledford, Kenny Lee, Yuri D. Lensky, Shannon Leon, Brian J. Lester, Wing Yan Li, Yin Li, Alexander T. Lill, Wayne Liu, William P. Livingston, Aditya Locharla, Erik Lucero, Daniel Lundahl, Aaron Lunt, Sid Madhuk, Fionn D. Malone, Ashley Malone, Salvatore Mandrá, Leigh S. Martin, Steven Martin, Orion Martin, Cameron

Maxfield, Jarrod R. McClean, Matt McEwen, Seneca Meeks, Anthony Megrant, Xiao Mi, Kevin C. Miao, Amanda Mieszala, Reza Molavi, Sebastian Molina, Shirin Montazeri, Alexis Morvan, Ramis Movassagh, Wojciech Mruczkiewicz, Ofer Naaman, Matthew Neeley, Charles Neill, Ani Nersisyan, Hartmut Neven, Michael Newman, Jium How Ng, Anthony Nguyen, Murray Nguyen, Chia-Hung Ni, Thomas E. O'Brien, William D. Oliver, Alex Opremicak, Kristoffer Ottosson, Andre Petukhov, Alex Pizzuto, John Platt, Rebecca Potter, Orion Pritchard, Leonid P. Pryadko, Chris Quintana, Ganesh Ramachandran, Matthew J. Reagor, David M. Rhodes, Gabrielle Roberts, Elliott Rosenberg, Emma Rosenfeld, Pedram Roushan, Nicholas C. Rubin, Negar Saei, Daniel Sank, Kannan Sankaragomathi, Kevin J. Satzinger, Henry F. Schurkus, Christopher Schuster, Andrew W. Senior, Michael J. Shearn, Aaron Shorter, Noah Shutty, Vladimir Shvarts, Shraddha Singh, Volodymyr Sivak, Jindra Skruzny, Spencer Small, Vadim Smelyanskiy, W. Clarke Smith, Rolando D. Somma, Sofia Springer, George Sterling, Doug Strain, Jordan Suchard, Aaron Szasz, Alex Sztein, Douglas Thor, Alfredo Torres, M. Mert Torunbalci, Abeer Vaishnav, Justin Vargas, Sergey Vdovichev, Guifre Vidal, Benjamin Villalonga, Catherine Vollgraff Heidweiller, Steven Waltman, Shannon X. Wang, Brayden Ware, Kate Weber, Theodore White, Kristi Wong, Bryan W. K. Woo, Cheng Xing, Z. Jamie Yao, Ping Yeh, Bicheng Ying, Juhwan Yoo, Noureldin Yosri, Grayson Young, Adam Zalcman, Yaxing Zhang, Ningfeng Zhu, and Nicholas Zobrist. Quantum error correction below the surface code threshold, 2024.

[23] A. Paetznick, M. P. da Silva, C. Ryan-Anderson, J. M. Bello-Rivas, J. P. Campora III, A. Chernoguzov, J. M. Dreiling, C. Foltz, F. Frachon, J. P. Gaebler, T. M. Gatterman, L. Grans-Samuelsson, D. Gresh, D. Hayes, N. Hewitt, C. Holliman, C. V. Horst, J. Johansen, D. Lucchetti, Y. Matsuoka, M. Mills, S. A. Moses, B. Neyenhuys, A. Paz, J. Pino, P. Siegfried, A. Sundaram, D. Tom, S. J. Wernli, M. Zanner, R. P. Stutz, and K. M. Svore. Demonstration of logical qubits and repeated error correction with better-than-physical error rates, 2024.

[24] Zhenyu Cai, Ryan Babbush, Simon C. Benjamin, Suguru Endo, William J. Huggins, Ying Li, Jarrod R. McClean, and Thomas E. O'Brien. Quantum error mitigation. *Rev. Mod. Phys.*, 95:045005, Dec 2023.

[25] Susan M. Clark, Daniel Lobser, Melissa C. Revelle, Christopher G. Yale, David Bossert, Ashlyn D. Burch, Matthew N. Chow, Craig W. Hogle, Megan Ivory, Jessica Pehr, Bradley Salzbrenner, Daniel Stick, William Sweatt, Joshua M. Wilson, Edward Winrow, and Peter Maunz. Engineering the quantum scientific computing open user testbed. *IEEE Transactions on Quantum Engineering*, 2:1–32, 2021.

