Digital Quantum Simulation of Open Quantum Systems Using Quantum Imaginary–Time Evolution

Hirsh Kamakari[ORCID],[1] Shi-Ning Sun[ORCID],[1] Mario Motta[ORCID],[2] and Austin J. Minnich[ORCID][1,*]

[1]*Division of Engineering and Applied Science, California Institute of Technology, Pasadena, California 91125, USA*

[2]*IBM Quantum, IBM Research Almaden, San Jose, California 95120, USA*

Quantum simulation on emerging quantum hardware is a topic of intense interest. While many studies focus on computing ground-state properties or simulating unitary dynamics of closed systems, open quantum systems are an interesting target of study owing to their ubiquity and rich physical behavior. However, their nonunitary dynamics are also not natural to simulate on digital quantum devices. Here, we report algorithms for the digital quantum simulation of the dynamics of open quantum systems governed by a Lindblad equation using adaptations of the quantum imaginary–time evolution algorithm. We demonstrate the algorithms on IBM Quantum's hardware with simulations of the spontaneous emission of a two-level system and the dissipative transverse field Ising model. Our work advances efforts to simulate the dynamics of open quantum systems on quantum hardware.

I. INTRODUCTION

The development of quantum algorithms to simulate the dynamics of quantum many-body systems is now a topic of interest owing to advances in quantum hardware [1–3]. While the real-time evolution of closed quantum systems on digital quantum computers has been extensively studied in the context of spin models [4–10], fermionic systems [11,12], electron-phonon interactions [13], and quantum field theories [14–16], fewer studies have considered the time evolution of open quantum systems, which exhibit rich dynamical behavior due to coupling of the system to its environment [17,18]. However, this coupling leads to nonunitary evolution that is not naturally simulable on quantum hardware.

Early approaches to overcome this challenge included use of the quantum simulators' intrinsic decoherence [19] and direct simulation of the environment [20–22]. Theoretical works examined the resources required for efficient quantum simulation of Markovian dynamics [23–25], concluding that arbitrary quantum channels can be efficiently simulated by combining elementary quantum channels. Recently, several algorithms have been proposed for the

digital quantum simulation of open quantum systems on the basis of the Kraus decomposition of quantum channels [26–31] as well as variational descriptions of general processes to simulate the stochastic Schrödinger equation [1,8] and the Lindblad equation [32]. Recently, explicit Trotterization of the Lindblad equation was used to simulate damping and dephasing of a single qubit using an additional ancilla qubit [33].

Simulation via Kraus decomposition is convenient when the Kraus operators corresponding to the time evolution of the system are known, such as modeling decoherence with amplitude damping or depolarizing channels. However, determining the Kraus operators of a general system requires either computing the full unitary evolution of both the system and environment or casting a master equation into an operator-sum representation for the density operator. The latter procedure can be approximated analogously to Trotterization [29,31], but requires either reset of ancillae qubits or a qubit overhead that scales linearly with the number of time steps in the simulation. Exactly determining the Kraus operators from the Lindblad equation is a classically hard task that is equivalent to solving the master equation [34] and so can only be applied to small systems. Explicit Trotterization circumvents the need to determine the Kraus operators representing the time evolution, but has the same ancilla qubit overhead as in Kraus decomposition methods. Variational approaches [1,8,35] offer an alternative for simulating open system dynamics, but as in the case of closed systems, require an ansatz and a potentially high-dimensional classical optimization

---

*aminnich@caltech.edu

that is an NP-hard problem [36]. A quantum simulation of the stochastic Schrödinger equation was emulated in Ref. [8]. In this case, the quantum jumps, or discontinuous changes in the quantum state, were implemented via variational matrix-vector multiplication, thus incurring the disadvantages previously mentioned for variational approaches.

The common feature of the above algorithms is that they reformulate nonunitary open system dynamics into unitary dynamics that can be simulated on a quantum computer. A similar approach is used in variational approaches to imaginary–time evolution [37] and the quantum imaginary–time evolution (QITE) algorithm, which has recently been introduced as a way to prepare ground states and compute thermal averages on near-term devices [38]. QITE has since been used to compute finite-temperature correlation functions of many-body systems [39], scattering in the Ising model [40], and binding energies in quantum chemistry [41,42] and nuclear physics [42]. It is therefore natural to consider how QITE might be adapted for open quantum system evolution.

Here, we report quantum algorithms to simulate open quantum dynamics using adaptations of the QITE algorithm and demonstrate them on IBM Quantum hardware. The first algorithm casts the Lindblad equation for the density operator into a Schrödinger-type equation with a non-Hermitian Hamiltonian. Time evolution is then achieved by simulating the unitary evolution via Trotterization, corresponding to the Hermitian component of the Hamiltonian and using QITE to simulate the anti-Hermitian component of the Hamiltonian. The second algorithm expresses the density operator in terms of an ansatz that is preserved during both real- and imaginary–time evolution. We demonstrate these algorithms on IBM Quantum hardware for two cases: the spontaneous emission of a two-level system in a heat bath at zero temperature, and the dissipative transverse field Ising model on two sites. We observe good agreement between the exact and hardware results, showing that the dynamics of open quantum systems are accessible on near-term quantum hardware.

II. THEORY

The dynamics of a Markovian open quantum system can be described by the Lindblad equation

$$\frac{d\rho}{dt} = -i[H, \rho] + \sum_k \left(L_k \rho L_k^\dagger - \frac{1}{2}\{L_k^\dagger L_k, \rho\} \right), \quad (1)$$

where ρ is the density operator of the system, H is the system's Hamiltonian, and the L_k are operators describing the coupling to the environment. The master equation in Lindblad form is often derived assuming weak coupling

between system and environment and absence of memory effects (Born-Markov approximation) [18,43].

We present two algorithms to simulate the master equation in Lindblad form on a digital quantum computer. The first quantum algorithm, based on a vectorization of the density operator, is described in Sec. II A; the second algorithm, which combines a QITE adaptation with an ansatz for the time-dependent density operator, is presented in Sec. II B.

