Preparation of excited states for nuclear dynamics on a quantum computer

Alessandro Roggero [1] Chenyi Gu [2] Alessandro Baroni,[3] and Thomas Papenbrock [2,4]

[1]*Institute for Nuclear Theory, University of Washington, Seattle, Washington 98195, USA*
[2]*Department of Physics and Astronomy, University of Tennessee, Knoxville, Tennessee 37996, USA*
[3]*Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico 87545, USA*
[4]*Physics Division, Oak Ridge National Laboratory, Oak Ridge, Tennessee 37831, USA*

We study two different methods to prepare excited states on a quantum computer, a key initial step to study nuclear dynamics within linear-response theory. The first method uses unitary evolution for a short time $T = \mathcal{O}(\sqrt{1-F})$ to approximate the action of an excitation operator $\hat{O}$ with fidelity F and success probability $P \approx 1 - F$. The second method probabilistically applies the excitation operator using the linear combination of unitaries (LCU) algorithm. We benchmark these techniques on emulated and real quantum devices, using a toy model for thermal neutron-proton capture. Despite its larger-memory footprint, the LCU-based method is efficient even on current generation noisy devices and can be implemented at a lower gate cost than a naive analysis would suggest. These findings show that quantum techniques designed to achieve good asymptotic scaling on fault-tolerant quantum devices might also provide practical benefits on devices with limited connectivity and gate fidelity.

I. INTRODUCTION

Quantum computing [1,2] holds a lot of promise for low-energy nuclear theory [3–8]. An error-corrected quantum computer with about 100 logical qubits, for instance, would make it possible to solve nuclear shell models [9] across the Segrè chart. Other examples of problems that are hard to solve with classical computers include time evolution of nuclear processes such as reactions, fission, and fusion. Present quantum devices consist of tens of noisy qubits, and decoherence is expected to remain a challenge for the foreseeable future. In this era of noisy intermediate-scale quantum (NISQ) devices [10], it is interesting to explore how to prepare, evolve, and solve for quantum states on quantum computers and how to mitigate or minimize the effects of decoherence [11,12].

In this paper we study how to prepare quantum states on existing superconducting devices. The nuclear physics problem we are motivated by is the $n(p, d)\gamma$ reaction, which is relevant for big-bang nucleosynthesis [13]. While this reaction is well understood and precise computations exist [14–16], it is a prototypical nuclear reaction and perhaps the simplest one to start with. How would one approach this problem on a quantum computer?

We note that quantum computers consists of interconnected qubits, i.e., two-level systems that can be thought of as spin-$1/2$ states. The Jordan-Wigner or Bravyi-Kitaev [17] transformations allow one to map fermionic creation and annihilation operators onto spin-lowering and -raising operators, respectively (see also Refs. [18–21] for other recent mappings). Having mapped a nuclear Hamiltonian onto qubits, quantum computing consists of two steps, namely, (i) the preparation of states via unitary operations, and (ii) the

readout via projection onto the canonical (spin-up, spin-down) basis. As in classical computing, we speak of "gates" and "circuits" when referring to individual unitary operations and a product of them. Any unitary operation between a set of qubits can be decomposed efficiently into products of unitary operations that involve only one or two qubits (see, for instance, Sec. 4.5 of Ref. [22]). On present superconducting devices, single-qubit operations have a relatively high fidelity of more than 99% while two-qubit gates are noisier with errors at the few percent level. Thus, it is an important task to reduce the number of two-qubit gates in quantum circuits, for instance, by keeping Hamiltonians and operators of interest as local as possible [23]. Finally, the readout is a quantum-mechanical measurement process and irreversible. Its outcome is stochastic, and expectation values result from many repeated experiments of state preparations followed by readouts. This measurement process is also plagued by noise, with readout fidelities larger than 95% on present superconducting devices. For an introduction to quantum computing, we refer the reader to Ref. [22].

The general problem we are interested in can be stated as follows: given a Hermitian excitation operator O and an initial state $|\Psi_0\rangle$ (for instance, the ground state of a many-body Hamiltonian) we want to find a protocol to prepare the normalized state

$$|\Phi_E\rangle = \frac{1}{\eta}O|\Psi_0\rangle \text{ with } \eta = \|O|\Psi_0\rangle\|, \quad (1)$$

on the same quantum register containing the original state. We note that the norm η in Eq. (1) above is the vector 2-norm, and $|\Phi_E\rangle$ is thus obtained from $|\Psi_0\rangle$ by a unitary operation. An efficient implementation of this procedure is important for

the performance of quantum algorithms developed to study exclusive scattering in linear response [7] could speedup the inclusion of pair-correlations with Jastrow factors [24], and more generally serve as a proxy to perform quantum computations with nonunitary gates [25].

This paper is organized as follows. In Sec. II we introduce the time evolution method and the LCU [26] to generate the excited state of Eq. (1). Section III presents the results for both methods, obtained from simulations and from running on quantum hardware. In Sec. IV we apply the methods to the $n(p, d)\gamma$ reaction. We present our conclusions in Sec. V. Several technical details can be found in the Appendixes.

II. STATE PREPARATION

In this section we present two procedures to perform the state preparation in Eq. (1). In both cases we start with a quantum register already initialized in the target quantum state $|\Psi_0\rangle$. For applications such as computing scattering cross sections, this state is the ground state of a many-body Hamiltonian that describes the nuclear target (but could also be a Gibbs state for applications at finite temperature). We note that the procedures we propose in this paper do not make use of any particular property of this state and can therefore be applied in general.

As with most numerical algorithms, both classical and quantum, the state-preparation techniques we propose are designed to be correct up to some finite error tolerance. Our goal is to find quantum procedures that produce an approximation $|\Phi_A\rangle$ to the target state $|\Phi_E\rangle$ of Eq. (1) with a guaranteed fidelity $F = |\langle\Phi_E|\Phi_A\rangle|^2$ and a success probability P_s. The efficiency of different algorithms is then related to the number of quantum operations needed to guarantee the desired "precision" metrics F and P_s.

When such algorithms are executed on NISQ devices without any form of error correction, the average fidelity of operations implemented on the quantum device provide effectively an upper bound f_U on the achievable state fidelity F. To give an example, let us assume that f_q is the average fidelity of a q-qubit gate operation. For a circuit consisting of n_1 1-qubit and n_2 2-qubit gates we thus estimate the maximum expected circuit fidelity as $f_U \equiv f_1^{n_1} f_2^{n_2}$. For practical purposes the target state fidelity fulfills then $F \approx f_U$. However, it is still desirable to devise schemes that are able to reach fidelities F arbitrarily close to one in an efficient way because they would be useful on a fully error-corrected machine.

To quantify fidelities and success probabilities, and to discuss the generation of excited states, we decompose the excitation operator O in terms of a linear combination of unitaries as

$$O = \sum_{k=0}^{L} \lambda_k U_k, \quad \lambda_k > 0. \tag{2}$$

It is also convenient for later use to introduce the 1-norm of the coefficient vector as

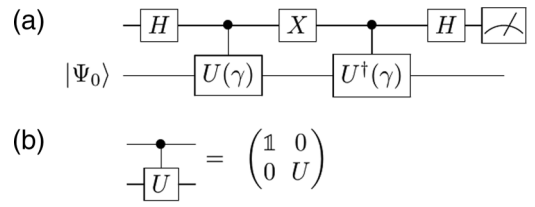$$\Lambda = \sum_{k=0}^{L} \lambda_k \geqslant \|O\|, \tag{3}$$

FIG. 1. Quantum circuits defined in the text: (a) state preparation circuit and (b) generic controlled unitary.

with $\|\cdot\|$ being the operator spectral norm. As we will see below, one can use the LCU algorithm [26] to prepare the excited state $|\Phi_E\rangle$ with success probability $P_s \propto 1/\Lambda^2$.

A. Time-dependent method

The first state preparation algorithm we consider was proposed in Ref. [7]. Here, we will also analyze its efficiency. The main idea is to exploit the time-evolution operator

$$U(\gamma) = \exp(-i\gamma O) = \cos(\gamma O) - i\sin(\gamma O), \tag{4}$$

associated with the excitation operator O to produce the approximate state

$$|\Psi_A(\gamma)\rangle \propto \sin(\gamma O)|\Psi_0\rangle = |\Phi_E\rangle + \mathcal{O}(\gamma^2). \tag{5}$$

Here $\gamma > 0$ is a "time" parameter. Using the unitary $U(\gamma)$, controlled by an auxiliary qubit or "ancilla," we perform this operation with the circuit shown in Fig. 1(a). Here and in what follows we use the convention that, if no initial state is specified on the left-hand side, then this defines the unitary operator corresponding to the circuit independently on the initial state. We also denote the Pauli matrices σ_x, σ_y, and σ_z compactly as X, Y, and Z. The gate denoted by H is the Hadamard gate, defined as

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \tag{6}$$

while the meter at the end of the ancilla line represents a projective measurement of the ancilla qubit polarization. Finally, the controlled-U operations are defined on the total Hilbert space, as shown in Fig. 1(b). Thus, they apply the unitary U to the qubit register containing the state $|\Psi_0\rangle$ when the ancilla qubit is in $|1\rangle$ and the identity operator $\mathbb{1}$ otherwise. It is convenient also to express the unitary operation depicted in the circuit from Fig. 1(a) explicitly in the relevant two-dimensional space spanned by the states $|0\rangle \otimes |\Psi_0\rangle$ and $|1\rangle \otimes |\Psi_0\rangle$ as

$$V(\gamma) = \begin{pmatrix} \cos(\gamma O) & i\sin(\gamma O) \\ -i\sin(\gamma O) & -\cos(\gamma O) \end{pmatrix}. \tag{7}$$

This is useful because the full evolution is contained in this subspace. Note that this operation differs slightly from the one proposed in Ref. [7] and it is simpler to implement.

Preparing the ancilla qubit in the state $|0\rangle$ and applying the circuit above [or equivalently the unitary matrix (7)] produces the final state

$$|\Omega(\gamma)\rangle = |0\rangle \otimes \cos(\gamma O)|\Psi_0\rangle - i|1\rangle \otimes \sin(\gamma O)|\Psi_0\rangle. \tag{8}$$

We can now use the projective measurement on the ancilla qubit displayed in the circuit in Fig. 1(a) to select only the second component of the state in Eq. (8). Thus, if the measurement finds the ancilla in the state $|1\rangle$ the system register will be left in the state

$$|\Psi_A(\gamma)\rangle = \frac{-i}{\sqrt{\langle\Psi_0|\sin^2(\gamma O)|\Psi_0\rangle}} \sin(\gamma O)|\Psi_0\rangle. \quad (9)$$

In the case that we measure the ancilla in $|0\rangle$ we restart the entire process from scratch and try again. Due to the presence of a measurement, this procedure is inherently stochastic and the success probability is

$$P_s = \langle\Psi_0|\sin^2(\gamma O)|\Psi_0\rangle = \mathcal{O}(\gamma^2). \quad (10)$$

It follows from Eqs. (9) and (10) that a small error in the state preparation requires a small time extent γ; however, small values of γ will also lead to a small success probability. This tension between our goals suggests that the time-evolution technique might be inefficient when a small infidelity $\Delta_f = 1 - F$ and a large success probability are sought. In what follows we further explore this tension and present an appropriate compromise.

Let us constrain the parameter γ to small values, as measured by the relevant operator norm Λ [see Eq. (3)], and use $\gamma \in [0, \pi/2\Lambda]$. With this constraint we derive the bounds for the success probability

$$\min[\sin^2(\gamma\Lambda), \gamma^2\eta^2] \geqslant P_s \geqslant \gamma^2\eta^2\left(1 - \frac{\gamma^2\Lambda^2}{3}\right), \quad (11)$$

and for the state fidelity,

$$\begin{aligned} F(\gamma) &= |\langle\Psi_E|\Psi_A(\gamma)\rangle|^2 \\ &= \frac{1}{\eta^2}\frac{|\langle\Psi_0|O\sin(\gamma O)|\Psi_0\rangle|^2}{\langle\Psi_0|\sin^2(\gamma O)|\Psi_0\rangle} \\ &\geqslant \left(1 - \frac{\gamma^2\Lambda^2}{6}\right). \end{aligned} \quad (12)$$

Details of the derivation are presented in Appendix A. Based on these bounds a minimum fidelity $F_{\min}$ for the state preparation results when the time extent γ fulfills

$$\gamma \leqslant \frac{\sqrt{6(1 - F_{\min})}}{\Lambda}. \quad (13)$$

This can be easily found by noticing that, if we want to ensure $F(\gamma) \geqslant F_{\min}$, it is sufficient to choose γ so that $(1 - \frac{\gamma^2\Lambda^2}{6}) \geqslant F_{\min}$. This yields a success probability bounded from above by

$$P_s \leqslant \sin^2(\sqrt{6(1 - F_{\min})}) \approx 6(1 - F_{\min}), \quad (14)$$

in the limit of small infidelities $1 - F_{\min} \ll 1$. Thus, the success probability P_s is proportional to the infidelity $\Delta_f = 1 - F_{\min}$. We note that this result does not depend on the norm η from Eq. (1). Similar bounds with the full η dependence can also be obtained directly from Eq. (11).

Especially important is the lower bound which guarantees a success probability of at least

$$P_s \geqslant \frac{\eta^2}{\Lambda^2}(2F_{\min} - 1) \quad (15)$$

for any target fidelity $F_{\min} \geqslant (1 - \pi^2/24) \approx 0.59$. This lower bound on $F_{\min}$ is based on $\gamma \leqslant \pi/2\Lambda$.

So far the estimates for the efficiency rely on the availability of an *exact* implementation of the time-evolution unitary $U(\gamma)$ of Eq. (4). In practice this is not a realistic assumption, and instead one has to make use of an approximate unitary operation $\widetilde{U}(\gamma)$ accurate only up to some additive error δ_U. Given the importance of such approximations we need to consider these next.