[26] P. J. Lee, K. A. Brickman, L. Deslauriers, P. C. Haljan, L. M. Duan, and C. Monroe. Phase control of trapped ion quantum gates. *Journal of Optics B: Quantum and Semiclassical Optics*, 2005.

[27] Matthew N. H. Chow, Christopher G. Yale, Ashlyn D. Burch, Megan Ivory, Daniel S. Lobser, Melissa C. Revelle, and Susan M. Clark. First-order crosstalk mitigation in parallel quantum gates driven with multi-photon transitions. *Applied Physics Letters*, 124(4):044002, 01 2024.

[28] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2011. See Exercise 4.20 in Section 4.3.

[29] Dmitri Maslov. Basic circuit compilation techniques for an ion-trap quantum machine. *New Journal of Physics*, 19(2):023035, feb 2017.

[30] Colin Campbell, Frederic T. Chong, Denny Dahl, Paige Frederick, Palash Goiporia, Pranav Gokhale, Benjamin Hall, Salahedeen Issa, Eric Jones, Stephanie Lee, Andrew Litteken, Victory Omole, David Owusu-Antwi, Michael A. Perlin, Rich Rines, Kaitlin N. Smith, Noah Goss, Akel Hashim, Ravi Naik, Ed Younis, Daniel Lobser, Christopher G. Yale, Benchen Huang, and Ji Liu. Superstaq: Deep optimization of quantum programs, 2023.

[31] Christopher G. Yale, Rich Rines, Victory Omole, Bharath Thotakura, Ashlyn D. Burch, Matthew N. H. Chow, Megan Ivory, Daniel Lobser, Brian K. McFarland, Melissa C. Revelle, Susan M. Clark, and Pranav Gokhale. Noise-aware circuit compilations for a continuously parameterized two-qubit gateset, 2024.

[32] Akel Hashim, Ravi K. Naik, Alexis Morvan, Jean-Loup Ville, Bradley Mitchell, John Mark Kreikebaum, Marc Davis, Ethan Smith, Costin Iancu, Kevin P. O'Brien, Ian Hincks, Joel J. Wallman, Joseph Emerson, and Irfan Siddiqi. Randomized compiling for scalable quantum computing on a noisy superconducting quantum processor. *Phys. Rev. X*, 11:041039, Nov 2021.

[33] Joel J. Wallman and Joseph Emerson. Noise tailoring for scalable quantum computation via randomized compiling. *Phys. Rev. A*, 94:052325, Nov 2016.

[34] Bradley K. Mitchell, Ravi K. Naik, Alexis Morvan, Akel Hashim, John Mark Kreikebaum, Brian Marinelli, Wim Lavrijsen, Kasra Nowrouzi, David I. Santiago, and Irfan Siddiqi. Hardware-efficient microwave-activated tunable coupling between superconducting qubits. *Phys. Rev. Lett.*, 127:200502, Nov 2021.

[35] Shi-Ning Sun, Brian Marinelli, Jin Ming Koh, Yosep Kim, Long B. Nguyen, Larry Chen, John Mark Kreikebaum, David I. Santiago, Irfan Siddiqi, and Austin J. Minnich. Quantum computation of frequency-domain molecular response properties using a three-qubit itoffoli gate. *npj Quantum Information*, 10(1):55, May 2024.

[36] Samuele Ferracin, Akel Hashim, Jean-Loup Ville, Ravi Naik, Arnaud Carignan-Dugas, Hammam Qassim, Alexis Morvan, David I. Santiago, Irfan Siddiqi, and Joel J. Wallman. Efficiently improving the performance of noisy quantum computers, 2022.

[37] Oliver G. Maupin, Ashlyn D. Burch, Christopher G. Yale, Brandon Ruzic, Antonio Russo, Daniel S. Lobser, Melissa C. Revelle, Matthew N. Chow, Susan M. Clark, Andrew J. Landahl, and Peter J. Love. Error mitigation, optimization, and extrapolation on a trapped ion testbed, 2023.