A. Algorithm I

The Lindblad equation can be rewritten as a Schrödinger-type equation with a non-Hermitian Hamiltonian by transforming the $2^n \times 2^n$ density operator ρ into a 4^n component vector $|\rho\rangle$ by column stacking the density operator [44]. The resulting transformation of the Lindblad equation is

$$\frac{d|\rho\rangle}{dt} = \left[-i\mathbb{I} \otimes H + iH^\top \otimes \mathbb{I} + \sum_k \left(\overline{L_k} \otimes L_k \right. \right.$$
$$\left. \left. - \frac{1}{2}\mathbb{I} \otimes (L_k^\dagger L_k) - \frac{1}{2}(L_k^\top \overline{L_k}) \otimes \mathbb{I} \right) \right] |\rho\rangle, \quad (2)$$

where the bar indicates entrywise complex conjugation and $|\rho\rangle = |\rho(t)\rangle$ is the vectorized density operator [44]. Separating Eq. (2) into Hermitian and anti-Hermitian parts, the time evolution of the initial state can be written as

$$|\rho(t)\rangle = \exp\left[-i(H_1 - iH_2)t\right]|\rho(0)\rangle = [\exp\left(-iH_1\tau\right)$$
$$\times \exp\left(-H_2\tau\right)]^N |\rho(0)\rangle + \mathcal{O}(\tau^2 N), \quad (3)$$

where in the last equality we have Trotterized to first order with a time step of $\tau = t/N$, and H_1 and iH_2 are the Hermitian and anti-Hermitian components of the vectorized Hamiltonian, respectively. The first term $\exp\left(-iH_1\tau\right)$ is unitary and can be implemented on a quantum simulator via Trotterization and standard quantum simulation techniques [2,45,46]. The term $\exp\left(-H_2\tau\right)$ is nonunitary and so cannot be directly applied to the quantum register. Instead, we implement it on a digital quantum simulator via analogy to quantum algorithms for imaginary–time evolution [38].

Imaginary–time evolution of the Schrödinger equation with Hamiltonian H is carried out formally by substituting $\beta = it$ into the real-time propagator $\exp(-itH)$. This technique is typically used to find ground states $|\psi\rangle = \lim_{\beta\to\infty} |\phi(\beta)\rangle / |||\phi(\beta)\rangle||$, where $|\phi(\beta)\rangle = \exp\left(-\beta H\right)|\phi(0)\rangle$ and $|\phi(0)\rangle$ has nonzero overlap with a ground state. If we interpret H_2 as the Hamiltonian of a system in the extended Hilbert space, $\exp\left(-H_2\tau\right)$ is an imaginary–time evolution operator generated by H_2. The full time evolution is then applied as sequence of real- and imaginary–time evolutions, as shown in Fig. 1(a).
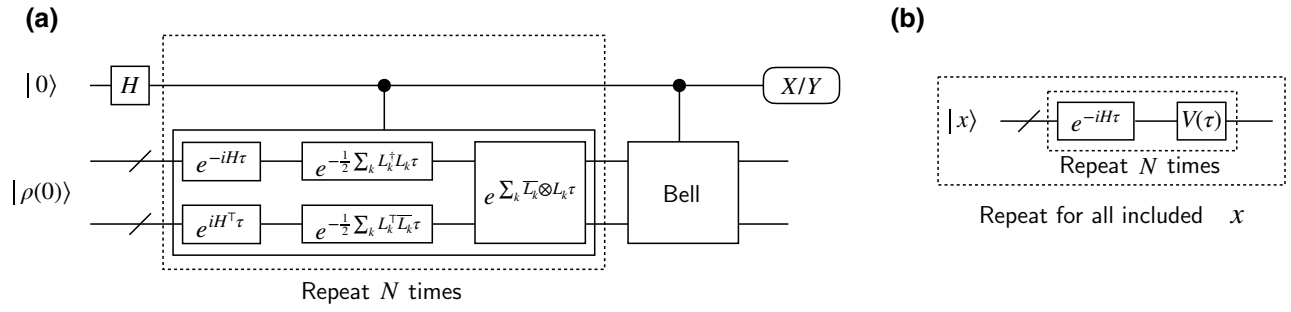
FIG. 1. Circuit diagrams for Trotterized time evolution of the density operator with N Trotter steps. (a) Time evolution for the vectorized density operator $|\rho\rangle$ (Algorithm I). Here $e^{-iH_1\tau}$ is a unitary operator and can be directly implemented on the quantum simulator. The nonunitary terms $e^{-(1/2)\sum_k L_k^\dagger L_k \tau}$ and $e^{\sum_k \overline{L_k} \otimes L_k \tau}$ are implemented via QITE. The unitary labeled "Bell" represents a unitary preparing the generalized $2n$-qubit Bell state. (b) Time evolution for the purification-based algorithm (Algorithm II). Here x is a bit string included in the index set I; $V(\tau)$ represents the nonunitary terms that need to be applied to the system for a time step τ. In both figures the slash (/) denotes a bundle of n qubits.

We present a brief review of the QITE algorithm reported in Ref. [38] that is used as a subroutine in this work. The QITE algorithm represents normalized imaginary–time evolution in terms of unitary evolution as

$$\frac{e^{-\beta H}|\psi\rangle}{||e^{-\beta H}|\psi\rangle||} = e^{-iA}|\psi\rangle, \qquad (4)$$

where H is the system Hamiltonian, β is the imaginary time, and A is a Hermitian operator. The operator A can be represented with real coefficients in a complete basis of Hermitian operators, typically chosen to be the Pauli strings σ_i over the qubits of the system:

$$A = \sum_i a_i \sigma_i. \qquad (5)$$

For an imaginary–time step β, the coefficients a_i are determined (up to order β^2) by the linear system $Sa = b$, with

$$S_{ij} = \langle\psi|\sigma_i^\dagger \sigma_j|\psi\rangle,$$
$$b_i = \frac{-i}{\sqrt{c}}\langle\psi|\sigma_i^\dagger H|\psi\rangle, \qquad (6)$$

where $c = \langle\exp(-2\beta H)\rangle$ is the norm squared of the unnormalized imaginary–time evolved state.

Once the desired time and state $|\rho(t)\rangle$ are reached, measurements of an observable O are obtained by evaluating the expectation value $\langle O\rangle(t) = \mathrm{Tr}[O\rho(t)]$ as $\langle O^\dagger|\rho\rangle$. Here $|O\rangle$ is the vector obtained from column stacking the matrix representation of O and so only the matrix representation of O in the computational basis is needed for this step. Lindbladian evolution preserves $\mathrm{Tr}(\rho)$ whereas the algorithm preserves $\mathrm{Tr}(\rho^2) = \langle\rho|\rho\rangle$, meaning that the operator ρ obtained from matricizing $|\rho(t)\rangle$ is not strictly

a density matrix. However, the final state can be renormalized to have unit trace as $\rho'(t) = \rho(t)/\mathrm{Tr}[\rho(t)]$. In practice, we normalize the final expectation value of a given observable instead. The final physical observables are thus given by $\langle O\rangle/\mathrm{Tr}(\rho)$. Therefore, obtaining measurements of observables on the state requires evaluating both $\langle O\rangle$ and $\mathrm{Tr}(\rho)$ at each time step.