Imperfect time evolution

Let us assume that we only have an approximation to the time evolution operator $U(\gamma)$ and that the error is δ_U. Using this approximate operation in the full circuit from Fig. 1(a) generates a unitary operation $\widetilde{V}(\gamma)$. We have

$$\|\widetilde{V}(\gamma) - V(\gamma)\| \leqslant \delta_V, \quad (16)$$

and a total error $\delta_V \leqslant 2\delta_U$. In Eq. (16) we denoted the spectral norm of the operator by the double bars. The final state before the ancilla measurement is given by

$$\begin{aligned} |\widetilde{\Omega}(\gamma)\rangle &= \widetilde{V}(\gamma)|0\rangle \otimes |\Psi_0\rangle \\ &= \cos(\alpha)|\Omega(\gamma)\rangle + \sin(\alpha)|\xi(\gamma)\rangle. \end{aligned} \quad (17)$$

Here we introduced in the second line the normalized state $|\xi(\gamma)\rangle$ which is orthogonal to the error-free vector $|\Omega(\gamma)\rangle = V(\gamma)|0\rangle \otimes |\Psi_0\rangle$. Using the inequality (16) we find that the overlap with the wanted state is at least

$$\cos(\alpha) \geqslant 1 - \frac{\delta_V^2}{2}. \quad (18)$$

A more detailed derivation of this result is presented in Appendix B. The new approximate state can now be obtained by postselection on the ancilla being in $|1\rangle$. It reads

$$|1\rangle \otimes |\widetilde{\Psi}_A(\gamma)\rangle = \frac{1}{\sqrt{\widetilde{P}_s}}\Pi_1|\widetilde{\Omega}(\gamma)\rangle. \quad (19)$$

Here we have introduced the projector $\Pi_1 = |1\rangle\langle1| \otimes \mathbb{1}$, and the new success probability $\widetilde{P}_s$ satisfies

$$|\widetilde{P}_s - P_s| \leqslant |\sin(\alpha)|. \quad (20)$$

The angle α is defined in Eq. (17), and we again refer to Appendix B for the derivation. Using Eq. (18) we now find that the approximate success probability $\widetilde{P}_s$ is close to P_s, because

$$P_s - \delta_V\sqrt{1 - \frac{\delta_V^2}{4}} \leqslant \widetilde{P}_s \leqslant P_s + \delta_V\sqrt{1 - \frac{\delta_V^2}{4}}. \quad (21)$$

One can use the bounds on the success probability from Eq. (11) to find the corresponding bounds for $\widetilde{P}_s$. For the fidelity $\widetilde{F}$ we follow a similar approach and find

$$\begin{aligned} \widetilde{F} &\geqslant \frac{P_s F - \delta_V\sqrt{1 - \delta_V^2/4}}{\widetilde{P}_s} \\ &\geqslant \left(1 - \frac{\gamma^2\Lambda^2}{2}\right) - \left(2 - \frac{\gamma^2\Lambda^2}{3}\right)\frac{\delta_V}{\gamma^2\eta^2}. \end{aligned} \quad (22)$$
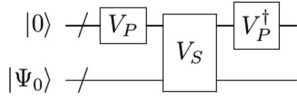
FIG. 2. Linear combination of unitaries (LCU) circuit diagram.

In the last line we used both Eqs. (11) and (21) to eliminate the explicit dependence on the success probabilities.

To guarantee a fidelity of at least $\widetilde{F}_{\min}$ we can now take

$$\gamma \leqslant \frac{\sqrt{1 - F_{\min}}}{\Lambda}, \tag{23}$$

for the time step. This ensures that the approximation error induced by the approximate time evolution is bounded by

$$\delta_V \leqslant \frac{\eta^2}{4} \gamma^4 \eta^2 \Lambda^2 \leqslant \frac{\eta^2}{2\Lambda^2} (1 - F_{\min})^2. \tag{24}$$

The error δ_U for the time-evolution unitary is

$$\delta_U(\gamma) \leqslant \frac{\eta^2}{8} \gamma^4 \eta^2 \Lambda^2 \leqslant \frac{\eta^2}{4\Lambda^2} (1 - F_{\min})^2. \tag{25}$$

This last inequality suggests that, even for small time intervals γ, higher-order Trotter-Suzuki decompositions are preferred thanks to their better error scaling. Alternatively, one should use error-efficient approximation schemes such as the Taylor expansion technique from Ref. [27] or quantum signal processing [28] that allow the computational cost to scale as $\mathcal{O}(\log(1/\delta))$.

B. Method based on linear combination of unitaries

The operation in Eq. (1) can be performed in an exact way with the LCU technique [26]. Here it is important that the decomposition in Eq. (2) of the excitation operator employs an efficient implementation of unitaries U_k and that the total number of terms, $L + 1$, grows as a low-order polynomial with increasing system size.

The procedure requires the implementation of two unitaries. The first one is the *prepare* unitary

$$V_P |0\rangle = \sum_{k=0}^{L} \sqrt{\frac{\lambda_k}{\Lambda}} |k\rangle, \tag{26}$$

acting on an ancilla register of size $M = \lceil \log_2(L + 1) \rceil$ initialized in the reference state $|0\rangle$. The second unitary operation is the *select* unitary

$$V_S = \sum_{k=0}^{L} |k\rangle\langle k| \otimes U_k. \tag{27}$$

Here the projectors $|k\rangle\langle k|$ act on the ancilla register. We follow convention and assume that $U_0 = \mathbb{1}$. This simplifies the execution of this unitary controlled by an ancilla qubit. We then execute the circuit displayed in Fig. 2 whose corresponding unitary we denote by W. The final state of this operation can be decomposed as (see, e.g., Refs. [26,29])

$$|\Omega\rangle = W|0\rangle \otimes |\Psi_0\rangle = \frac{1}{\Lambda} |0\rangle \otimes O|\Psi_0\rangle + |\Phi^\perp\rangle. \tag{28}$$

Here $|\Phi^\perp\rangle$ is orthogonal to the $|0\rangle$ state of the ancilla register

$$(|0\rangle\langle 0| \otimes \mathbb{1})|\Phi^\perp\rangle = 0. \tag{29}$$

Thus, we can isolate the first component of $|\Omega\rangle$ in Eq. (28) by measuring the ancilla register and discard the experiment if we do not measure $|0\rangle$. This postselection allows us to obtain the (not normalized) state

$$(|0\rangle\langle 0| \otimes \mathbb{1})|\Omega\rangle = \frac{1}{\Lambda} |0\rangle \otimes |\Psi_0\rangle = \frac{\eta}{\Lambda} |0\rangle \otimes |\Phi_E\rangle, \tag{30}$$

with a finite success probability

$$P_s^{\text{LCU}} = \frac{\eta^2}{\Lambda^2}. \tag{31}$$

From an operational standpoint we can use this strategy to compute expectation values over the excited state $|\Phi_E\rangle$ by first estimating empirically the success probability P_s^{LCU} and then consider the estimator

$$\langle \Phi_E | A | \Phi_E \rangle = \frac{\langle \Omega | \mathbb{1} \otimes A | \Omega \rangle}{P_s^{\text{LCU}}}. \tag{32}$$

Comparing the result presented in Eq. (31) with the bounds obtained for the time-dependent method in Eqs. (11) and (13) shows that the LCU-based method has a higher success probability (but not necessarily shorter depth) whenever the target infidelity satisfies $\Delta_f \leqslant 1/6$.

III. RESULTS

In this section, we analyze the performance of the two approaches by using a simple excitation operator. To quantify the quality of the results of the quantum computation and to assess the efficacy of our error mitigation procedure, we introduce two different metrics. Let us consider a set of N_O observables (e.g., the success probabilities for different excitation operators). We use $v_k^{(t)}$ to denote the expected theoretical value for the kth observable, and $v_k^{(e)}$ and $\varepsilon_k^{(e)}$, respectively, to denote its experimental value and estimated error, respectively. Throughout this paper we use two quality metrics. First, the chi squared

$$\chi^2 = \sum_{k=1}^{N_O} \frac{\left(v_k^{(e)} - v_k^{(t)}\right)^2}{\left(\varepsilon_k^{(e)}\right)^2} \tag{33}$$

quantifies if the data are compatible with expected results within errors. We regard the results which are closer to the value of one as better.

Second, the normalized sum of squared deviations

$$\text{nssd}(r) = \sqrt{\frac{\sum_{k=1}^{N_O} \left(v_k^{(e)} - v_k^{(t)}\right)^2}{\sum_{k=1}^{N_O} \left(r v_k^{(t)}\right)^2}}, \tag{34}$$

quantifies the accuracy of the calculation. The numerator is the 2-norm of the residuals while the denominator is the 2-norm of residuals corresponding to a relative error r. Unless stated otherwise, we use $r = 0.1$ and drop the argument, i.e., $\text{nssd} \equiv \text{nnsd}(0.1)$. We expect nssd to be close to zero. In other words, nssd $\ll 1$ indicates that the relative error is much

smaller than 10%. To summarize, the χ^2 is used to assess the degree of compatibility with the expected result, while the nnsd is used to quantify the accuracy.

A. Time-dependent method

To explore the efficiency of the first method we use $\theta \in [0, \pi]$ for the simple excitation operator

$$O(\theta) = \cos(\theta)X + \sin(\theta)\mathbb{1}, \qquad (35)$$

which acts on a single qubit. In this basis we identify $|\Psi_0\rangle = |0\rangle$ with the initial state of the reaction and $|1\rangle$ as the final state. The time evolution operator for this excitation operator is a rotation around the qubit's X axis and can be implemented exactly as

$$e^{-i\gamma O} = e^{-i\gamma \sin(\theta)} R_X[2\gamma \cos(\theta)], \qquad (36)$$

with R_X being a rotation around the X direction (see Appendix D for the explicit definition).

Based on this result, the full state preparation circuit from Fig. 1(a) can be implemented by using four CNOT gates, four Z rotations, and five additional single-qubit gates (see Appendix E 1 for a full derivation). We note that this CNOT count is suboptimal, because a generic 2-qubit unitary only requires three CNOT gates [30,31]. However, such a circuit comes at the price of needing 15 single-qubit rotations.

For the operator (35), the excited-state norm from Eq. (1) simply is $\eta = 1$ for every angle, and this will facilitate the analysis of the results. In addition we also have a simple bound for this operator's norm, because

$$\Lambda(\theta) = |\sin(\theta)| + |\cos(\theta)| \leqslant \sqrt{2} \equiv \Lambda_{\max}. \qquad (37)$$

In what follows we use the simple bound $\Lambda_{\max}$ instead of $\Lambda(\theta)$. This allows us to simplify the presentation because we can choose a θ-independent value for γ in Eq. (36). Using the bounds derived in the previous section, we can choose a sensible value for the "time" parameter γ as a function of the target maximum infidelity Δ_f. From Eq. (13) we find $\gamma \leqslant \sqrt{3\Delta_f}$, and the corresponding allowed values of γ are depicted in the lower panel of Fig. 3 as the gray region.

In the top part of Fig. 3 we show the corresponding bounds on the success probability P_s, see Eq. (11). The two upper bounds correspond to the arguments on the left-hand side of Eq. (11), i.e., the upper-bound A is $\sin^2(\gamma\Lambda)$ and the upper-bound B is $\gamma^2\eta^2$. The lower bound is from the right-hand side of Eq. (11). For our excitation operator (35), we have $\eta = 1$ and $\Lambda > 1$, and the upper bound A is tighter for most values of Δ_f. Based on these results, we set $\gamma = 0.3$ because this choice will guarantee a high fidelity $F(0.3) \geqslant 97\%$ while keeping the success probability (and therefore the effective sample size) sufficiently large. We have

$$9\% \geqslant P_s(0.3) \geqslant 8.46\%, \qquad (38)$$

and better bounds could have been obtained by using the tighter bound $\Lambda(\theta)$ instead of $\Lambda_{\max}$.
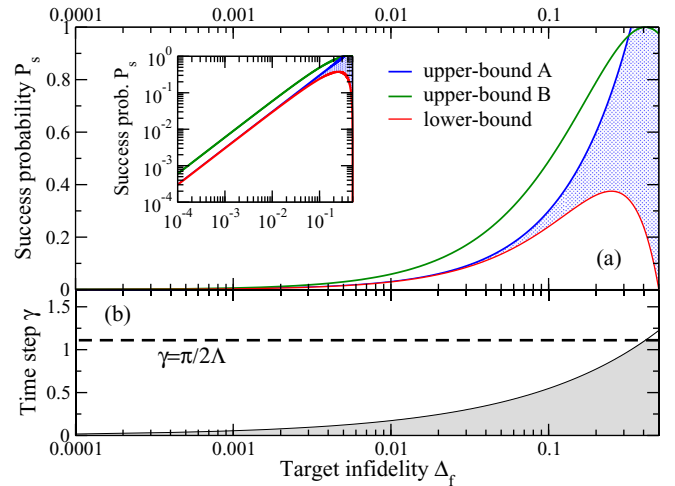

FIG. 3. Estimated bounds for both the success probability P_s (a) and the maximum time-step parameter γ (b) as a function of target infidelity Δ_f. The inset in the top panel shows more details for small values of Δ_f.

We used QISKIT [32] and implemented the relevant circuits on the IBM quantum device Vigo [33]. The layout of this quantum chip is shown in Fig. 4. Our quantum computation used qubits 1 and 2 of Vigo, and we employed both, the virtual machine (VM) and the quantum processor unit (QPU).

We present in Fig. 5 the results obtained with the VM and the QPU and compare them to the exact results from Eq. (10). Note that the VM simulations use a noise model that is initialized from calibration data coming directly from the device.

The bare results are directly collected from runs on the VM or QPU. The mitigated results are obtained after applying an error mitigation procedure described below. We perform two types of error mitigation: a readout correction on the measured distributions and a noise extrapolation assuming that the dominant noise channel is the one associated with the execution of entangling two-qubit gates like CNOT (see Refs. [5,34,35]). The detailed error mitigation scheme we use is explained in Appendix H. We see that the errors are strongly correlated, i.e., not random on the VM, and that error mitigation moves the bare results toward the benchmark. On the QPU, noise is significant, and error mitigation only somewhat improves the results.

We present in Table I the quality metrics for the results on the success probability shown in Fig. 5. Here, we listed the metrics for the bare result, readout (RO) error mitigated
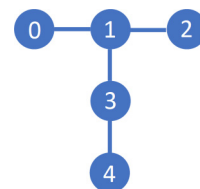

FIG. 4. Layout of the IBM quantum device Vigo [33]. Shown are the qubits, labeled from 0 to 4, and their connectivity.
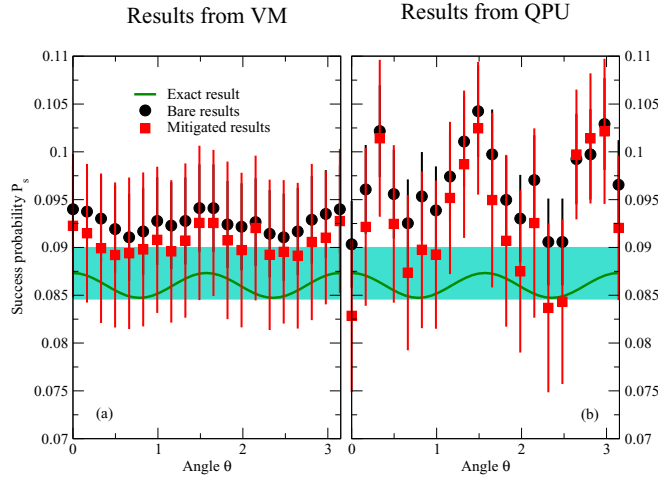
Results from VM   Results from QPU

FIG. 5. Results on the success probability P_s for the time-dependent method with the VM run (a) and QPU run (b) and exact analysis (green line) as a function of angle θ in excitation operator (35). Results are given with (red squares) and without (black squares) full mitigation. This convention applies to all the plots in this paper. The turquoise band is the bound Eq. (38).

results, and fully mitigated results (using readout correction and CNOT extrapolation). On average, the results for running on the QPU differ from the exact results by about 10% (as nssd $\approx$ 1), and apparently the error extrapolation technique is not successful in reducing this discrepancy. In contrast, we can see from the results in Table I that error mitigation improves the accuracy on the VM. The reason for this apparent discrepancy can be understood by noticing that the value of χ^2 for the QPU calculation is reduced by a factor of two as a result of the error extrapolation. This suggests that, even though the central values do not improve considerably, the dispersion of results is reduced by the error extrapolation step while keeping the results compatible with the reference values.