[38] Mi Lei, Rikuto Fukumori, Jake Rochman, Bihui Zhu, Manuel Endres, Joonhee Choi, and Andrei Faraon. Many-body cavity quantum electrodynamics with driven inhomogeneous emitters. *Nature*, 617(7960):271–276, May 2023.

[39] M. J. Kastoryano, F. Reiter, and A. S. Sørensen. Dissipative preparation of entanglement in optical cavities. *Phys. Rev. Lett.*, 106:090502, Feb 2011.

[40] Brian Marinelli, Alex H. Rubin, Victoria A. Norman, Zainab Rizvi, Ravi Naik, David I. Santiago, Christopher Spitzer, John Mark Kreikebaum, Marina Krstic-Marinkovic, Irfan Siddiqi, and Marina Radulaski. Digital

tavis-cummings simulation on superconducting quantum hardware with error mitigation. In *Optica Quantum 2.0 Conference and Exhibition*, page QM2A.3. Optica Publishing Group, 2023.

### Appendix A: QSCOUT gate fidelities

The data from QSCOUT was collected in four runs with 500 shots each. In order to minimize the effect of system drift on the results, the four versions of the circuit were interleaved within each run. Tables I-IV give the fidelities of the 2-qubit bare MS gate measured during each run, for each pair of qubits used in the circuit. Single qubit gate fidelities for these runs were 99.3%.

Gate Pair	Connectivity	Fidelity Upper Bound
{q[0], q[1]}	nearest neighbor	0.984 <sup>+0.008</sup> <sub>-0.009</sub>
{q[0], q[2]}	nearest neighbor	0.993 <sup>+0.006</sup> <sub>-0.008</sub>
{q[0], q[3]}	next-nearest neighbor	0.976 <sup>+0.009</sup> <sub>-0.010</sub>
{q[1], q[3]}	nearest neighbor	0.992 <sup>+0.006</sup> <sub>-0.008</sub>
{q[2], q[3]}	outer	0.985 <sup>+0.008</sup> <sub>-0.009</sub>

TABLE I. Bare MS entangling gate fidelities from QSCOUT run 0.

Gate Pair	Connectivity	Fidelity Upper Bound
{q[0], q[1]}	nearest neighbor	0.975 <sup>+0.009</sup> <sub>-0.010</sub>
{q[0], q[2]}	nearest neighbor	0.983 <sup>+0.008</sup> <sub>-0.009</sub>
{q[0], q[3]}	next-nearest neighbor	0.983 <sup>+0.008</sup> <sub>-0.010</sub>
{q[1], q[3]}	nearest neighbor	0.990 <sup>+0.007</sup> <sub>-0.009</sub>
{q[2], q[3]}	outer	0.978 <sup>+0.008</sup> <sub>-0.010</sub>

TABLE II. Bare MS entangling gate fidelities from QSCOUT run 1.

Gate Pair	Connectivity	Fidelity Upper Bound
{q[0], q[1]}	nearest neighbor	0.988 <sup>+0.006</sup> <sub>-0.008</sub>
{q[0], q[2]}	nearest neighbor	0.990 <sup>+0.006</sup> <sub>-0.009</sub>
{q[0], q[3]}	next-nearest neighbor	0.984 <sup>+0.008</sup> <sub>-0.009</sub>
{q[1], q[3]}	nearest neighbor	0.988 <sup>+0.006</sup> <sub>-0.008</sub>
{q[2], q[3]}	outer	0.981 <sup>+0.008</sup> <sub>-0.010</sub>

TABLE III. Bare MS entangling gate fidelities from QSCOUT run 2.

Gate Pair	Connectivity	Fidelity Upper Bound
$\{q[0], q[1]\}$	nearest neighbor	$0.995^{+0.005}_{-0.007}$
$\{q[0], q[2]\}$	nearest neighbor	$0.988^{+0.007}_{-0.009}$
$\{q[0], q[3]\}$	next-nearest neighbor	$0.988^{+0.007}_{-0.009}$
$\{q[1], q[3]\}$	nearest neighbor	$0.994^{+0.005}_{-0.007}$
$\{q[2], q[3]\}$	outer	$0.985^{+0.007}_{-0.009}$

TABLE IV. Bare MS entangling gate fidelities from QSCOUT run 3.