Both quantities $\langle O\rangle(t)$ and $\mathrm{Tr}[\rho(t)]$ can be obtained using a Hadamard test circuit [47]. In particular, $\mathrm{Tr}(\rho)$ can be evaluated up to a prefactor of $2^{-n/2}P^{-1/2}$ as $\langle 0|V^\dagger U|0\rangle$, where U is the circuit that prepares $|\rho\rangle$, V prepares the generalized Bell state $|\beta\rangle = 2^{-n/2}\sum_x |x\rangle \otimes |x\rangle$, $|x\rangle$ are the computational basis states on n qubits, and P is the purity of the initial state. Preparing the $2n$-qubit Bell state requires n Hadamard and n controlled NOT (CNOT) gates. Assuming that $|\rho\rangle = U|0\rangle$ for a unitary U with gate decomposition requiring u_1 and u_2 single-qubit and CNOT gates, respectively, the measurement of $\mathrm{Tr}(\rho)$ requires a circuit with $\mathcal{O}(n + u_1)$ single-qubit gates, $\mathcal{O}(n + u_1)$ CNOT gates, and $\mathcal{O}(n + u_2)$ CCNOT gates.

Measurement of k-local observables can be carried out similarly. We assume here without loss of generality that the k-local observable O has support on the first k qubits. The vectorized state can then be written as $|\rho\rangle = P^{-1/2}\sum_{x_1,x_2,y_1,y_2} \rho_{x_1 x_2 y_1 y_2}|x_1 x_2 y_1 y_2\rangle$, where x_1, y_1 and x_2, y_2 are length k and $(n - k)$ bit strings, respectively, and P is the purity of the initial state. Defining the state

$$|O^\dagger\rangle = \sum_{x_1 y_1 z} \frac{\overline{O_{x_1 y_1}}}{\sqrt{2^{n-k}\mathrm{Tr}(O^\dagger O)}}|x_1 z y_1 z\rangle, \qquad (7)$$

where the overbar indicates complex conjugation, the expectation value of O can be evaluated (up to a prefactor)

as

$$\langle O^\dagger|\rho\rangle = \sum_{x_1y_1z} \frac{O_{x_1y_1}}{\sqrt{2^{n-k}}} \frac{\rho_{x_1zy_1z}}{\sqrt{P}} = \frac{\mathrm{Tr}(O\rho)}{\sqrt{2^{n-k}\mathrm{Tr}(O^\dagger O)P}}. \quad (8)$$

The state $|O^\dagger\rangle$ can be prepared as

$$U_{O^\dagger}V_{n-k}|0_k, 0_{n-k}, 0_k, 0_{n-k}\rangle = U_{O^\dagger}\left[\frac{1}{\sqrt{2^{n-k}}}\sum_z |0_k, z, 0_k, z\rangle\right], \quad (9)$$

where V_{n-k} prepares the $n-k$ generalized Bell state and $U_{O^\dagger}$ prepares the $2k$-qubit state

$$|O^\dagger\rangle = \sum_{x_1y_1} \frac{\overline{O_{x_1y_1}}}{\sqrt{\mathrm{Tr}(O^\dagger O)}}|x_1y_1\rangle. \quad (10)$$

We then measure the unnormalized expectation value of O using the Hadamard test. Since the purity is conserved by the algorithm, all observables can be renormalized after the measurement. Assuming a decomposition of U into u_1 and u_2 single-qubit and CNOT gates, respectively, and V into v_1 and v_2 single-qubit and CNOT gates, the total overhead for measurement of observables (including the trace evaluation) is $\mathcal{O}(n + u_1 + v_1)$ single-qubit gates, $\mathcal{O}(n + u_1 + v_1)$ CNOT gates, and $\mathcal{O}(n + u_2 + v_2)$ CCNOT gates.

B. Algorithm II

Algorithm I allows for efficient simulation of the full density operator for many physical systems characterized by local interactions; however, it requires a doubling of the number of qubits and an overhead of an ancilla and controlled operations for evaluating observables. In particular, the circuit required for measurements is too deep for near-term hardware. We therefore introduce a second algorithm based on the variational ansatz used to obtain the nonequilibrium steady states of Markovian systems [35,48] that overcomes these limitations. The isomorphism maps a density operator as

$$\rho = \sum_{x\in I} p_x U|x\rangle\langle x|U^\dagger \to |\rho\rangle = \sum_{x\in I} p_x U|x\rangle \otimes \overline{U}|x\rangle, \quad (11)$$

where the $|x\rangle$ label the n-qubit computational basis states and I is a subset of all 2^n possible bit strings. In the rest of the paper the index set I is implied. We note that, although we are using an ansatz for this algorithm, any density operator can be represented in this form provided the index set I is large enough. However, it should be noted that assuming polynomial resources to store the bit-string weights implies that the present algorithm employs a sparse approximation to represent the density matrix.

The Lindblad master equation is mapped identically to the vectorization mapping, resulting in Eq. (2). The propagator is again Trotterized and each term can be applied term by term. The unitary part of the propagator preserves the ansatz, as

$$\exp\left[(-i\mathbb{I}\otimes H + iH^\top\otimes\mathbb{I})\tau\right]\sum_x p_x U|x\rangle\otimes\overline{U}|x\rangle$$
$$= \sum_x p_x e^{iH^\top\tau}U|x\rangle\otimes e^{-iH\tau}\overline{U}|x\rangle \quad (12)$$

and $\overline{e^{-iH\tau}} = e^{iH^\top\tau}$ for Hermitian H. The remaining terms in the Trotterized propagator are of the form $\exp(-L_k^\top\overline{L_k}\tau/2)\otimes\exp(-L_k^\dagger L_k\tau/2)$ and $\exp(\overline{L_k}\otimes L_k\tau)$. The first term preserves the ansatz but is nonunitary, while the second term does not preserve the ansatz and is nonunitary. Considering the first of nonunitary terms, as in the original QITE algorithm, we seek a set of numbers q_x and a Hermitian operator A such that

$$V_k\sum_x p_x U|x\rangle\otimes\overline{U}|x\rangle = \sum_x (p_x + q_x)\exp(iA)U|x\rangle$$
$$\otimes\exp(-i\overline{A})\overline{U}|x\rangle + \mathcal{O}(\tau^2), \quad (13)$$

where $V_k = \exp(-\tau L_k^\top\overline{L_k}/2)\otimes\exp(-\tau L_k^\dagger L_k/2)$. As shown in Sec. I of the Supplemental Material [49], we find that

$$q_x = -\tau p_x\mathrm{Re}[\langle x|U^\dagger L_k^\top\overline{L_k}U|x\rangle]. \quad (14)$$