To assess the quality of the prepared state $|\Psi_A\rangle$ from Eq. (9), we directly measure its projection P_t onto the final state $|1\rangle$ and compare it with the expected result. We have for
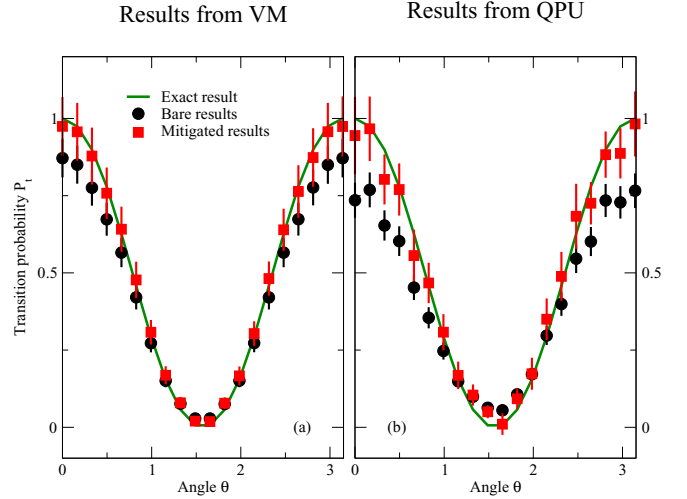
Results from VM   Results from QPU

FIG. 6. Results on the transition probability P_t for time-dependent method with the VM run (a) and QPU run (b) and exact analysis (green line). Results are given with (red squares) and without (black squares) full mitigation.

the transition probability

$$
\begin{aligned}
P_t &\equiv |\langle\Psi_A|1\rangle|^2 \\
&= \frac{|\langle 00|U(\gamma)|11\rangle|^2}{|\langle 00|U(\gamma)|11\rangle|^2 + |\langle 00|U(\gamma)|01\rangle|^2} \\
&= \frac{|\langle 00|U(\gamma)|11\rangle|^2}{P_s(\gamma)}.
\end{aligned} \tag{39}
$$

In the second line we re-introduced the ancilla state used in the implementation of the unitary $U(\gamma)$ from Eq. (7).

We show the results for P_t in Fig. 6. The left and right panels correspond to the results from the VM and QPU, respectively. The exact results are calculated from Eq. (39). The corresponding quality metrics are shown in Table II. On average the transition probability for running on a real QPU differs from the exact results only by 7%, and $\chi^2 = 0.65$ shows that P_t is compatible with the exact results within the uncertainties. Although not as complete as a full state

TABLE I. Quality metrics for the success probability P_s obtained (a) on the VM run and (b) from the QPU run. Shown are bare results, and with error correction for read out, and full error mitigation.

(a) VM		
	χ^2	nssd
Bare	2.08	0.765
RO mit.	1.46	0.656
Full mit.	0.37	0.539
(b) QPU		
	χ^2	nssd
Bare	5.64	1.299
RO mit.	3.00	1.004
Full mit.	1.47	1.052

TABLE II. Quality metrics for the transition probability obtained (a) on the VM run and (b) from the QPU run.

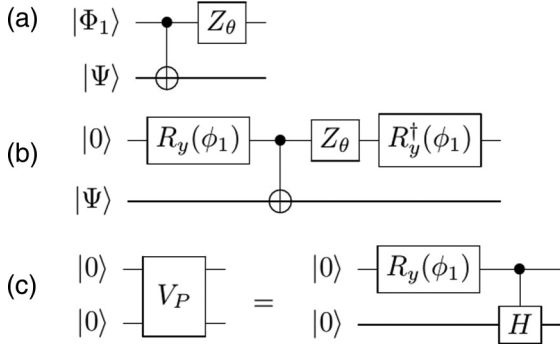| (a) VM		
	χ^2	nssd
Bare	2.80	1.259
RO mit.	0.87	0.757
Full mit.	0.39	0.386
(b) QPU		
	χ^2	nssd
Bare	11.27	2.324
RO mit.	3.19	1.059
Full mit.	0.65	0.718

FIG. 7. Explicit circuit construction for the unitaries defined in the text: (a) is the *select* unitary, (b) the full LCU circuit for the excitation in Eq. (35), and (c) the *prepare* unitary in second quantization, Eq. (46).

tomography, we can use these results as strong indicators that the time-dependent method runs with reasonable efficiency on a real quantum computer. We also note that the error mitigation works reasonably well because χ^2 and nssd are decreased.

B. Method based on linear combination of unitaries

We now discuss the results based on the state preparation using the LCU technique described in Sec. II B. The single-qubit excitation operator (35) can easily be implemented with this method. The excitation operator is a sum of two unitaries, and we only need $M = 1$ ancilla in addition to the qubit to represent the system. For every angle $\theta \in [0, \pi]$ the *prepare* unitary from Eq. (26) can be implemented by using a single rotation R_y around the Y axis (cf. Appendix D), and the resulting state is

$$|\Phi_1\rangle = R_y(\phi_1)|0\rangle = \cos\frac{\phi_1}{2}|0\rangle + \sin\frac{\phi_1}{2}|1\rangle. \quad (40)$$

Here, the angle is

$$\phi_1 = 2 \arcsin\left(\sqrt{\frac{|\cos(\theta)|}{|\cos(\theta)| + \sin(\theta)}}\right). \quad (41)$$

Given the state $|\Phi_1\rangle$, the *select* unitary from Eq. (27) can be implemented with the circuit displayed in Fig. 7(a), where Z_θ is the unitary whose matrix representation in the computational basis is

$$Z_\theta \equiv \begin{cases} \mathbb{1} & \text{for} \quad \theta \in [0, \pi/2] \\ Z & \text{for} \quad \theta \in [\pi/2, \pi] \end{cases} = \begin{pmatrix} 1 & 0 \\ 0 & 1 - 2H(\theta - \pi/2) \end{pmatrix}. \quad (42)$$

Here $H(x)$ is the Heaviside step function. This additional operation is necessary to account for the sign of the cosine for $\theta > \pi/2$. The complete circuit is presented in Fig. 7(b).

This circuit is too simple for a realistic test of the LCU method. For a more serious challenge, we implemented the excitation operator (35) in second quantization using two qubits for the system register. In this formulation, the initial state will be $|10\rangle$ while the final state is mapped to $|01\rangle$. The
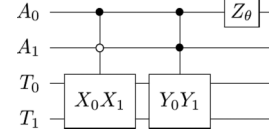

FIG. 8. Implementation of the *select* unitary for the excitation operator in Eq. (45).

excitation operator defined in Eq. (35) becomes

$$\widetilde{O}(\theta) = \sin(\theta)(c_1^\dagger c_1 + c_0^\dagger c_0) + \cos(\theta)(c_0^\dagger c_1 + c_1^\dagger c_0). \quad (43)$$

We use the Jordan-Wigner transformation to represent the creation and annihilation operators in terms of Pauli operators and define

$$c_k = \frac{X_k - iY_k}{2}, \quad c_k^\dagger = \frac{X_k + iY_k}{2}, \quad (44)$$

which act on qubit k. Thus, the excitation operator (43) becomes

$$\widetilde{O}(\theta) = \frac{\cos(\theta)}{2}(X_0 X_1 + Y_0 Y_1) + \sin(\theta)\mathbb{1}. \quad (45)$$

In this case we can use a *prepare* unitary very similar to the one above and producing the state

$$|\Phi_2\rangle = \cos(\phi_1)|00\rangle + \sin(\phi_1)|1\rangle \otimes \frac{|0\rangle + |1\rangle}{\sqrt{2}}, \quad (46)$$

using the implementation reported in Fig. 7(c), and an explicit decomposition of the needed two-qubit gate can be found in Appendix F 1. Note that the angle ϕ_1 entering Eq. (46) coincides with Eq. (41) thanks to the factor $1/2$ in front of the first two terms in Eq. (43).

Let us consider the possible schematic implementation of the *select* unitary V_s as presented in Fig. 8, where the labels (A_0, A_1) indicate the ancilla and labels (T_0, T_1) the target qubits, respectively.

A direct implementation of this circuit requires four Toffoli gates (see Appendix D for definition) gates and at most seven Clifford gates (among $X, Z, S, S^\dagger$). Using the optimal decomposition presented in Ref. [36] of six CNOT gates per Toffoli we arrive at a total of 24 CNOT gates. Constraints from the qubit connectivity on QPUs might further increase this count.

It is important to note, however, that, for the operator in Eq. (45), we are only using three of the four possible unitaries we could implement using two qubits. This implies that, when used in conjunction with the *prepare* operation, we would obtain the same final outcome if we used a more general unitary of the form

$$\widetilde{V}_S = V_S + \sum_{k=L+1}^{2^M - 1} |k\rangle\langle k| \otimes F_k, \quad (47)$$

with arbitrary operations F_k. In our case, for instance, we could perform any unitary operation controlled on $|01\rangle$ as the state $|\Phi_2\rangle$ is orthogonal to it. If done properly, a circuit implementation of $\widetilde{V}_S$ could be cheaper in CNOT cost than the original V_S. Furthermore one can try to cancel operations needed for $\widetilde{V}_S$ with those used for the prepare steps before and after.
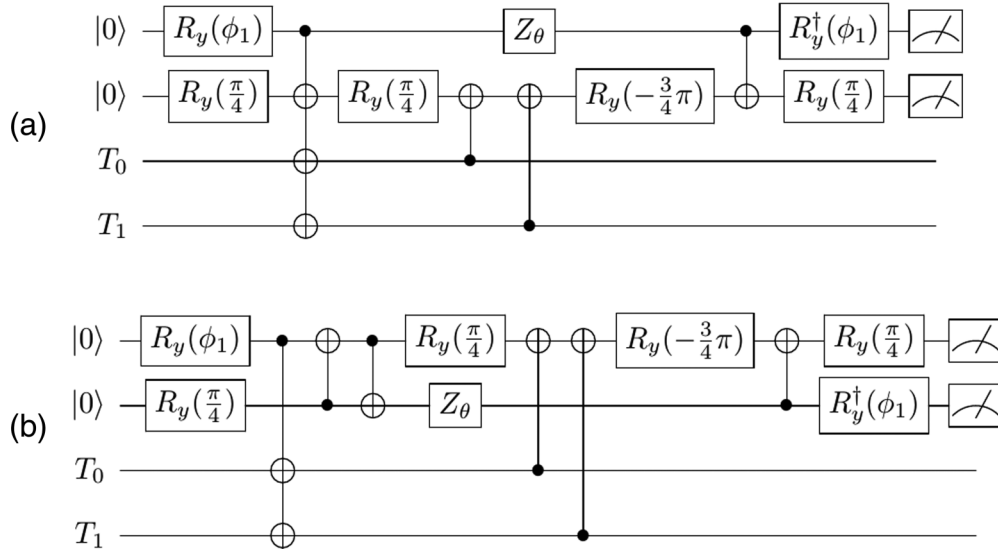
FIG. 9. Complete LCU circuits in second quantization: (a) assumes all-to-all connectivity while (b) is designed for implementation on the device displayed in Fig. 4.

Based on these insights we implemented the entire four-qubit circuit needed for the LCU with only six CNOT and up to seven single qubit gates, as displayed in Fig. 9(a). A complete derivation of this implementation can be found in Appendix F 1. We note that this construction is general because it does not rely on a particular input state for the target qubits.

Unfortunately the circuit in Fig. 9(a) requires the four qubits to be almost fully connected and cannot be implemented natively on the QPUs with the topology shown in Fig. 4. A simple modification that solves the problem with minimal additional cost is presented in Fig. 9(b), obtained by applying a SWAP gate (see Appendix D for definition) between the A_0 ancilla qubit and the A_1 ancilla qubit. This circuit requires just one CNOT more than the one in Fig. 9(a) and needs the A_0 qubit to have connections with the other three qubits. For the connectivity map shown in Fig. 4, this corresponds to qubit 1.

We define two estimators for the transition probability. First, since in this case we have $\eta = 1$, the success probability for the LCU method, Eq. (31) is completely determined by the operator norm $\Lambda(\theta)$, which for this operator takes the same form as Eq. (37). This gives the estimator

$$P_t^A(\theta) = \Lambda^2(\theta)\text{Tr}[|\Omega\rangle\langle\Omega|\Pi_0 \otimes |\psi_f\rangle\langle\psi_f|], \quad (48)$$

where the state $|\Omega\rangle \equiv |\Omega(\theta)\rangle$ is from Eq. (28), Π_0 is the projector to the state of the ancilla register with all qubits in $|0\rangle$, and $|\psi_f\rangle = |01\rangle$ is the final state. The trace operation is denoted as Tr.

An alternative estimator can be obtained by rescaling with the empirical success probability instead of the expected one,[1]

leading to the ratio estimator

$$P_t^B(\theta) \equiv \frac{\text{Tr}[|\Omega\rangle\langle\Omega|\Pi_0 \otimes |\psi_f\rangle\langle\psi_f|]}{\text{Tr}[|\Omega\rangle\langle\Omega|\Pi_0 \otimes \mathbb{1}]}. \quad (49)$$

In the absence of systematic errors, the two estimators P_t^A and P_t^B should coincide.

As for the time-dependent method above, we performed the quantum computations using QISKIT [32] and the IBM quantum device Vigo [33] using both a VM and the real QPU. In this case however, we mapped Vigo's qubits 1, 0, 3, and 2 onto the ancilla qubits A_0, A_1, and target qubits T_0, T_1 in Fig. 9(b), respectively.