Decomposing A into a weighted sum of Pauli strings, $A = \sum_j a_j\sigma_j$, we find that the coefficients a_j satisfy the linear system $Sa = b$, with

$$S_{ij} = \sum_x p_x^2\mathrm{Re}[\langle x|U^\dagger(\sigma_i\sigma_j + \sigma_j\sigma_i)U|x\rangle]$$
$$- 2\sum_{xy} p_x p_y\mathrm{Re}[\langle x|U^\dagger\sigma_i U|y\rangle\langle x|U^\dagger\sigma_j U|y\rangle], \quad (15)$$

$$b_i = -\tau\left(\sum_x p_x^2\mathrm{Im}[\langle x|U^\dagger\sigma_i L_k^\top\overline{L_k}U|x\rangle]\right.$$
$$\left. + \sum_{xy} p_x p_y\mathrm{Im}[\langle x|U^\dagger\sigma_i U|y\rangle\langle y|U^\dagger L_k^\top\overline{L_k}U|x\rangle]\right). \quad (16)$$

The elements q_x, S, and b for the second nonunitary term, $\exp(\overline{L_k}\otimes L_k\tau)$, take a similar form. The derivation for both terms is given in Sec. I of the Supplemental Material [49]. We note that the total probability $\sum_x p_x = 1$ is conserved by the algorithm since $\sum_x q_x = 0$ at each time step as shown in Eq. (S19) of the Supplemental Material [49].

With this ansatz, observables are computed as $\langle O \rangle = \sum_x p_x \langle x|U^\dagger O U|x \rangle$, requiring the propagation of each $|x\rangle$ in parallel while storing the p_x. The final observable is computed as a classical average over all the propagated states and p_x. It is important to note that, although the ansatz lies in a dilated Hilbert space, all measurements take place on the original system and no entangling operations between the system and ancilla qubits are needed, and so no ancilla qubits are needed. In particular, for each time step, measurements on the original Hilbert space are used to determine the Hermitian matrix A. Expectation values of observables on this state are computed using the standard methods [1,2].

The benefits of Algorithm II are that it requires no ancilla qubits, and no Hadamard test is required for measurements of observables. These characteristics, trading quantum for classical resources and simulating large quantum circuits using smaller quantum computers are important for near-term hardware [50–54]. In particular, Algorithm II allows for halving the number of required qubits as in Ref. [54], allowing simulation of larger physical systems by increasing the classical and quantum computational time while decreasing the required number of qubits. Its drawbacks are the sparse representation of the density matrix and the number of measurements required to evolve the system. We discuss this overhead in the following section.

III. RUNTIME BOUNDS, COMPUTATIONAL OVERHEADS, AND ERRORS

In this section, we discuss the runtimes, quantum and classical computational overheads, and errors associated with each algorithm. Other sources of errors, such as those associated with noisy hardware, are not addressed here as they are nonalgorithmic errors.

A. Runtime bounds

We first bound the runtime of Algorithm I. For each time step in the Trotterization, the algorithm requires applying the imaginary–time propagator $\exp(-H_2\tau)$, where $\tau = t/N$ and N is the number of Trotter steps for the time evolution. Assuming a local Lindblad equation, H_2 is a sum of m_2 local terms h_l such that $H_2 = \sum_{l=1}^{m_2} h_l$, where m_2 scales polynomially with system size. The imaginary–time evolution operator $\exp(-H_2\tau)$ is implemented by additional Trotterization. For a given desired error ϵ_2, we Trotterize the imaginary–time evolution into p_2 steps. From Eq. (3.8) of Ref. [55], we find that, for $p_2 > 1/\epsilon_2$, the error in the p_2-step approximation is bounded by ϵ_2, assuming the number of Trotterization steps for time evolution N is sufficiently large that $3m_2tv_2/N < 1$, where $v_2 = \max_l\{\|h_l\|\}$.

Each term in the Trotterization is an imaginary–time increment and so corresponds to a rotation by a unitary operator supported on D qubits where D is the domain

size. An arbitrary D qubit unitary can be decomposed exactly into $\mathcal{O}(D^2 4^D)$ single-qubit and CNOT gates [45]. The total contribution to the running time from all the imaginary–time evolutions is $\mathcal{O}(Nm_2D^2 4^D/\epsilon_2)$.

Algorithm I also has additional unitaries $\exp(-iH_1\tau)$ interleaved between each QITE step, leading to an additional overhead. Because H_1 is a sum of local terms, $H_1 = \sum_{l=1}^{m_1} h_l$, $\exp(-iH_1\tau)$ needs to be Trotterized as well. Performing a similar analysis for the real-time evolution, we find the total running time to be

$$T = \mathcal{O}(Nm_1k^2 4^k/\epsilon_1 + Nm_2D^2 4^D/\epsilon_2), \qquad (17)$$

where ϵ_1 is the allowable Trotter error for the real-time evolution and k is the maximum number of qubits acted on by each term in the Hamiltonian. In the first term on the right-hand side, we have assumed that each k-local unitary can be exactly decomposed into $\mathcal{O}(k^2 4^k)$ single-qubit and CNOT gates [45].

A similar analysis can be carried out for Algorithm II, resulting in the same runtime up to constant factors with the following difference. The errors appearing in the runtime bound for Algorithm II do not include errors incurred from approximating the density operator with a strict subset of all bit strings. Although, in principle, any density operator can be represented by the sum $\sum_x p_x U|x\rangle \otimes \overline{U}|x\rangle$, this sum contains exponentially many terms and so only a subset of all possible bit strings can be included efficiently. Exclusion of bit strings leads to an error in representing the state given by $\sum_{x \in I^c} p_x$, where I is an index set containing all bit strings to be included, and I^c is its complement. Whether a polynomial scaling of the number of included bit strings is sufficient for a desired error will be problem dependent. In practice, this scaling can be assessed by stochastically sampling the bit strings until the simulation converges.

B. Measurement and classical computational overheads

Provided that the finite domain approximation holds, the largest computational overhead (apart from the running time) of both algorithms is the measurement overhead. For Algorithm I, this measurement overhead is the same as in the original QITE algorithm. State tomography over each domain consisting of D qubits needs to be carried out to construct the unitaries over that domain, requiring $\mathcal{O}(4^D)$ measurements. Assuming a one-dimensional lattice, there are $\mathcal{O}(n/D)$ domains, and so the total measurement overhead is $\mathcal{O}[(n/D)4^D]$ per time step. Similar bounds can be obtained for lattices in higher dimensions.