In Fig. 10 we show the results for the success probability, and in Table III we report the values for the quality metrics in both VM and QPU runs. The comparison with Fig. 5 indicates
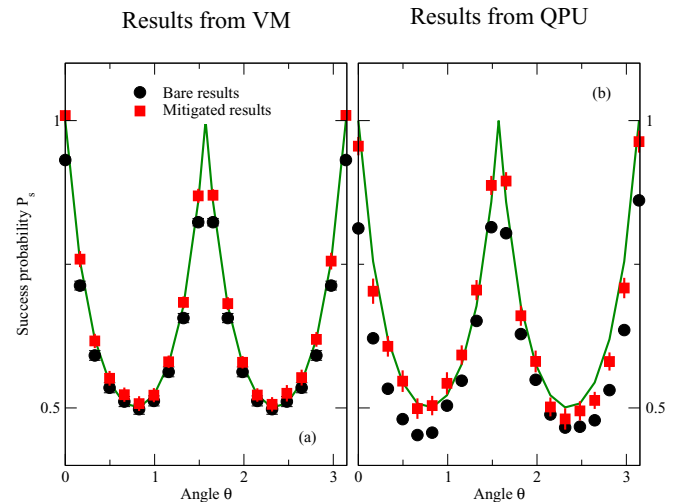

FIG. 10. Results on the success probability P_s for LCU method, as a function of angle θ in excitation operator (45). The symbols follow the same convention as in Fig. 5.

---

[1]Note that this corresponds to postselection on successful runs.

TABLE III. Quality metrics for the success probability from (a) the VM and (b) the QPU using the LCU method.

(a) VM		
	χ^2	nssd
Bare	26.52	0.453
RO mit.	3.63	0.211
Full mit.	0.44	0.118
(b) QPU		
	χ^2	nssd
Bare	112.56	1.190
RO mit.	70.17	1.085
Full mit.	2.61	0.415

TABLE IV. Quality metrics for the results on the transition probabilities P_t^A and P_t^B with (a) the VM run and (b) the QPU run for the LCU method.

(a) VM				
	$\chi^2(P_t^A)$	$\chi^2(P_t^B)$	nssd(P_t^A)	nssd(P_t^B)
Bare	76.20	10.09	1.022	0.623
RO mit.	6.57	0.29	0.321	0.106
Full mit.	0.29	0.12	0.129	0.053
(b) QPU				
	$\chi^2(P_t^A)$	$\chi^2(P_t^B)$	nssd(P_t^A)	nssd(P_t^B)
Bare	251.70	23.00	2.367	1.151
RO mit.	109.78	4.51	1.661	0.487
Full mit.	2.48	1.13	0.580	0.326

that, as anticipated in Sec. II, the LCU method has a much larger success probability than the time-evolution technique. Furthermore, the error mitigation works remarkably well and is able to bring the results close to their error-free value. This effect is especially visible in the evolution of the χ^2 measure, which in the real QPU case is reduced by almost two orders of magnitude by using error mitigation.

The corresponding results for the transition probability are presented in Fig. 11 using both estimator P_t^A from Eq. (48) and P_t^B from Eq. (49). The first striking difference between these estimators is the quality of the bare results as quantified by the χ^2 measure: for both the simple noise model implemented in the VM and the real noise present in the hardware we see an order of magnitude difference between P_t^A and P_t^B, with the latter being the better. This effect can be understood by qualitatively assuming the noise model to be a simple depolarizing channel (see, e.g., Sec. 8.3.4 of Ref. [22]), and we discuss the idea in detail in Appendix G.
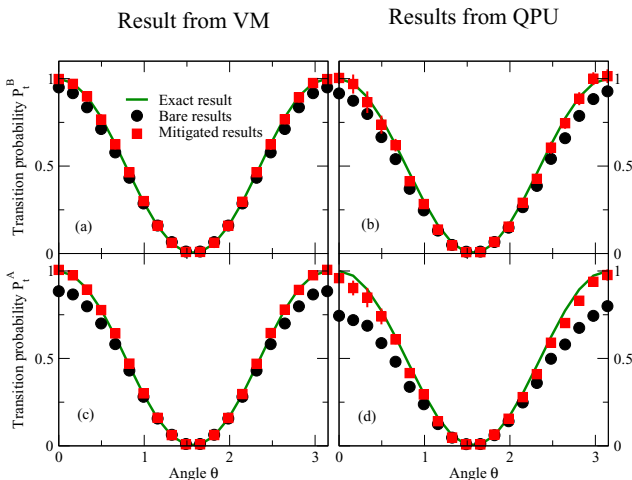

FIG. 11. Results for the transition probabilities P_t^A (c,d) and P_t^B (a,b) for the LCU method as a function of angle θ in excitation operator (45). The symbols follow the same convention as in Fig. 5.

Error mitigation works very well in this case too, and this is particularly evident for the lower two panels of Fig. 11 corresponding to the P_t^A estimator. The bare values for the QPU runs of the transition probability at both $\theta \approx 0$ and $\theta \approx \pi$ are off by about 25% but the fully mitigated results bring this down by a factor of five. As we can see from the results in Table IV, this behavior is also tracked by the average precision metric nssd and by the value of χ^2, which drops by two orders of magnitude for the P_t^A observable and down to values around unity for the second estimator.

To summarize this section, we used both the time-dependent and LCU methods to implement the excitation operator (35) and its generalization to second quantization (45). While both methods are clearly useful on existing quantum devices, the LCU-based method is more efficient and error tolerant, and also more accurate if we use estimators like Eq. (49). It is important to note at this point that, even though the success probability could be estimated off-line on a classical computer since it corresponds to a static structure factor, the more common situation is when we do not have access to this information (as in the next section) and we are instead forced to choose Eq. (49). It is encouraging to see that this is also the preferred way in general.

IV. APPLICATION TO THE $n(p, d)\gamma$ REACTION

In this section we apply the methods described in Sec. II and benchmarked in Sec. III to a simple model for the excitation operator that induces the $M1$ transition in the $n(p, d)\gamma$ reaction. In the two-dimensional space spanned by the $^1S_0$ and $^3S_1$ states the excitation operator takes the form

$$O(\theta) = \alpha\mathbb{1} + \beta X - \alpha Z = \begin{pmatrix} 0 & \beta \\ \beta & 2\alpha \end{pmatrix}, \qquad (50)$$

where the two real constants are given by

$$\alpha = \sin(\theta')\frac{g_p + g_n}{4}\mu_N, \qquad (51)$$

$$\beta = \frac{\mu_N}{2\sqrt{2}}(g_p - g_n)\cos(\theta'). \qquad (52)$$

Here, μ_N is the nuclear magneton and g_p and g_n denotes the proton and neutron g factor, respectively. The derivation of this expression is presented in Appendix C. In what follows, we set $\mu_N = 1$ because we can always scale the operator by multiplying by a constant.

In the following we initialize the system in the $^1S_0$ state (represented by $|0\rangle$ in the computational basis) and compute the transition probability to the $^3S_1$ ground-state (represented by $|1\rangle$ in the computational basis).

A. First-quantization version

As discussed in Sec. II A, the main obstacle to an efficient implementation of the time-dependent method is the low success probability for a high target fidelity. To find a reasonable value for the time interval γ we use the bounds derived in Eq. (11) and carry out a similar analysis to what was done in Sec. III A. The result of this analysis is that the value $\gamma = 0.3$ used in Sec. III A is also a good choice in this case, with a guaranteed high fidelity $F(0.3) \geqslant 98.8\%$ together with relatively high success probability

$$73.8\% \geqslant P_s(0.3) \geqslant 4.5\%. \tag{53}$$

The exact lower bound for P_s in this case is 6.8% and our lower bound [obtained from Eq. (11)] comes very close.

To implement the circuit shown in Fig. 1(a) we use a construction similar to the simpler case studied in Sec. III A which needs four CNOT gates and additional single qubit rotations. We present a detailed derivation of the construction in Appendix E 2.

For the LCU-based method instead, there is no hyperparameter to tune and the success probability is guaranteed to be the value computed from Eq. (31):

$$P_s = \frac{\langle 1|O^2(\theta)|1\rangle}{\Lambda^2} = \frac{\beta^2 + 4\alpha^2}{4\alpha^2 + \beta^2 + 4\alpha|\beta|} \geqslant 0.5. \tag{54}$$

The three operators appearing in the definition (50) of the excitation operator require an auxiliary register made of two ancillas. As described in detail in Appendix F 2, the LCU circuit can be implemented with only three CNOT gates and additional one-qubit rotations, even with the restricted connectivity shown in Fig. 4.

Both algorithms where executed on the Vigo QPU [33] using physical qubits (1,2) (ancilla in 1 and system in 2) for the time-dependent method and qubits (2,3,1) for the LCU-based method. In the latter case, qubits (2,3) represented ancilla qubits and qubit 1 was the system.

As we have seen from the results of the previous section, simulations using a virtual machine are not realistic enough to predict accurately the behavior of the real QPU, and for this reason we choose to only show results obtained from QPU runs in this section.

In Fig. 12 we report the results for both the success probability (left panel) and the $^1S_0 \rightarrow {}^3S_1$ transition probability (right panels). The red empty circles are results that were flagged by our error-mitigation scheme because no self-consistent extrapolation was found. The reported values are only mitigated with respect to the readout error (see Appendix H for more details on our framework). Note that this
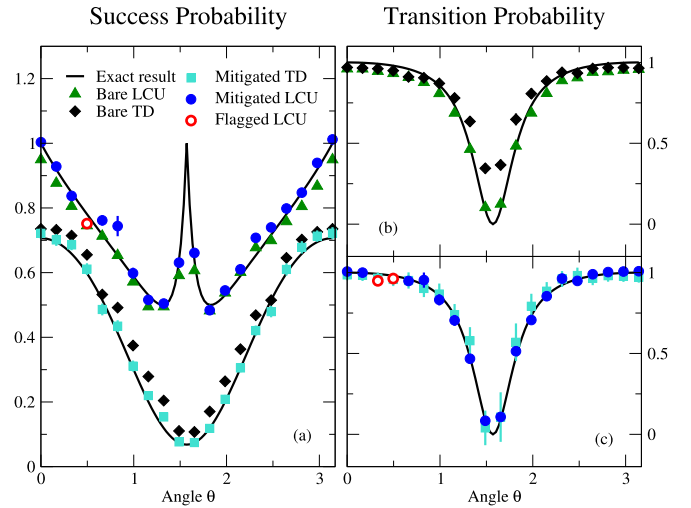

FIG. 12. Results on the success probability [panel (a) on the left] and transition probability (right panels) for the excitation operator in Eq. (50) using both the time-dependent method (TD) and the LCU-based method. Panel (b) shows the bare results from the QPU and panel (c) shows the results obtained after error mitigation.

is the only situation in this work where this problem was present; this could either be due to difficulties in executing our specific implementation or could simply be an effect coming from calibration of the device (the data for the time-dependent method was collected on July 20, 2020, while the LCU data were generated on August 27, 2020). As we can see from the plot, this flagged point does not show any discernible systematic error and it could also be possible that our flagging procedure is too conservative.

As we have anticipated with Eqs. (53) and (54), the success probability for the LCU method [top curve in Fig 12(a)] is larger than the one of the time-dependent method (indicated with TD in the plot). A direct effect of this is the reduced size of the statistical sample of experimental results that can be used to compute observables of the excited state like the transition probability. This is especially evident for the bare TD results for the transition probability [black diamonds in Fig. 12(b)] which are off by a factor of about three in the region of lowest success probability around $\theta = \pi/2$. This effect is also present in the mitigated results shown in Fig. 12(c) (the turquoise squares), which show large fluctuations in the same region. The results for the transition probability with the LCU technique, on the other hand, are more stable, with bare results already twice as accurate than with the TD method on average (see values of the nssd in Table V). Interestingly we can see from Table V that the readout correction for this observable is the dominant source of improvement over the bare results with the additional noise extrapolation step only changing the results slightly.

B. Second quantization

The second-quantization formalism is very useful in order to scale up simulations of nuclear dynamics since it guarantees a polynomial increase of the computational cost as

TABLE V. Quality metrics for the results on the $^1S_0 \to {}^3S_1$ transition probability in the $n(p, d)\gamma$ reaction with (a) the time-dependent (TD) method and (b) the LCU-based method. Both sets of results were obtained by using the Vigo QPU [33].

(a) TD		
	χ^2	nssd
Bare	11.77	1.198
RO mit.	6.04	1.013
Full mit.	0.31	0.436
(b) LCU		
	χ^2	nssd
Bare	13.62	0.494
RO mit.	1.41	0.173
Full mit.	2.19	0.253

a function of basis size. In this section we implement the transition operator (50) in the Fock space spanned by the two angular-momentum states using a register of $n = 2$ qubits. This excitation operator has only real matrix elements and can be expressed in second quantization as

$$\widetilde{O} = 2\alpha c_1^\dagger c_1 + \beta(c_0^\dagger c_1 + c_1^\dagger c_0). \tag{55}$$

Again we use the Jordan-Wigner transformation (44) to represent the creation and annihilation operators in terms of Pauli operators. This yields the excitation operator

$$\widetilde{O} = \alpha\mathbb{1} + \frac{\beta}{2}(X_0 X_1 + Y_0 Y_1) + \alpha Z_1$$

$$= \begin{pmatrix} 2\alpha & 0 & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & \beta & 2\alpha & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{56}$$

and we recover the original operator in the wanted subspace of one-particle states (the innermost 2×2 matrix).

Note that both operators in Eqs. (56) and (57) contain noncommuting operators and therefore the implementation of the time-dependent method will require a suitable approximation of the evolution operator $\exp(-i\gamma O)$ and, as shown in the previous section, will give a relatively low success probability. On the other hand the LCU method can be applied directly to either operator without making any approximation while maintaining a large success probability. For these reasons we study here only the extension of the LCU to the second-quantized case.

We could also obtain the same operator in this subspace by using an alternative definition:

$$\overline{O} = \alpha\mathbb{1} + \frac{\beta}{2}(X_0 X_1 + Y_0 Y_1) - \frac{\alpha}{2}(Z_0 - Z_1)$$

$$= \begin{pmatrix} \alpha & 0 & 0 & 0 \\ 0 & 0 & \beta & 0 \\ 0 & \beta & 2\alpha & 0 \\ 0 & 0 & 0 & \alpha \end{pmatrix}, \tag{57}$$
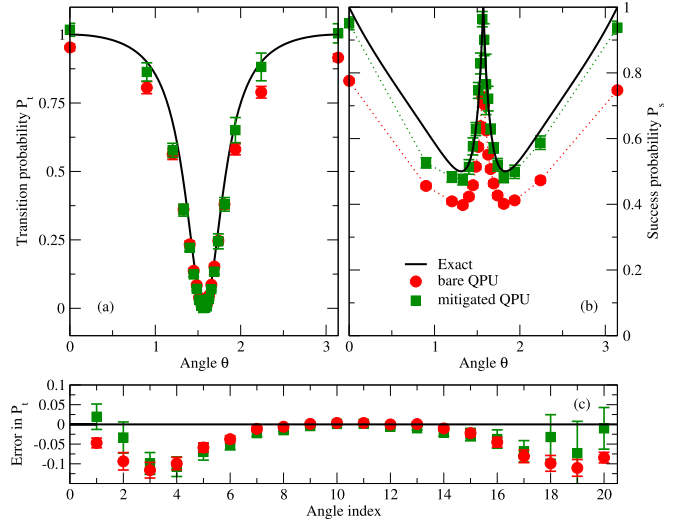

FIG. 13. Results for the (a) transition probability P_T and (b) success probability P_s for the $np \to d\gamma$ reaction obtained using the LCU method on the Vigo QPU [33]. In panel (c) we present, for every angle considered here, the deviation from the expected result. The dotted lines in panel (b) are just a guide to the eye.

which can be implemented with LCU with less depth. As we show in Appendix F 2, the *select* unitary for the operator in Eq. (57) can be implemented with only seven CNOT gates together with three for the *prepare* unitary, while the former requires 11 CNOT and two CNOT, respectively (see Ref. [31] for the prepare).

To increase the density of points in the region around $\pi/2$ where the cross section drops to zero, we chose to implement instead the simpler operator

$$\overline{O}_M(\theta) = \sin(\theta)\mathbb{1} + \frac{\cos(\theta)}{2}(X_0 X_1 + Y_0 Y_1)$$

$$- \frac{\sin(\theta)}{2}(Z_0 - Z_1), \tag{58}$$

for uniformly spaced values of the angle θ. The original excitation operator is then obtained as

$$\overline{O} = \sqrt{\alpha^2 + \beta^2}\, \overline{O}_M(\phi_{\alpha\beta}), \quad \phi_{\alpha\beta} = \arctan\left(\frac{\alpha}{\beta}\right). \tag{59}$$

Note that, since every observable we compute is independent on a global scale factor like $(\alpha^2 + \beta^2)^{1/2}$, we only need to perform the change of variables in the angle.

This simulation was executed on the Vigo QPU [33] using the circuit from Fig. 19(d). This requires 15 CNOT gates for the connectivity graph in Fig. 4. For this simulation, we used qubits 1, 4, 3 for the ancilla qubits A_0, A_1, and A_2 and qubits 0 and 2 for the target qubits T_0, T_1, respectively.

We present the results obtained from this simulation in Fig. 13. In Fig. 13(a) we report the result for the transition probability and in Fig. 13(b) the corresponding success probability for the LCU step. In all panels, black curves represented the expected exact values, red circles are bare result from the QPU and green squares are the fully mitigated results (with read-our correction and error extrapolation). Finally, in

TABLE VI. Quality metrics for (a) the results on the success probability and (b) transition probability for the excitation operator in Eq. (57) using the LCU method on the Vigo QPU [33].

(a) Success probability		
	χ^2	nssd
Bare	372.01	2.18
RO mit.	250.72	2.04
Full mit.	3.81	0.52
(b) Transition probability		
	χ^2	nssd
Bare	15.4	1.219
RO mit.	8.0	0.835
Full mit.	5.8	0.928

Fig. 13(c) at the bottom, we show the error in the transition probability for all the 20 angles we considered here (a uniform grid in $[0, \pi]$).

Note that, as mentioned above, the change of variables in Eq. (59) produces a larger density of results in the interesting central region. Once again we see that, for the LCU method, the quality of the results for the transition probability is higher than for the success probability thanks to cancellation of errors. For instance, we can see that, even though the bare result for the success probability at $\theta = \pi$ is off by more than 25%, the corresponding (bare) result for the transition probability has only an error of about 8%. This explains why the difference between bare and mitigated results [very apparent in Fig. 13(b)] is not seen directly in the errors shown in Fig. 13(c) at the bottom. Interestingly, even though error mitigation is able to bring some results to be compatible with the exact case, for some of the angles the mitigation procedure fails to improve the agreement. We interpret this as further evidence that the calculation of ratio estimators, like the transition probability, enjoys a much reduced sensitivity to hardware noise.

This behavior is in fact not directly observed in the results on the success probability presented in Table VI, where error extrapolation is responsible for the majority of the two orders of magnitude improvement in the value of the χ^2.

V. CONCLUSIONS

We applied the time-evolution method and the LCU to generate excited states on a quantum computer and studied their accuracy and efficiency. Both methods are feasible and efficient for the state preparation algorithm. The time-evolution method is accurate in the quantum computation of transition matrix elements. However, keeping a small target infidelity goes together with a relatively low success probability. In contrast, the LCU method has a much larger success probability while also being accurate in the quantum computation of the transition matrix element. It is also more resilient to depolarizing noise with the help of ratio estimators. This is true for the examples studied in this paper and also for most cases where it is easy to add ancilla qubits and postselect on

their measured state. However, there might still be some cases where the time-dependent method could be superior. It is an advantage that this method only needs one ancilla qubit.

Note that these considerations would change when considering fault-tolerant quantum devices that allow us to perform calculation of arbitrary length: the success probability P_s of a quantum algorithm can be increased to 1 by using $\mathcal{O}(1/\sqrt{P_s})$ additional unitary gates using standard amplitude amplification techniques. Alternatively, we could implement the excitation operator deterministically also using quantum signal processing [37] or its generalization, the quantum singular value transformation [38]. In the context of nuclear physics, similar ideas were used in past work for the evolution operator itself [8] and have the potential to substantially reduce the overall cost to carry out *ab initio* calculations of both inclusive and exclusive nuclear cross sections [39].

Our quantum computations employed both the quantum virtual machine and a quantum processor, and we used noise mitigation to reduce readout noise and nonperfect fidelities of the single-qubit and two-qubit operations. We also derived bounds for success probabilities of the excitation operators and developed metrics that quantify the accuracy of the studied methods.

Based on a very simple model, we also performed a quantum computation of the $n(p, d)\gamma$ reaction, using the time-dependent method and the LCU-based method. The applications of this work are a step towards computing nuclear reactions on quantum devices.

APPENDIX A: PROOF OF BOUNDS FOR TIME-DEPENDENT METHOD

Here we derive the bounds that were presented in Sec. II A of the main text. Using Taylor's theorem we know that, for some $y \in [0, x]$, the following holds:

$$\sin(x) = x - \frac{x^3}{6}\cos(y). \tag{A1}$$

Using this relation we arrive at the following bounds on the expectations values:

$$\langle X^2 \rangle - \frac{\langle X^4 \rangle}{6} \leqslant \langle X \sin(X) \rangle \leqslant \langle X^2 \rangle \tag{A2}$$

and

$$\langle X^2 \rangle - \frac{\langle X^4 \rangle}{3} + \frac{\langle X^6 \rangle}{36} \leqslant \langle \sin^2(X) \rangle \leqslant \langle X^2 \rangle. \quad \text{(A3)}$$

Here X is a bounded operator with $\|X\| \leqslant \pi/2$ and its averages $\langle X^k \rangle = Tr[X^k \rho]$ are computed on a generic input state ρ. The lower bounds can be relaxed into a form more amenable for estimation by noticing that

$$\langle X^4 \rangle = \text{Var}[X^2] + \langle X^2 \rangle^2 \leqslant \Lambda^2 \langle X^2 \rangle, \quad \text{(A4)}$$

where Λ is a bound on the norm $\|X\|$. The final inequalities used in the main text are then

$$\langle X^2 \rangle \left(1 - \frac{\Lambda^2}{6} \right) \leqslant \langle X \sin(X) \rangle \leqslant \langle X^2 \rangle, \quad \text{(A5)}$$

and

$$\langle X^2 \rangle \left(1 - \frac{\Lambda^2}{3} \right) \leqslant \langle \sin(X)^2 \rangle \leqslant \langle X \sin(X) \rangle, \quad \text{(A6)}$$

or the alternative upper bound

$$\langle X^2 \rangle \left(1 - \frac{\Lambda^2}{3} \right) \leqslant \langle \sin(X)^2 \rangle \leqslant \sin(\Lambda)^2, \quad \text{(A7)}$$

which is obtained by noticing that $\sin^2(x)$ is monotonically increasing for $x \in [0, \pi/2]$.

APPENDIX B: DETAILS ON BOUNDS WITH IMPERFECT TIME EVOLUTION

Here we derive the bounds that were used in Sec. II A of the main text. As a starting point we take two normalized states

$$|\phi\rangle = A|0\rangle, \quad |\psi\rangle = B|0\rangle, \quad \text{(B1)}$$

obtained using two unitaries A and B satisfying

$$\|A - B\| \leqslant \delta. \quad \text{(B2)}$$

Let us consider the (unnormalized) state obtained by taking their difference

$$|\varepsilon\rangle \equiv |\phi\rangle - |\psi\rangle = (A - B)|0\rangle. \quad \text{(B3)}$$

Its norm is bound as follows:

$$\begin{aligned}
\||\varepsilon\rangle\|^2 &= \langle 0|(A^\dagger - B^\dagger)(A - B)|0\rangle \\
&\leqslant \max_{\lambda \in \sigma[(A^\dagger - B^\dagger)(A - B)]} \lambda \\
&\equiv \max_{s_i \in \{\text{singular values of } M = (A-B)\}} s_i^2 \\
&\equiv \|A - B\|^2 \leqslant \delta^2, \quad \text{(B4)}
\end{aligned}$$

where in the second line $\sigma[M]$ denotes the spectrum of the operator M, in the third line we used the definition of singular values, and in the last line we used Eq. (B2). At this point we can decompose the second state as

$$|\psi\rangle = \cos(\theta)|\phi\rangle + \sin(\theta)|\xi\rangle, \quad \text{(B5)}$$

for some normalized state $|\xi\rangle$ orthogonal to $|\phi\rangle$. This yields

$$|\varepsilon\rangle = [1 - \cos(\theta)]|\phi\rangle - \sin(\theta)|\xi\rangle, \quad \text{(B6)}$$

so that we have

$$\begin{aligned}
\||\varepsilon\rangle\| &= [1 - \cos(\theta)]^2 + \sin^2(\theta) \\
&= 1 + \cos^2(\theta) - 2\cos(\theta) + 1 - \cos^2(\theta) \\
&= 2 - 2\cos(\theta) \leqslant \delta^2. \quad \text{(B7)}
\end{aligned}$$

In the last line we used the result from Eq. (B4).

Given a projection operator Π with $\Pi^2 = \Pi$, we have

$$\begin{aligned}
|\widetilde{P}_s - P_s| &:= |\langle\psi|\Pi|\psi\rangle - \langle\phi|\Pi|\phi\rangle| \\
&= |\text{Tr}(\Pi\rho) - \text{Tr}(\Pi\sigma)| \\
&= |\text{Tr}(\Pi(\rho - \sigma))| \\
&\leqslant D(\rho, \sigma) = \tfrac{1}{2}\text{Tr}|\rho - \sigma| = |\sin(\theta)|. \quad \text{(B8)}
\end{aligned}$$

Here we have defined for convenience the density matrices $\rho = |\psi\rangle\langle\psi|$ and $\sigma = |\phi\rangle\langle\phi|$, and in the last line we used the definition of the trace distance $D(\rho, \sigma)$.

APPENDIX C: $M1$ TRANSITION MATRIX ELEMENTS

Here we derive in more detail the form of the $M1$ transition operator used in Sec. IV of the main text. The dominant process for the $np \rightarrow d\gamma$ reaction is from the continuum $^1S_0$ state (with spin isospin $T = 1$, $T_z = 0$) to the deuteron bound state. For a calculation of this process in pionless EFT, we refer the reader to Refs. [14–16]. In leading order the bound-state is the $^3S_1$ state with isospin $T = T_z = 0$. Thus, the electromagnetic transition will be of $M1$ multipole order, and the magnetic moment

$$\mathbf{m} = 2\mu_N \mathbf{l} + g_p \mu_N \mathbf{S}_p + g_n \mu_N \mathbf{S}_n \quad \text{(C1)}$$

is the transition operator. Here, μ_N is the nuclear magneton, and g_p and g_n are the proton and neutron g factors, respectively. For the $^1S_0 \rightarrow {}^3S_1$ transition, the orbital angular momentum $\mathbf{l}$ does not contribute, and the spatial wave functions only contribute through the overlap between a continuum state $u_1(r) \approx (a - r)/a$ and a bound state $u_3(r) = e^{-\gamma r}$, with a and γ real parameters. Focusing on the spin-isospin part of the initial $|i\rangle$ and final $|f\rangle$ states we write

$$|i\rangle \equiv \tfrac{1}{2}(|p \uparrow n \downarrow\rangle + |n \uparrow p \downarrow\rangle - |p \downarrow n \uparrow\rangle - |n \downarrow p \uparrow\rangle),$$

$$|f\rangle \equiv \tfrac{1}{2}\sqrt{2}(|n \uparrow p \uparrow\rangle - |p \uparrow n \uparrow\rangle). \quad \text{(C2)}$$

Thus,

$$\langle f|\mathbf{m}|i\rangle = \frac{\mu_N}{\sqrt{2}}(g_p - g_n)\langle\uparrow |\mathbf{S}| \downarrow\rangle. \quad \text{(C3)}$$

Using $\mathbf{S} = \tfrac{1}{2}\sigma$ finally yields

$$\langle f|\mathbf{m}|i\rangle = \frac{\mu_N}{2\sqrt{2}}(g_p - g_n)\begin{pmatrix} 1 \\ -i \\ 0 \end{pmatrix}. \quad \text{(C4)}$$

Thus, $|\langle f|\mathbf{m}|i\rangle|^2 = \mu_N^2(g_p - g_n)^2/4$, and we have $g_p = 5.586$, and $g_n = -3.826$.

Two comments are in order. First, the spatial matrix element is a number that we cannot compute on a quantum chip using a minimum number of states. Our states are structureless, i.e., they contain only spin and isospin information. Second, we could also compute the $E2$ transition to the $^3D_1$

component of the deuteron wave function. This contribution is suppressed as the asymptotic D to S state amplitude is small.