Algorithm II requires measurement of the matrix elements $\langle x|U^\dagger \sigma_i U|y \rangle$ for all Pauli strings σ_i supported on a domain D (measured in qubits) and all bit strings in $x, y \in I$ for some subset I of the 2^n n-bit strings. Measuring all matrix elements necessitates running $\mathcal{O}(L4^D|I|^2)$ circuits

TABLE I.   Asymptotic number of circuits required per time step per Lindblad operator for both algorithms for an open system on n sites. Here, D is the domain size and I is a subset of all n-bit strings for which the corresponding matrix elements are measured.

Algorithm	Number of qubits	Circuits per Lindblad operator
I	$2n + 1$	$(n/D)4^D$
II	n	$(n/D)4^D\|I\|^2$

per time step, where L is the number of Lindblad operators on the domain and $|I|$ is the number of bit strings included in the computation. For the algorithm to be efficient, the number of bit strings included in I must scale polynomially or slower with system size.

The finite-domain approximation required from QITE is accurate in many cases because the domain size D can generally be taken to be smaller for dissipative systems compared to the same system with no dissipation, as dissipation generally reduces a system's correlation length [56]. It should be noted that a reduced correlation length that decreases the cost for quantum algorithms might also permit an efficient classical description of the quantum evolution. This imprecise boundary is a consideration for quantum simulation algorithms generally and remains a topic of active investigation.

Table I summarizes the asymptotic scaling of the number of circuits required per time step of both algorithms for open quantum system dynamical simulation on n sites.

IV. RESULTS

We demonstrate both algorithms on IBM Quantum hardware for two cases: the spontaneous emission of a two-level system (TLS) in a heat bath at zero temperature, and the dissipative transverse field Ising model (TFIM) on two sites. The TLS ($n = 1$ from Table I) requires three physical qubits and one physical qubit to simulate with Algorithms I and II, respectively. The TFIM ($n = 2$ from Table I) requires five and two physical qubits, respectively.

Considering Algorithm I, neither the TLS nor the two-site dissipative TFIM on five qubits have constant depth circuit decompositions; Trotterizing both the real- and imaginary–time propagators results in a circuit with depth linear in the number of time steps. The resulting circuits are too deep for near-term devices. To overcome this limitation, we recompile the circuits as in Ref. [39]. In all simulations, we correct for readout error using the built-in noise models in Qiskit [57–60]. All measurements reported represent the average of 8192 shots and are repeated three times. Sampling noise in the measurement of the expectation value of the Pauli strings can lead to numerical instabilities in the QITE linear system. Therefore, when

constructing the QITE matrix for Algorithm I, regularizers 1×10^{-6} and 0.01, for the TLS and TFIM, respectively, are added to the diagonal terms of the S matrix to increase the condition number of the matrix S following the procedure in Ref. [38]. No regularizers are used for Algorithm II.

We first present results for the TLS model with the Hamiltonian

$$H = -\frac{\delta}{2}\sigma_z - \frac{\Omega}{2}\sigma_x \qquad (18)$$

and the Lindblad operator $\sqrt{\gamma}\sigma_-$, where σ_- is the lowering operator, δ is the detuning, Ω is the Rabi frequency, and γ is the spontaneous emission rate. We consider here the overdamped case where γ is of the order of the other energies in the system. It is found via numerical simulations that, to accurately capture the dynamics, only the Pauli strings in the set $\{\sigma_x \otimes \sigma_z, \sigma_y \otimes \sigma_x, \sigma_y \otimes \sigma_z, \sigma_z \otimes \sigma_x\}$ need to be included in the QITE unitary.

We set $\delta = \Omega = \gamma = 1$, and the initial state is chosen to be the excited state. In Fig. 2(a), we show the populations of the ground and excited states, with the experimental data averaged from three runs on IBM's *ibmq_mumbai* [63] for Algorithm I and *ibmq_casablanca* [63] for Algorithm II. Good qualitative agreement is obtained for all observables, with the deviation between the theoretical and experimental curves largely due to gate errors, as confirmed by numerical simulations and noisy hardware emulations.

We observe an initial exponential decay in the population of the excited state due to spontaneous emission into the bath followed by an approach to the nonequilibrium steady state (NESS) for $\gamma t \gg 1$. Damped Rabi oscillations are visible between these two regimes. The populations in the NESS can be interpreted as a balance between the spontaneous emission due to coupling to the bath and the absorption and stimulated emission due to the Hamiltonian driving term σ_x [64]. In the NESS, the combined spontaneous and stimulated emission rates are equal to the absorption rate.

In the absence of driving by an external electric field ($\Omega = 0$) the Hamiltonian is diagonal in the computational basis, resulting in the off-diagonal matrix elements $\rho_{01} = \overline{\rho_{10}}$ approaching zero as the system thermalizes. Figure 2(b) shows that these matrix elements remain nonzero as the NESS is approached, indicating that the hardware correctly obtains the expected quantum coherence as measured in the canonical basis. Also shown in Fig. 2(b) is the purity $\text{Tr}(\rho^2)$, which does not correspond to a time-independent Hermitian observable on the system but can nonetheless be obtained from the density operator representation on the hardware. Time evolution preserves the inner product $\text{Tr}(\rho^2) = \langle\rho|\rho\rangle$ on the quantum simulator, but the physical quantity, the normalized purity, $\text{Tr}(\rho^2)/\text{Tr}(\rho)$, is not constant.
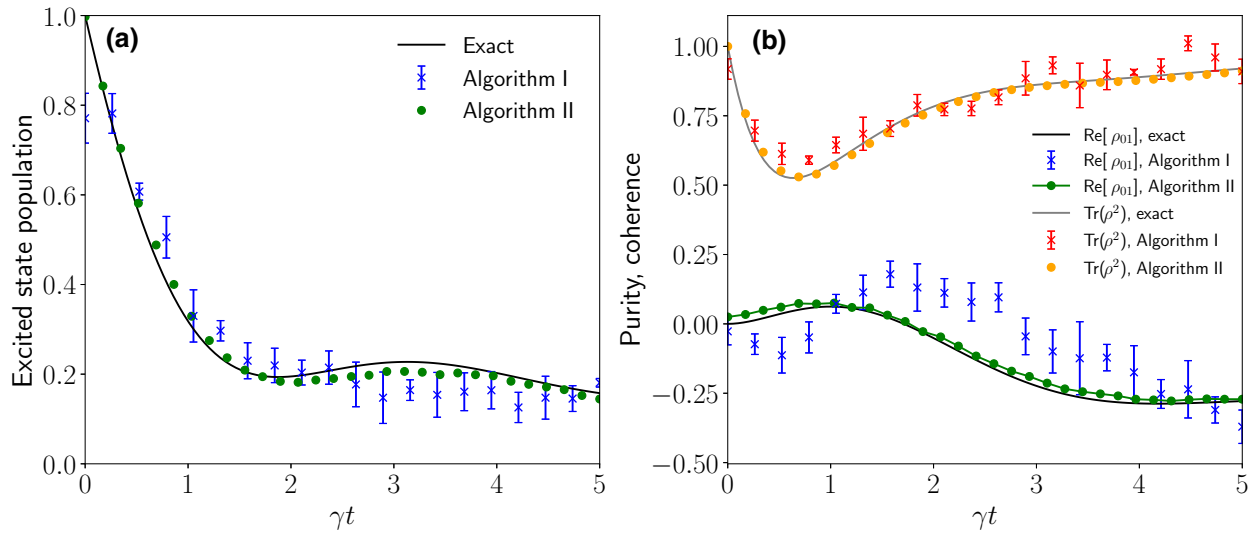
FIG. 2. (a) Population of the excited state from numerical simulations obtained with QuTiP [61,62] (black line), hardware using Algorithm I on *ibmq_mumbai* [63] (blue crosses), and Algorithm II (green circles) on *ibmq_casablanca* [63]. The deviation between the theoretical and experimental curves is largely due to gate error. The system approaches a nonequilibrium steady state for $\gamma t \gtrsim 5$. (b) Purity, $\mathrm{Tr}(\rho^2)$ (gray line), and the off-diagonal term, $\mathrm{Re}[\rho_{10}]$ (black line), corresponding to nondiagonal observables obtained in QuTiP [61,62]. Hardware results are shown for Algorithm I (purity, red crosses; $\mathrm{Re}[\rho_{10}]$, blue crosses) and for Algorithm II (purity, orange circles; $\mathrm{Re}[\rho_{10}]$, green circles). Hardware results for the observable $\mathrm{Im}[\rho_{10}]$ agree with the exact solution similarly to $\mathrm{Re}[\rho_{10}]$ but are omitted for clarity. For all hardware results for Algorithm I, the error bars are the standard deviation from three runs. The error bars for Algorithm II are smaller than the symbol size.