More generally, we have also the following diagonal matrix elements:

$$\langle f|\mathbf{m}|f\rangle = \frac{g_p + g_n}{2}\mu_N \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \tag{C5}$$

and

$$\langle i|\mathbf{m}|i\rangle = 0. \tag{C6}$$

If we take the polarization vector ξ on the XZ plane we find

$$\langle f|\mathbf{m}\cdot\xi|f\rangle = \sin(\theta')\frac{g_p + g_n}{2}\mu_N, \tag{C7}$$

$$\langle i|\mathbf{m}\cdot\xi|i\rangle = 0, \tag{C8}$$

$$\langle f|\mathbf{m}\cdot\xi|i\rangle = \frac{\mu_N}{2\sqrt{2}}(g_p - g_n)\cos(\theta'). \tag{C9}$$

In this two-dimensional space we can then express the excitation operator as

$$O(\theta') = \alpha\mathbb{1} + \beta X + \gamma Z = \begin{pmatrix} \alpha + \gamma & \beta \\ \beta & \alpha - \gamma \end{pmatrix}, \tag{C10}$$

with

$$\alpha = \sin(\theta')\frac{g_p + g_n}{4}\mu_N, \tag{C11}$$

$$\beta = \frac{\mu_N}{2\sqrt{2}}(g_p - g_n)\cos(\theta'), \tag{C12}$$

$$\gamma = -\sin(\theta')\frac{g_p + g_n}{4}\mu_N. \tag{C13}$$

In particular we note that

$$\alpha = -\gamma \geqslant 0, \quad \beta \begin{cases} \geqslant 0 & \text{if } \theta' \in [0, \pi/2] \\ < 0 & \text{if } \theta' \in (\pi/2, \pi]. \end{cases} \tag{C14}$$

APPENDIX D: QUANTUM GATES

In this section we provide more details on the conventions used in the definitions of the quantum gates used in the main text. We indicate the single qubit Pauli matrices σ_x, σ_y, and σ_z and the corresponding gates as X, Y, and Z. The Hadamard gate, denoted with H, was introduced in Eq. (6) in the main text. The one-qubit rotation gates R_x, R_y, and R_z used throughout this work are defined in the following way:

$$R_x(\phi) = \begin{pmatrix} \cos(\phi/2) & -i\sin(\phi/2) \\ -i\sin(\phi/2) & \cos(\phi/2) \end{pmatrix}, \tag{D1}$$

$$R_y(\phi) = \begin{pmatrix} \cos(\phi/2) & -\sin(\phi/2) \\ \sin(\phi/2) & \cos(\phi/2) \end{pmatrix}, \tag{D2}$$

$$R_z(\phi) = \begin{pmatrix} e^{-i\phi/2} & 0 \\ 0 & e^{i\phi/2} \end{pmatrix}, \tag{D3}$$

where we note that only the rotation about the z axis above differs from the implementation on IBM QISKIT [32] by an
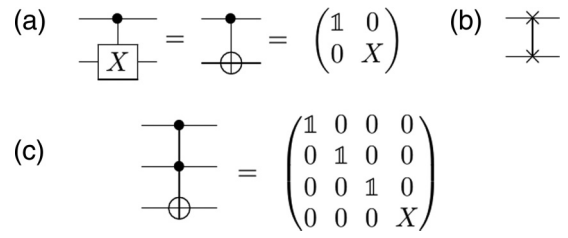

FIG. 14. Circuit representation of common unitaries (see text).

overall global phase $e^{i\phi/2}$. The two-qubit gate used more frequently in the main text is the CNOT gate defined in Fig. 14(a), with $\mathbb{1}$ being a 2×2 identity matrix. We also use the SWAP gate which is represented by the symbol in Fig. 14(b) and described by the following unitary matrix:

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{D4}$$

Finally, the 3-qubit Toffoli gate used in Fig. 8 of the main text is defined in Fig. 14(c) and compactly expressed as the following linear combination:

$$T = (\mathbb{1} - |11\rangle\langle 11|) \otimes \mathbb{1} + |11\rangle\langle 11| \otimes X. \tag{D5}$$

APPENDIX E: CIRCUIT IMPLEMENTATION DETAILS FOR THE TIME-DEPENDENT METHOD

Here we derive in detail the quantum circuits needed to implement the time-dependent state preparation described in Fig. 1(a) of the main text.

We start by discussing the simplest case when the excitation operator O is a single-qubit operator and express explicitly Fig. 1(a) in terms of one-qubit gates and CNOTs. Using the Euler decomposition for a unitary 2×2 matrix, we can express the exact propagator $U(\gamma) = \exp(-i\gamma O)$ using three angles and a phase parameter

$$e^{-i\gamma O} = e^{-i\delta}R_z(x_1)R_y(x_2)R_z(x_3). \tag{E1}$$

The time-dependent method requires an implementation of this unitary controlled by an ancilla qubit and it is clear from this decomposition that this can be implemented by using three controlled one-qubit rotation gate and one controlled phase gate. Here we choose the decomposition ZYZ for the Euler angles since the rotation gate R_z can usually be implemented with higher fidelity.

A controlled Z rotation can be written as in Fig. 15(a). A controlled Y rotation can be written similarly to Fig. 15(a) by replacing R_z rotations with R_y rotations. We can implement the controlled time-evolution operator by decomposing each controlled one-qubit rotation separately.

A more efficient approach is described in Ref. [40] (Lemmas 5.1 and 5.2) and requires only two CNOTs. Using this strategy, the implementation of the controlled time-evolution operator is reported in Fig. 15(b), where we have defined the
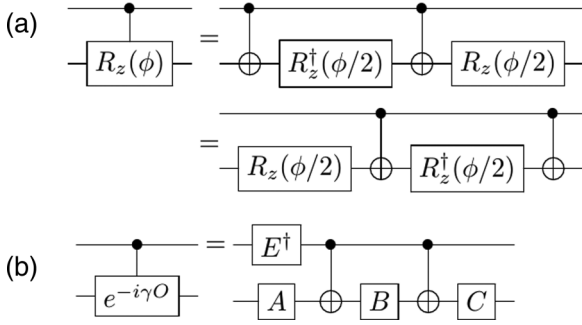
FIG. 15. Controlled single qubit rotations: (a) single z rotation, (b) general case.

following gates:

$$A = R_z\left(\frac{x_3 - x_1}{2}\right), \tag{E2}$$

$$B = R_y\left(-\frac{x_2}{2}\right)R_z\left(-\frac{x_1 + x_3}{2}\right), \tag{E3}$$

$$C = R_z(x_1)R_y\left(\frac{x_2}{2}\right), \tag{E4}$$

$$E = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\delta} \end{pmatrix}. \tag{E5}$$

Notice that the gate denoted by $E(\delta)$ is equivalent (up to an overall phase) to the $R_z(\delta)$ gate defined in Eq. (D3).

1. Simple excitation operator

Let us consider the simple model excitation

$$O(\theta) = \cos(\theta)X + \sin(\theta)\mathbb{1}. \tag{E6}$$

We first notice that the propagator $U(\gamma)$ is a rotation around the X axis

$$e^{-i\gamma O} = e^{-i\gamma \sin(\theta)}R_x[2\gamma \cos(\theta)], \tag{E7}$$

because the constant term contributes only to a global phase. We use the identity $R_x(\theta) = HR_z(\theta)H$, and the controlled time-evolution operator can be expressed as in Fig. 16(a) with $\delta = \gamma \sin\theta$ and $\alpha = \gamma \cos\theta$.

The final implementation of the complete state-preparation circuit with the excitation operator (E6) is given in Fig. 16(b).
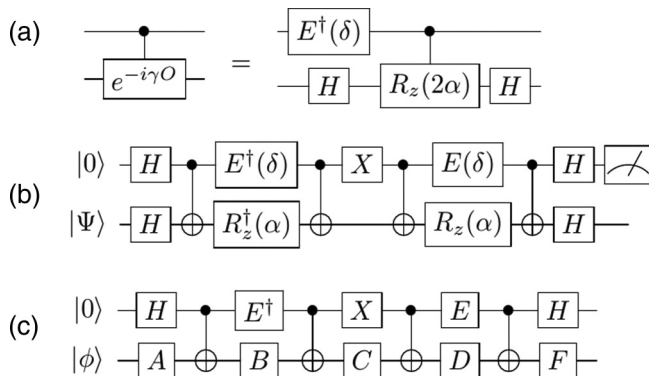

FIG. 16. Quantum circuits described in the text.

Here we used the two different variants in Fig. 15(a) to remove two rotations.

2. Nuclear excitation operator

Here we present the quantum circuit for the nuclear excitation operator from Eq. (C10):

$$O(\theta') = \alpha I + \beta X + \gamma Z. \tag{E8}$$

The rotation obtained by exponentiation of this operator can be again decomposed in Euler angles as above and the result for $U^\dagger(\gamma)$ reads

$$e^{i\gamma O} = e^{i\delta}R_z(-x_1)R_y(x_2)R_z(-x_3). \tag{E9}$$

Here the rotation angles are those for the original $U^\dagger(\gamma)$ as in Eq. (E1) instead. We then use Fig. 15(b) twice to get the final circuit as in Fig. 16(c), where

$$C = R_z\left(\frac{x_1 - x_3}{2}\right)R_z(x_1)R_y\left(\frac{x_2}{2}\right), \tag{E10}$$

$$D = R_y\left(-\frac{x_2}{2}\right)R_z\left(\frac{x_1 + x_3}{2}\right), \tag{E11}$$

$$F = R_z(-x_1)R_y\left(\frac{x_2}{2}\right), \tag{E12}$$

while gate A and B are the same as defined in Fig. 15(b).

APPENDIX F: CIRCUIT IMPLEMENTATION FOR THE LINEAR COMBINATION OF UNITARIES METHOD

We derive and present the final expressions of the circuits used with the LCU method in the main text.

1. Simple excitation operator

The simple excitation operator of interest is

$$O_2(\theta) = \frac{\cos(\theta)}{2}(X_0X_1 + Y_0Y_1) + \sin(\theta)\mathbb{1}. \tag{F1}$$

Here $\mathbb{1}$ denotes the 4×4 identity. We use the following mapping between ancillary qubit states and the operators on the left-hand side of Eq. (F1):

$$|00\rangle \rightarrow \mathbb{1}, \quad |10\rangle \rightarrow X_0X_1, \quad |11\rangle \rightarrow Y_0Y_1. \tag{F2}$$

The circuit corresponding to the *select* unitary is displayed in Fig. 8 of the main text, where A_0 and A_1 refer, respectively, to the first and second ancillary qubit while T_0 and T_1 denote the target qubits. We recall that the circuit above should be applied after the *prepare* unitary circuit from Fig. 7(c). To simplify the *select* circuit in Fig. 8, we use the identity

$$X_0Z_0X_1Z_1 = -Y_0Y_1. \tag{F3}$$

This leads to the equivalent *select* circuit displayed in Fig. 17(a). Here the last controlled-Z operation has been added to correct the sign of the Y_0Y_1 term.

We note that the second and third gates in Fig. 17(a) commute with each other. This implies the identity in Fig. 17(b). Furthermore, we are not using the state $|01\rangle$ of the ancilla register in the mapping (F2). Thus, we can directly remove the controls on the first ancillary qubit for the fifth and sixth
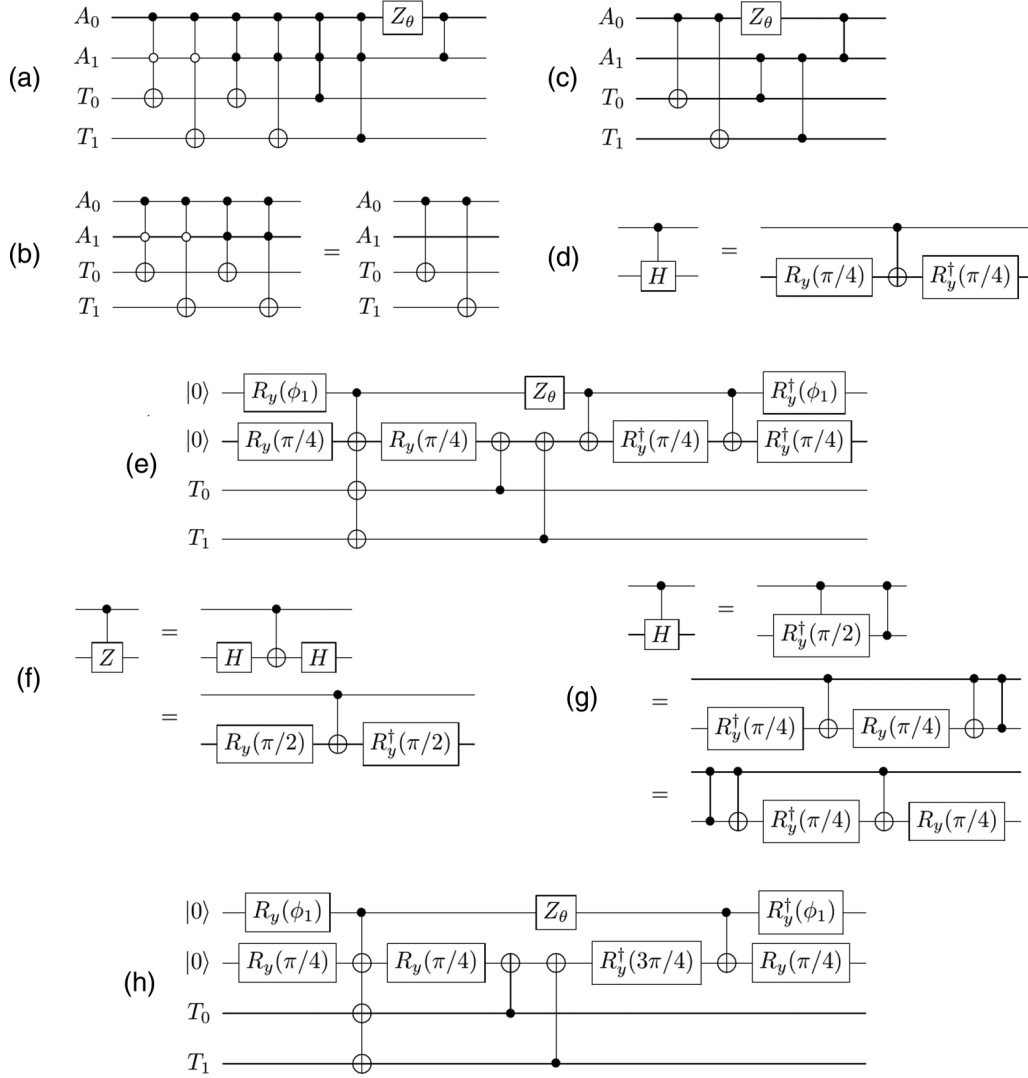
FIG. 17. Quantum circuits described in the text.

gate in Fig. 17(a). The final circuit for *select* becomes as in Fig. 17(c).

We now turn to discuss the implementation of the *prepare* unitary and consider the circuit presented in Fig. 7(c) of the main text. We start by using the decomposition

$$H = X R_y(\pi/2) = R_y^\dagger(\pi/4) X R_y(\pi/4), \qquad (F4)$$

to implement the controlled-H operation as described in Fig. 17(d). This only requires a single CNOT gate. Using these identities, the full circuit needed for LCU implementation of the operator in Eq. (F1) is reported in Fig. 17(e). In this circuit we used for the first *prepare* unitary the implementation from the first line of Eq. (F4) while for the final *prepare* unitary we used the second line of Eq. (F4) and then took the inverse. In addition, we also used the identity from Fig. 17(F) which can be obtained by using Eq. (F4). It is also possible to eliminate one additional CNOT gate by using a different decomposition of the controlled-H gate, as described in Fig. 17(g) and obtained from $H = Z R_y^\dagger(\pi/2)$. We notice that, if the bottom qubit is set to $|0\rangle$ before applying the circuit on the second

line above, one can drop the second CNOT gate because it only contributes a global phase. The same is true if instead we measure the projector in the bottom qubit state $|0\rangle$. We finally obtain the circuit in Fig. 17(h).

2. Nuclear excitation operator

We now show how to implement the LCU oracle for the more complicated case of the nuclear excitation operator,

$$O(\theta) = \alpha \mathbb{1} + \beta X - \alpha Z, \qquad (F5)$$

where the two real constants are given by Eq. (C10). We choose the state-to-operator mapping as follows

$$|00\rangle \to \mathbb{1} \quad |01\rangle \to \mathbb{1} \quad |10\rangle \to X \quad |11\rangle \to -Z. \qquad (F6)$$

For the case $\beta \geqslant 0$, the full circuit is given in Fig. 18(a), where the $\tilde{H}$ is defined as $\tilde{H} = R_y^\dagger(3\pi/4) X R_y(3\pi/4)$, we have the
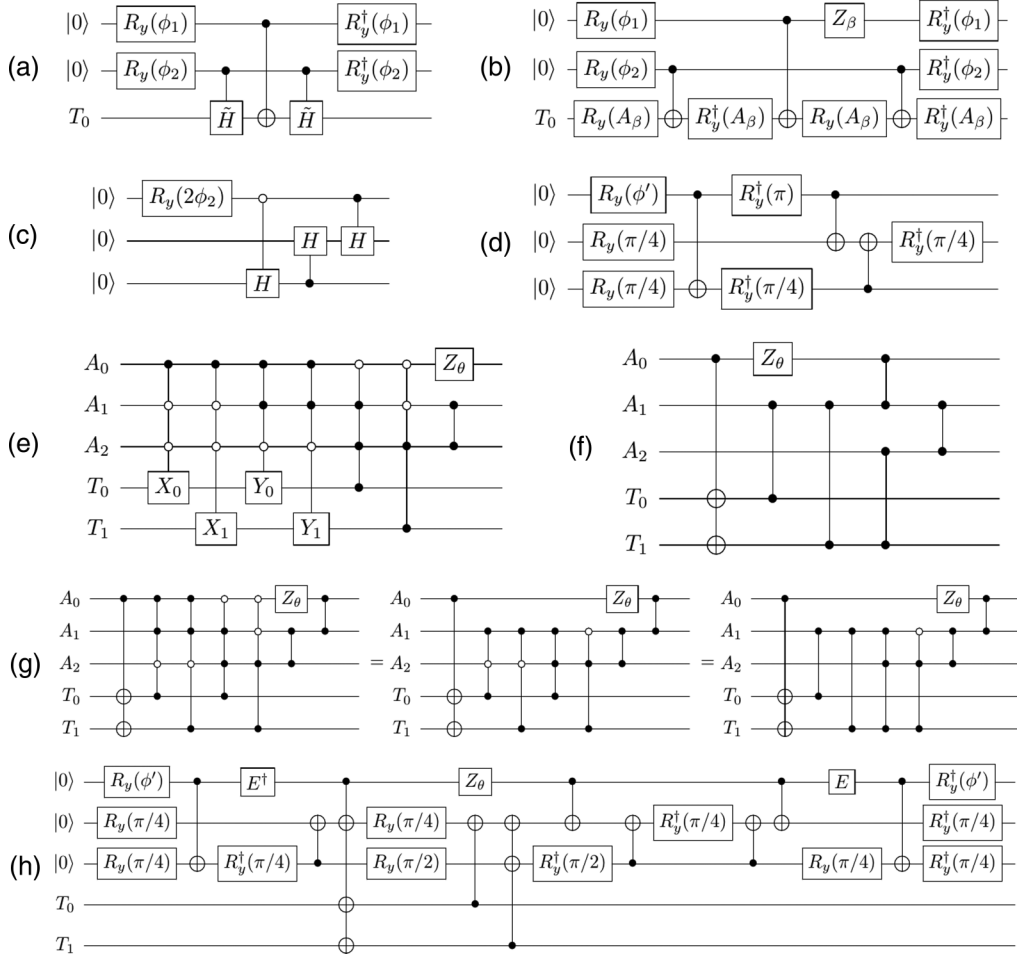
FIG. 18. Quantum circuits described in the text.

identity $\tilde{H}X\tilde{H} = -Z$. Also, the angles are

$$\phi_1 = 2\arcsin\left(\sqrt{\frac{\alpha + |\beta|}{2\alpha + |\beta|}}\right), \quad \phi_2 = 2\arcsin\left(\sqrt{\frac{\alpha}{\alpha + |\beta|}}\right). \tag{F7}$$

If $\beta \leqslant 0$ instead, we express the Hadamard gate H (modulo a phase) as $H = R_y(3\pi/4)XR_y^\dagger(3\pi/4)$ and use $HXH = Z$. We account for the negative sign in front of the last term in Eq. (F5) by applying the Z gate to the first ancilla qubit. The final circuit is shown in Fig. 18(b), where similarly to Eq. (42) we use

$$Z_\beta \equiv \begin{cases} \mathbb{1} & \text{for} \quad \beta \geqslant 0 \\ Z & \text{for} \quad \beta < 0, \end{cases} \tag{F8}$$

and the additional angle

$$A_\beta = \begin{cases} \frac{3\pi}{4} & \text{for} \quad \beta \geqslant 0 \\ \frac{-3\pi}{4} & \text{for} \quad \beta < 0. \end{cases} \tag{F9}$$

As explained in the main text, to test the flexibility of the LCU method on a more scalable version of the problem we also consider the implementation of the nuclear excitation operator in second quantization. In the following, we show how to implement the LCU oracle for the operator $\overline{O}_M(\theta)$

introduced in Eq. (58) as

$$\overline{O}_M(\theta) = \sin(\theta)\mathbb{1} + \frac{\cos(\theta)}{2}(X_0X_1 + Y_0Y_1)$$
$$- \frac{\sin(\theta)}{2}(Z_0 - Z_1). \tag{F10}$$

As we note in the main text, this operator differs from the original second-quantized operator in Eq. (57) by a global scale factor $(\alpha^2 + \beta^2)^{1/2}$.

Using an ancilla register of three qubits, it is useful to consider the following state-to-operator map

$$|000\rangle \to \mathbb{1}, \quad |001\rangle \to Z_1, \quad |011\rangle \to -Z_0,$$
$$|100\rangle \to X_0X_1, \quad |110\rangle \to Y_0Y_1, \tag{F11}$$

to construct the *prepare* and *select* unitaries. The former can be chosen as depicted in Fig. 18(c), with the angle

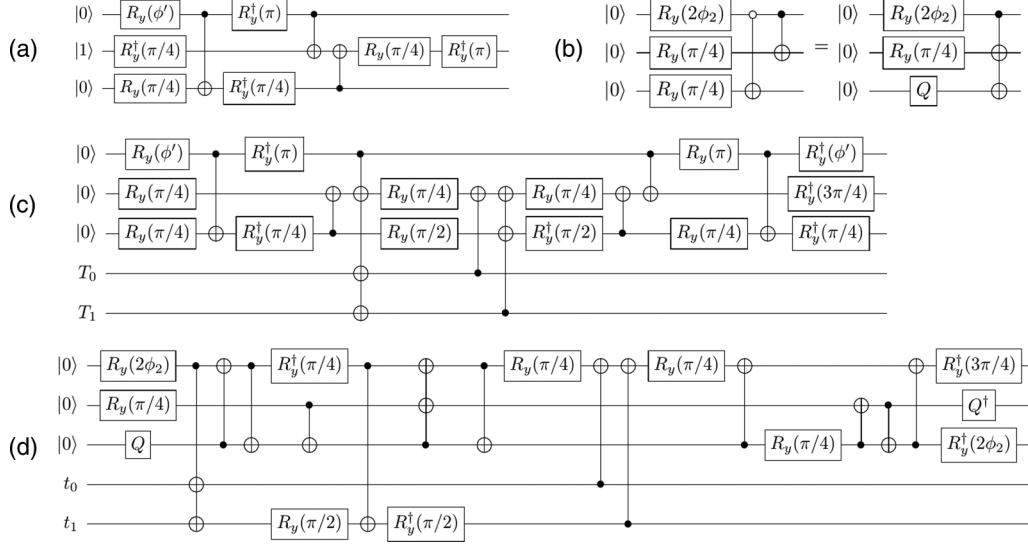$$\phi_2 = \arcsin\left(\sqrt{\frac{|\cos(\theta)|}{2\sin(\theta) + |\cos(\theta)|}}\right). \tag{F12}$$

FIG. 19. Quantum circuits described in the text.

The flag state prepared by this circuit reads

$$|\Phi_3\rangle = \frac{\cos(\phi_2)}{\sqrt{2}}\left(|000\rangle + \frac{|001\rangle + |011\rangle}{\sqrt{2}}\right) \\ + \frac{\sin(\phi_2)}{\sqrt{2}}(|100\rangle + |110\rangle). \quad \text{(F13)}$$

Using results and notations from the previous section, we can give an explicit implementation as depicted in Fig. 18(d), where $\phi' = 2\phi_2 + \pi$. This requires only three CNOT gates with fully connected qubits.

The *select* unitary corresponding to the mapping in Eq. (F11) can be represented as depicted in Fig. 18(e), with Z_θ defined in Eq. (42) and used to account for the sign associated with $\cos(\theta)$ and to recover the relative sign difference between Z_0 and Z_1 in Eq. (F10). As done for the simpler four-qubit case in Sec. III B, we exploit the fact that three states in the ancilla register are left unused to construct the implementation of V_S depicted in Fig. 18(f). This requires only seven entangling operations among pairs of qubits. To obtain this circuit we used the trick from Eq. (F3) and wrote the circuit in Fig. 18(e) as reported in Fig. 18(g).

With one additional step we obtain Fig. 18(f). The total circuit from Fig. 2 of the main text can be expressed as in Fig. 18(h).

This circuit has 13 CNOT gates and can be further simplified as we show next. We can absorb two of the controlled-Z operations by unpreparing the following flag state (slightly different from the previously prepared state):

$$|\widetilde{\Phi}_3\rangle = \frac{\cos(\phi_2)}{\sqrt{2}}\left(|000\rangle + \frac{|001\rangle - |011\rangle}{\sqrt{2}}\right) \\ + \frac{\sin(\phi_2)}{\sqrt{2}}(|100\rangle - |110\rangle), \quad \text{(F14)}$$

which can be produced by using a modified *prepare* unitary as depicted in Fig. 19(a). Here, the second ancilla qubit is initialized in $|1\rangle$. This scheme is very similar in spirit to the

asymmetric qubitization introduced in Ref. [41]. The total circuit is now given by the expression in Fig. 19(c) and the number of CNOT gates is reduced to 11. The final hardware-efficient circuit takes into account the connectivity constraints. It consists of a total number of 14 CNOT gates, and we have explicitly the circuit in Fig. 19(d). Here, $Q = R_y(\pi/4)X$ is obtained as in Fig. 19(b).

APPENDIX G: DEPOLARIZING NOISE AND RATIO ESTIMATORS

In this section we show how the ratio estimators for the transition probabilities, such as Eq. (39) for the time-dependent case and Eq. (49) for the LCU method, exhibit error resilience for a simple error model. We consider the simple situation where the whole system is initially in a pure state $\rho = |\Psi\rangle\langle\Psi|$ and then becomes subject to a depolarizing noise channel

$$\Omega_\epsilon[\rho] = (1-\epsilon)\rho + \frac{\epsilon}{d}\mathbb{1}. \quad \text{(G1)}$$

Here $d = 2^q$ for $q = n + k$ qubits, with n qubits for the system register and k ancillary qubits. We recall that, for the time-dependent method $k = 1$, and $k = \lceil\log(L+1)\rceil$ for the LCU-based method. The two observables we consider are the success probability P_s and the transition probability m. They can be expressed as

$$P_s = \text{Tr}[\Pi_s\rho], \quad m = \frac{1}{P_s}\text{Tr}[\Pi_s\Pi_{fin}\rho], \quad \text{(G2)}$$

where Π_s is a rank-K projector with $K = q - k = n$ and Π_{fin} is a rank-1 projector onto the final scattering state. For the LCU method we can find situations where the success probability is known beforehand to be equal to $\widetilde{P}_s$ (for instance, in the problem discussed in Sec. III where $\widetilde{P}_s = 1/\Lambda^2$), in this case we want to consider the alternative estimator

$$\widetilde{m} = \frac{1}{\widetilde{P}_s}\text{Tr}[\Pi_s\Pi_{fin}\rho], \quad \text{(G3)}$$

which is equivalent to m in the absence of errors.

We now want to compare the effect of the depolarizing channel (G1) on the two estimators m and $\widetilde{m}$ and show that the former is more stable against such perturbations. We denote observables evaluated on the modified density matrix $\rho_\epsilon = \Omega_\epsilon[\rho]$ with the superscript E. For the success probability we have

$$P_s^E = \text{Tr}[\Pi_s \Omega_\epsilon[\rho]] = (1 - \epsilon)P_s + \frac{\epsilon}{d}\text{Tr}[\Pi_s], \qquad (G4)$$

and from the definition of Π_s we arrive at

$$P_s^E = (1 - \epsilon)P_s + \frac{\epsilon}{2^k}. \qquad (G5)$$

Similarly for the transition probabilities we have

$$m^E = \frac{1}{P_s^E}\text{Tr}[\Pi_s \Pi_{fin} \Omega_\epsilon[\rho]] = \frac{1}{P_s^E}\left((1 - \epsilon)P_s m + \frac{\epsilon}{d}\right), \qquad (G6)$$

and for the alternative version instead

$$\widetilde{m}^E = \frac{1}{\widetilde{P}_s}\text{Tr}[\Pi_s \Pi_{fin} \Omega_\epsilon[\rho]] = (1 - \epsilon)\widetilde{m} + \frac{\epsilon}{d\widetilde{P}_s}. \qquad (G7)$$

In both of these derivations we used the definition of $\Pi_s \Pi_{fin}$ which implies $\text{Tr}[\Pi_s \Pi_{fin}] = 1$.