The larger deviation between the hardware results and the exact results for Algorithm I is attributed to the fact that Algorithm I is a three-qubit circuit requiring two-qubit gates, which are generally of lower fidelity than single-qubit gates. Since Algorithm I for the TLS is a single-qubit circuit, there are no infidelity contributions from two-qubit gates. In addition, the circuits required for Algorithm I are deeper than for Algorithm II, resulting in more gate errors. An additional breakdown of the error contributions due to hardware error and algorithmic error is provided in Fig. S1 within the Supplemental Material [49], in which we compare the hardware results to noiseless numerical emulations.

We next present experimental and numerical results on the two-site TFIM. The TFIM has the Hamiltonian

$$H = -J \sum_k \sigma_z^{(k)} \sigma_z^{(k+1)} - h \sum_k \sigma_x^{(k)} \qquad (19)$$

and Lindblad operators $\sqrt{\gamma}\sigma_-^{(k)}$, with nearest-neighbor coupling J, transverse magnetic field h, and decay rate γ. For this model, the number of required Pauli strings cannot be reduced by symmetry in Algorithm I. To reduce circuit depth, 16 Pauli strings are randomly selected out of the 256 possible Pauli strings on four qubits to implement the QITE unitary. We choose 16 Pauli strings as a balance between too few Pauli strings, which results in a poor approximation to normalized imaginary–time

evolution, and too many Pauli strings, which results in a large computational overhead and an ill-conditioned QITE matrix. Increasing the number of Pauli strings does not qualitatively increase the accuracy, as shown in Fig. S2 within the Supplemental Material [49].

Figure 3 shows the average magnetization of the dissipative TFIM with the initial state given by both spins in the spin-up state and $J = h = 1$ and $\gamma = 0.1$. Oscillations in magnetization are evident due to the relatively large transverse field. We observe qualitative agreement between theoretical and experimental curves from Algorithm I with a Trotter step $\gamma t / N \sim 0.5$. For the small system size of two sites, all 4 bit strings on two qubits are included in Algorithm II. Experimental results for Algorithm II are also in good qualitative agreement with the exact curve for all times.

In Sec. III, we discussed the runtime and resources required by both algorithms in a general setting. We now discuss the relative computational cost required by each algorithm for the specific case of the two-site TFIM hardware simulations. For the simulations considered here, Algorithm I is able to accurately describe the dissipative dynamics when using 16 out of the total of 256 Pauli strings. Simulations of Algorithm I using up to 48 Pauli strings, shown in Fig. S2 within the Supplemental Material [49], show no significant increase in accuracy when using more than 16 Pauli strings. In general, the number of required Pauli strings will be problem dependent.
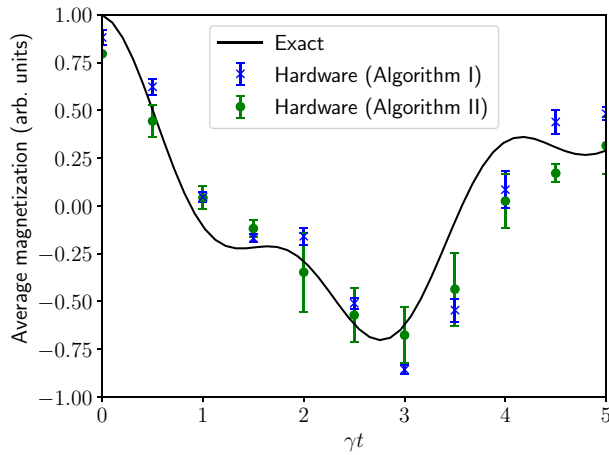
FIG. 3. Average magnetization $n^{-1} \sum_i \langle Z_i \rangle$ for the dissipative transverse field Ising model on two sites (five physical qubits for Algorithm I, two physical qubits for Algorithm II) using IBM Quantum's *ibmq_guadalupe* [63] for Algorithm I (blue symbols) and *ibmq_casablanca* [63] for Algorithm II (green symbols). Numerical solutions obtained in QuTiP are shown with black lines. The error bars for both algorithms are the standard deviation from three hardware runs. Both algorithms qualitatively agree with the exact dynamics for all simulated times. The deviation between the hardware results and the exact result for Algorithm II is due mainly to gate error.

Algorithm II requires measuring the matrix elements of all two-qubit Pauli strings at each time step, requiring 836 circuits per time step, versus only measuring expectation values of 16 operators in the case of Algorithm I, which requires 16 circuits. These measurements are only needed on a domain of fixed size.

For larger dissipation rates $\gamma \sim J, h$, separate numerical simulations, presented in Fig. S3 within the Supplemental Material [49], show that both algorithms are able to accurately capture the magnetization dynamics. However, these simulations do not include the error incurred by including a subset of bit strings in Algorithm II. The actual algorithmic error of Algorithm II will thus depend on the accuracy of the representation of the density matrix with a subset of bit strings for the given problem. Stochastic sampling of bit strings may be a viable approach for larger systems.

V. SUMMARY

We have introduced digital quantum algorithms for the time evolution of open quantum systems described by a Lindblad equation based on quantum imaginary–time evolution. Algorithm I uses QITE to implement the nonunitary evolution introduced when the density operator is vectorized, whereas Algorithm II uses an adaptation of QITE to maintain a purification-based ansatz throughout the computation. Calculations for the spontaneous emission of a two-level system and the dissipative transverse field Ising

model were carried out on IBM Quantum's quantum processors. Good qualitative agreement with the exact result was observed in all cases. These algorithms decrease the quantum resources required to simulate open quantum systems governed by Lindblad master equations on quantum hardware.

[1] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms (2020), ArXiv:2012.09265.

[2] I. M. Georgescu, S. Ashhab, and F. Nori, Quantum simulation, Rev. Mod. Phys. **86**, 153 (2014).

[3] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, and *et al.*, Noisy intermediate-scale quantum (nisq) algorithms, (2021), arXiv preprint ArXiv:2101.08448.

[4] B. Fauseweh and J.-X. Zhu, Digital quantum simulation of non-equilibrium quantum many-body systems, (2020), ArXiv:2009.07375.

[5] A. Smith, M. Kim, F. Pollmann, and J. Knolle, Simulating quantum many-body dynamics on a current digital quantum computer, npj Quantum Inf. **5**, 1 (2019).

[6] A. Chiesa, F. Tacchino, M. Grossi, P. Santini, I. Tavernelli, D. Gerace, and S. Carretta, Quantum hardware simulating four-dimensional inelastic neutron scattering, Nat. Phys. **15**, 455 (2019).

[7] H. Lamm and S. Lawrence, Simulation of Nonequilibrium Dynamics on a Quantum Computer, Phys. Rev. Lett. **121**, 170501 (2018).

[8] S. Endo, J. Sun, Y. Li, S. C. Benjamin, and X. Yuan, Variational Quantum Simulation of General Processes, Phys. Rev. Lett. **125**, 010501 (2020).

[9] C. Cirstoiu, Z. Holmes, J. Iosue, L. Cincio, P. J. Coles, and A. Sornborger, Variational fast forwarding for quantum simulation beyond the coherence time, npj Quantum Inf. **6**, 1 (2020).

[10] J. Gibbs, K. Gili, Z. Holmes, B. Commeau, A. Arrasmith, L. Cincio, P. J. Coles, and A. Sornborger, Long-time simulations with high fidelity on quantum hardware, (2021), arXiv preprint ArXiv:2102.04313.

[11] R. Barends, L. Lamata, J. Kelly, L. García-Álvarez, A. Fowler, A. Megrant, E. Jeffrey, T. White, D. Sank, J. Mutus, and *et al.*, Digital quantum simulation of fermionic models with a superconducting circuit, Nat. Commun. **6**, 1 (2015).

[12] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, A. Bengtsson, S. Boixo, M. Broughton, and B. B. Buckley, *et al.*, Observation of separated dynamics of charge and spin in the fermi-hubbard model, (2020), ArXiv:2010.07965.

[13] A. Macridin, P. Spentzouris, J. Amundson, and R. Harnik, Digital quantum computation of fermion-boson interacting systems, Phys. Rev. A **98**, 042312 (2018).

[14] S. P. Jordan, K. S. Lee, and J. Preskill, Quantum algorithms for quantum field theories, Science **336**, 1130 (2012).

[15] D. E. Kharzeev and Y. Kikuchi, Real-time chiral dynamics from a digital quantum simulation, Phys. Rev. Res. **2**, 023342 (2020).

[16] M. Kreshchuk, W. M. Kirby, G. Goldstein, H. Beauchemin, and P. J. Love, Quantum simulation of quantum field theory in the light-front formulation, (2020), ArXiv:2002.04016.

[17] D. A. Lidar, Lecture notes on the theory of open quantum systems, (2019), ArXiv:1902.00967.

[18] H.-P. Breuer and F. Petruccione, *The Theory of Open Quantum Systems* (Oxford University Press, New York, 2002).

[19] C. H. Tseng, S. Somaroo, Y. Sharf, E. Knill, R. Laflamme, T. F. Havel, and D. G. Cory, Quantum simulation with natural decoherence, Phys. Rev. A **62**, 032309 (2000).

[20] H. Wang, S. Ashhab, and F. Nori, Quantum algorithm for simulating the dynamics of an open quantum system, Phys. Rev. A **83**, 062317 (2011).

[21] H.-Y. Su and Y. Li, Quantum algorithm for the simulation of open-system dynamics and thermalization, Phys. Rev. A **101**, 012328 (2020).

[22] M. Cattaneo, G. De Chiara, S. Maniscalco, R. Zambrini, and G. L. Giorgi, Collision Models can Efficiently Simulate any Multipartite Markovian Quantum Dynamics, Phys. Rev. Lett. **126**, 130403 (2021).

[23] D. Bacon, A. M. Childs, I. L. Chuang, J. Kempe, D. W. Leung, and X. Zhou, Universal simulation of markovian quantum dynamics, Phys. Rev. A **64**, 062302 (2001).

[24] R. Sweke, I. Sinayskiy, D. Bernard, and F. Petruccione, Universal simulation of markovian open quantum systems, Phys. Rev. A **91**, 062308 (2015).

[25] M. Kliesch, T. Barthel, C. Gogolin, M. Kastoryano, and J. Eisert, Dissipative Quantum Church-Turing Theorem, Phys. Rev. Lett. **107**, 120501 (2011).

[26] S.-J. Wei, D. Ruan, and G.-L. Long, Duality quantum algorithm efficiently simulates open quantum systems, Sci. Rep. **6**, 30727 (2016).

[27] Z. Hu, R. Xia, and S. Kais, A quantum algorithm for evolving open quantum dynamics on quantum computing devices, Sci. Rep. **10**, 3301 (2020).

[28] J. Hubisz, B. Sambasivam, and J. Unmuth-Yockey, Quantum algorithms for open lattice field theory, (2020), ArXiv:2012.05257.

[29] Z. Hu, K. Head-Marsden, D. A. Mazziotti, P. Narang, and S. Kais, A general quantum algorithm for open quantum dynamics demonstrated with the fenna-matthews-olson complex, (2021), arXiv preprint ArXiv:2101.05287.

[30] K. Head-Marsden, S. Krastanov, D. A. Mazziotti, and P. Narang, Capturing non-markovian dynamics on near-term quantum computers, Phys. Rev. Res. **3**, 013182 (2021).

[31] L. Del Re, B. Rost, A. F. Kemper, and J. K. Freericks, Driven-dissipative quantum mechanics on a lattice: Simulating a fermionic reservoir on a quantum computer, Phys. Rev. B **102**, 125112 (2020).

[32] T. Haug and K. Bharti, Generalized quantum assisted simulator, (2020), arXiv preprint ArXiv:2011.14737.

[33] J. Han, W. Cai, L. Hu, X. Mu, Y. Ma, Y. Xu, W. Wang, H. Wang, Y. P. Song, C.-L. Zou, and L. Sun, Experimental Simulation of Open Quantum System Dynamics via Trotterization, Phys. Rev. Lett. **127**, 020504 (2021).

[34] E. Andersson, J. D. Cresser, and M. J. W. Hall, Finding the kraus decomposition from a master equation and vice versa, J. Mod. Opt. **54**, 1695 (2007).

[35] N. Yoshioka, Y. O. Nakagawa, K. Mitarai, and K. Fujii, Variational quantum algorithm for nonequilibrium steady states, Phys. Rev. Res. **2**, 043289 (2020).

[36] L. Bittel and M. Kliesch, Training Variational Quantum Algorithms is Np-Hard, Phys. Rev. Lett. **127**, 120502 (2021).

[37] S. McArdle, T. Jones, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, Variational ansatz-based quantum simulation of imaginary time evolution, npj Quantum Inf. **5**, 75 (2019).

[38] M. Motta, C. Sun, A. T. K. Tan, M. J. O'Rourke, E. Ye, A. J. Minnich, F. G. S. L. Brandão, and G. K.-L. Chan, Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution, Nat. Phys. **16**, 205 (2020).

[39] S.-N. Sun, M. Motta, R. N. Tazhigulov, A. T. Tan, G. K.-L. Chan, and A. J. Minnich, Quantum computation of finite-temperature static and dynamical properties of spin systems using quantum imaginary time evolution, PRX Quantum **2**, 010317 (2021).

[40] K. Yeter Aydeniz, G. Siopsis, and R. C. Pooser, Scattering in the ising model with the quantum lanczos algorithm, New J. Phys. **23**, 043033 (2021).

[41] N. Gomes, F. Zhang, N. F. Berthusen, C.-Z. Wang, K.-M. Ho, P. P. Orth, and Y. Yao, Efficient step-merged quantum imaginary time evolution algorithm for quantum chemistry, J. Chem. Theory Comput. **16**, 6256 (2020).

[42] K. Yeter-Aydeniz, R. C. Pooser, and G. Siopsis, Practical quantum computation of chemical and nuclear energy levels using quantum imaginary time evolution and lanczos algorithms, npj Quantum Inf. **6**, 63 (2020).

[43] G. Lindblad, On the generators of quantum dynamical semigroups, Comm. Math. Phys. **48**, 119 (1976).

[44] T. F. Havel, Robust procedures for converting among lindblad, kraus and matrix representations of quantum dynamical semigroups, J. Math. Phys. **44**, 534 (2003).

[45] M. A. Nielsen and I. Chuang, Quantum computation and quantum information (2002).

[46] S. Lloyd, Universal quantum simulators, Science **273**, 1073 (1996).

[47] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Simulating physical phenomena by quantum networks, Phys. Rev. A **65**, 042323 (2002).

[48] S. Endo, Z. Cai, S. C. Benjamin, and X. Yuan, Hybrid quantum-classical algorithms and quantum error mitigation, J. Phys. Soc. Jpn. **90**, 032001 (2021).

[49] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PRXQuantum.3.010320.

[50] S. Bravyi, G. Smith, and J. A. Smolin, Trading Classical and Quantum Computational Resources, Phys. Rev. X **6**, 021043 (2016).

[51] T. Peng, A. W. Harrow, M. Ozols, and X. Wu, Simulating Large Quantum Circuits on a Small Quantum Computer, Phys. Rev. Lett. **125**, 150504 (2020).

[52] T. Gujarati, A. Eddins, S. Bravyi, C. Hadfield, A. Mezzacapo, M. Motta, and S. Sheldon, Reducing circuit size in the variational quantum eigensolver–part 1: Theory, Bulletin of the American Physical Society (2021).

[53] A. Eddins, T. Gujarati, S. Bravyi, C. Hadfield, A. Mezzacapo, M. Motta, and S. Sheldon, Reducing circuit size in the variational quantum eigensolver–part 2: Experiment, Bulletin of the American Physical Society (2021).

[54] A. Eddins, T. Gujarati, S. Bravyi, C. Hadfield, A. Mezzacapo, M. Motta, and S. Sheldon, Doubling the size of quantum simulators by entanglement forging, (2021), ArXiv:2104.10220.

[55] M. Suzuki, Generalized trotter's formula and systematic approximants of exponential operators and inner derivations with applications to many-body problems, Comm. Math. Phys. **51**, 183 (1976).

[56] M. J. Kastoryano and J. Eisert, Rapid mixing implies exponential decay of correlations, J. Math. Phys. **54**, 102201 (2013).

[57] G. Aleksandrowicz, T. Alexander, P. Barkoutsos, L. Bello, Y. Ben-Haim, D. Bucher, F. Cabrera-Hernández, J. Carballo-Franquis, A. Chen, C. Chen, and *et al.*, Qiskit: An open-source framework for quantum computing, (2019), accessed: 2021-03-16.

[58] K. Temme, S. Bravyi, and J. M. Gambetta, Error Mitigation for Short-Depth Quantum Circuits, Phys. Rev. Lett. **119**, 180509 (2017).

[59] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, and J. M. Gambetta, Error mitigation extends the computational reach of a noisy quantum processor, Nature **567**, 491 (2019).

[60] S. Bravyi, S. Sheldon, A. Kandala, D. C. Mckay, and J. M. Gambetta, Mitigating measurement errors in multi-qubit experiments, (2020), ArXiv:2006.14044.

[61] J. Johansson, P. Nation, and F. Nori, Qutip 2: A python framework for the dynamics of open quantum systems, Comp. Phys. Commun. **184**, 1234 (2013).

[62] J. R. Johansson, P. D. Nation, and F. Nori, Qutip: An open-source python framework for the dynamics of open quantum systems, Comp. Phys. Commun. **183**, 1760 (2012).

[63] *ibmq_mumbai* v1.4.11, *ibmq_guadalupe* v1.2.14, and *ibmq_casablanca* v1.2.14, IBM Quantum Team, Retrieved from https://quantum-computing.ibm.com (Yorktown Heights, 2020), accessed: 2021-03-16.

[64] R. Loudon, *The Quantum Theory of Light* (OUP Oxford, 2000).