We start by considering the absolute error for the estimator in Eq. (G7), this is given by

$$\Delta \widetilde{m} = \epsilon \left| \frac{1}{d\widetilde{P}_s} - \widetilde{m} \right| \approx \epsilon \widetilde{m}, \qquad (G8)$$

where the last approximation valid in the limit $d\widetilde{P}_s \gg 1$. We consider now the ratio estimator from Eq. (G6), it is convenient to first express it as

$$m^E = \frac{(1 - \epsilon)P_s m + \frac{\epsilon}{d}}{(1 - \epsilon)P_s + \frac{\epsilon}{2^k}}$$

$$= \frac{(1 - \epsilon)P_s m + \frac{\epsilon}{d}}{(1 - \epsilon)P_s} - \frac{\epsilon}{2^k}\frac{(1 - \epsilon)P_s m + \frac{\epsilon}{d}}{[(1 - \epsilon)P_s + \epsilon\xi]^2}$$

$$= m + \frac{\epsilon}{1 - \epsilon}\frac{1}{dP_s} - \frac{\epsilon}{2^k}\frac{(1 - \epsilon)P_s m + \frac{\epsilon}{d}}{[(1 - \epsilon)P_s + \epsilon\xi]^2}, \qquad (G9)$$

for some $\xi \in [0, 1/2^k]$. In the second line of the expression above we used directly Lagrange's remainder theorem as we did in Eq. (A1) above. This yields the error estimate

$$\Delta m = \epsilon \left| \frac{1}{1 - \epsilon}\frac{1}{dP_s} - \frac{1}{2^k}\frac{(1 - \epsilon)P_s m + \frac{\epsilon}{d}}{((1 - \epsilon)P_s + \epsilon\xi)^2} \right|$$

$$\leqslant \frac{\epsilon}{1 - \epsilon}\left(\frac{1}{dP_s} + \frac{(1 - \epsilon)P_s m + \frac{\epsilon}{d}}{2^k(1 - \epsilon)P_s^2} \right)$$

$$= \frac{\epsilon}{1 - \epsilon}\left(\frac{1}{dP_s} + \frac{m}{2^k P_s} + \frac{\epsilon}{1 - \epsilon}\frac{1}{d2^k P_s^2} \right)$$

$$\approx \frac{\epsilon}{1 - \epsilon}\frac{m}{2^k P_s} = \frac{\Delta \widetilde{m}}{(1 - \epsilon)2^k P_s}, \qquad (G10)$$

with the approximation in the last line valid for $d\widetilde{P}_s \gg 1$ as above. Notice now that, for the LCU method, we also have $2^k P_s \gg 1$ whenever we use enough ancillas and therefore the relative error is exponentially suppressed. For the time-dependent method instead the inequality in Eq. (G10) does not lead to a suppression since $k = 1$ in the algorithm described in

the main text; note, however, that one can still add redundant qubits and perform postselection at the end.

APPENDIX H: ERROR MITIGATION

We want to describe now the error mitigation procedures employed in the main text. We used a modified version of the protocol of Ref. [8] which is based on techniques from Refs. [11,34,35]. The approach consists of two main steps. First we correct for the readout error noise (or measurement errors). This initial correction requires in principle 2^n measurements for n qubits; however, as we will show in the next section, we are able to reduce the total number of measurements to $2n$, assuming that the readout error is qubit independent. The second step of the error mitigation procedure aims to correct gate noise. Here we assume, consistent with the current error rates of the IBMQ machine, that the dominant noise channel is the one associated with the execution of CNOT gates. Therefore, given a circuit that implements the desired algorithm, we generate additional circuits, logically equivalent to the original one but with multiple CNOT gates. (We note that products of two CNOT gates are the identity.) We compute the wanted observables at different noise levels and extrapolate them to the zero-noise limit. The zero-noise-extrapolation procedure is scalable because it adds only a linear overhead in the number of gates to the original circuit depth (this can be further improved by using a stochastic variant, see, e.g., Ref. [42]).

1. Readout-error mitigation

We review here the readout-error correction scheme used in this work (see also Ref. [43]). In the following we assume the readout errors are independent for different qubits and can be described in terms of two parameters:

i. e_0: probability to get $|1\rangle$ when we prepare $|0\rangle$;
ii. e_1: probability to get $|0\rangle$ when we prepare $|1\rangle$;

which are both zero in the limit of no noise. These parameters can be obtained by performing two independent calculations: the first one prepares $|0\rangle$ and measures, the second prepares $|1\rangle$ and measures. The result of this calibration experiment is a calibration matrix P with entries

$$P = \begin{pmatrix} 1 - e_0 & e_1 \\ e_0 & 1 - e_1 \end{pmatrix}. \qquad (H1)$$

We can now use this information to mitigate this error source when computing observables. We now focus on the single-qubit case and comment on generalizations to the more general multiqubit scenario at the end. We indicate with $\mathbf{p} = (p, 1 - p)$ the probability vector in the limit of no errors and with $\mathbf{p}_e = (p_e, 1 - p_e)$ the corresponding probabilities with readout errors. The two are related through the calibration matrix and its inverse

$$P \cdot \mathbf{p} = \mathbf{p}_e \Rightarrow P^{-1} \cdot \mathbf{p}_e = \mathbf{p}, \qquad (H2)$$

and an explicit expression for the inverse transformation can be found

$$P^{-1} = \frac{1}{d} \begin{pmatrix} 1 - e_1 & -e_1 \\ -e_0 & 1 - e_0 \end{pmatrix}, \quad \text{(H3)}$$

where we have introduced $d = 1 - e_0 - e_1$. Note that the error parameters (e_0, e_1) are known only with finite precision Δe_0 and Δe_1 and this can have an important effect when applying this inverse transformation directly. In particular, it is possible that the resulting mitigation probability vector **p** contains unphysical negative components. In the calculations presented in this paper, we first estimate the expected error on the mitigated probabilities as explained below, if we find negative probabilities compatible with zero within the uncertainty band we manually set it to zero, otherwise we perform an additional least square inversion enforcing the calibration matrix to be physical (as implemented in QISKIT-IGNIS). We then use the resulting probabilities as corrected central values for the mitigated result while we keep the same error estimates coming from linear inversion. In practice we never had to use this fall-back procedure for the calculations reported in this work.

Here we indicate with Var[M] the matrix containing the variances of the matrix elements of a matrix M. With this notation we have for the calibration matrix

$$\text{Var}[P] = \begin{pmatrix} \Delta e_0^2 & \Delta e_1^2 \\ \Delta e_0^2 & \Delta e_1^2 \end{pmatrix}, \quad \text{(H4)}$$

while for the inverse matrix the variance to first order is

$$\text{Var}[P^{-1}] = \frac{1}{d^2} \begin{pmatrix} l_1 & r_1 \\ r_0 & l_0 \end{pmatrix}, \quad \text{(H5)}$$

where we have introduced

$$l_k = \Delta e_k^2 + (1 - e_k)^2 \Delta_{01}^2, \quad r_k = \Delta e_k^2 + e_k^2 \Delta_{01}^2, \quad \text{(H6)}$$

together with

$$\Delta_{01} = \frac{\Delta e_0^2 + \Delta e_1^2}{d^2}. \quad \text{(H7)}$$

It is now easy to propagate uncertainties to the final observable, here we describe in more detail the case of a single qubit observable but this can be easily generalized. Consider the following expectation value:

$$\langle O \rangle = \mathbf{a} \cdot \mathbf{p} = a_0 p_0 + a_1 p_1, \quad \text{(H8)}$$

we can estimate its value by using the error-affected probabilities $\mathbf{p}_e$ using the inverse of the calibration matrix

$$\langle \widetilde{O} \rangle = \mathbf{a} \cdot P^{-1} \mathbf{p}_e = \sum_{ij} a_i [P^{-1}]_{ij} [p_e]_j, \quad \text{(H9)}$$

together with the error estimate

$$\text{Var}[\langle \widetilde{O} \rangle] = \sum_i a_i^2 \left(\sum_j [P^{-1}]_{ij}^2 \Delta [p_e]_j^2 \right)$$
$$+ \sum_i a_i^2 \left(\sum_j \text{Var}[P^{-1}]_{ij} [p_e]_j^2 \right). \quad \text{(H10)}$$

In the more general multiqubit case the above equations carry out in a straightforward generalization, and for n qubits we have in principle to perform 2^n measurements in order to build the calibration matrix given in Eq. (H1). This cost can be substantially reduced if it is reasonable to assume that errors happen independently on different qubits, in this case the full calibration matrix factors into a series of n diagonal 2×2 blocks and the number of measurement needed is only $2n$. For the results reported in this work we checked that the assumption of independence is to realize to good accuracy by performing explicitly the full 2^n calibration runs.

2. Zero-noise extrapolation

We now describe in more detail the zero-noise extrapolation described briefly above. The idea is to collect data at different noise levels and then use a sensible parametrization of the noise dependence of an observable to extract a noise-free estimator. Since, as discussed above, in the quantum devices used in this work, the CNOT gate is the dominant source of noise, we use the total number N_C of CNOT in the circuit considered as a proxy to the noise level.

Results at higher noise levels are obtained by generating a series of equivalent circuits adding an even number of CNOT gates for each CNOT in the original circuit. We then compute the observables and the corresponding errors for each one of the circuits and denote them with (O_k, E_k) where $k \geqslant 0$ indicates the total number of CNOT gates, given by $N_C(k) = (2k - 1)N_C$, used in the noise-amplified circuit used to extract the expectation value. For example $k = 1$ and $k = 2$ denote two circuits having respectively N_C and $3N_C$ CNOT gates. We will also use O_F to indicate the error-free expectation value we are trying to estimate. In the absence of gate noise all the pairs (O_k, E_k) should be identical, but when CNOT gates have a failure probability p_ε, the effective failure probability will grow linearly with k.

The mitigation is performed by attempting three different types of zero-noise extrapolations. In particular for low noise levels we perform a Richardson extrapolation, proposed in Ref. [11], and a polynomial extrapolation, while for larger noise levels we use a two-point exponential extrapolation [35,44]. These techniques are briefly described below.

i. Richardson extrapolation: for small error probabilities p_ε we can expand the observable O as a power series in the error $\epsilon = N_C p_\varepsilon$,

$$O(\epsilon) = O_F + \sum_{j=1}^{M} c_j \epsilon^j + \mathcal{O}(\epsilon^{M+1}), \quad \text{(H11)}$$

where O_F is the error-free result and $c_j \in \mathbb{R}$ are expansion coefficients. The idea is to determine O_F by inverting the polynomial expression above by using $M + 1$ evaluations of $O(\epsilon)$ for different values of ϵ. We use the error-amplified results O_k defined above for this purpose. For instance, in the simple case $M = 1$ we have the following estimator for O_F:

$$O_F = \frac{1}{2}(3O_1 - O_2) + \mathcal{O}(\epsilon^2), \quad \text{(H12)}$$

which is correct up to first order in the error ϵ. This construction can be easily generalized to higher orders; in the calculations presented in the main text we attempt extrapolations up to order $k = 4$ (corresponding to a ninefold increase in CNOT count).

ii. Polynomial: in this case we perform a polynomial fit to $O(\epsilon)$ using all the available results, as done in Refs. [5,34]. In this work we use four results at progressively higher noise levels (O_1 through O_4) and attempt to perform polynomial fits up to third order. The lowest-order polynomial that achieves a result with $\chi^2 \leqslant 1$ is the preferred extrapolation (see also Sec. H 2 a for additional details on how we combine different extrapolations).

iii. Exponential: in this case we use a simple two point exponential fit to the results by assuming the following functional dependence of the observable from the noise strength ϵ:

$$O(\epsilon) = O_F \exp(-A\epsilon), \quad \text{with } A > 0. \qquad \text{(H13)}$$

We can now find the parameter A and an estimate for the error-free result O_F by combining two values of $O(\epsilon)$ at different noise levels. For instance, if we take $O_1 = O(\epsilon)$ and $O_k = O((2k-1)\epsilon)$, we find

$$O_F = O_1 \exp(A\epsilon) = O_1 \left(\frac{O_1}{O_k}\right)^{\frac{1}{2(k-1)}}. \qquad \text{(H14)}$$

In the remaining sections we describe how we test for consistency in these extrapolations and how we combine the extrapolated results into a single value.

a. Validating extrapolation results

It is important to check *a posteriori* that the results of the zero noise extrapolations described above are consistent with the data. This can help in both detecting hardware problems in specific runs (cf. filtered results in Ref. [8]) but also in identifying the best extrapolation procedure for a given dataset and target observable.

In this work we use the same procedure devised in Ref. [8] and explained in more details in this section. In our work we always attempt to use three additional data points at higher error levels (corresponding to the choices $k = 2, 3, 4$ for the CNOT repetitions), and for convenience we denote with N_{ex}^{max} the maximum order in this expansion (i.e., $N_{ex}^{max} = 4$ in our case). We then use the procedure detailed below to check for internal consistency of the N_{ex}^{max} extrapolated results $\{O_k^e\}_{k=1,\dots,N_{ex}^{max}}$ and their associated error estimates $\{E_k^e\}_{k=1,\dots,N_{ex}^{max}}$. In this notation, the result obtained after readout-error mitigation, denoted with a superscript r, corresponds to the pair $(O_1^e, E_1^e) = (O_1^r, E_1^r)$.

Before describing the algorithm for consistency check, we need to introduce one more definition: we say that two data points (O_A, E_A) and (O_B, E_B) are *compatible* with each other at the $m\sigma$ level if

$$|O_A - O_B| \leqslant m\sqrt{E_A^2 + E_B^2}, \quad m \geqslant 1. \qquad \text{(H15)}$$

We indicate this relation compactly as

$$(O_A, E_A) \overset{m}{==} (O_B, E_B). \qquad \text{(H16)}$$

We can now describe the consistency-check procedure we used. We consider the extrapolation at some order k to have succeeded if the result (O_k^e, E_k^e) is *compatible* at the 1σ level[2] with all the higher ones (O_l^e, E_l^e) for $l > k$, if this check fails for $k > 1$ we increase by one an internal error count. The reason behind this is that, if a result changes significantly by adding one more order, the lower-order extrapolation is unlikely to have converged. At the same time we want to keep the order as small as possible since higher-order extrapolations will rely on possibly much noisier data.

We use the algorithm described in pseudocode in Algorithm 1 to determine the lowest-order result that is compatible with the available higher-order ones and also collect the number of failures using the error count. If the check returns this corresponds to $k = 1$, then we flag the result as not needing extrapolation. If instead the check returns $k > 1$ but an error count of zero, we flag the result as *error free*. Finally, if no order passes the consistency check we flag the result as *failed*.

Algorithm 1 Consistency check

1: define BOOL array *ctest* of size $[N_{ex}^{max}]$
2: $error\_count \leftarrow 0$
3: **for** $i = 1$ to N_{ex}^{max} **do**
4:     $ctest[i] \leftarrow True$
5:     **for** $j = i + 1$ to N_{ex}^{max} **do**
6:         **if not** not $[(O_A, E_A) \overset{m}{==} (O_B, E_B)]$ **then**
7:             $ctest[i] \leftarrow False$
8:             $error\_count \leftarrow error\_count + 1$
9:             **break**
10: return array *ctest* and integer *error_count*

Since the data obtained at high order in noise amplification could be too noisy to allow extrapolations that pass this test, in the case of a *failed* extrapolation we attempt again by removing the highest order result at $k = N_{ex}^{max}$ and repeat the procedure. Note that, as the new consistency checks are performed, the error count is not re-initialized and keeps collecting the number of failures. If the consistency check fails the second time we definitively flag the result as *failed* and avoid using it in the next manipulations.

The procedure just described attempts to use all the information collected in the experiment to select extrapolations in which we have a higher confidence and at the same time alerting us if the confidence level is too low. For this to work well it is important to check that the result at the highest noise levels do contain nontrivial information in the first place, here we use the same strategy used to find decohered runs in Ref. [8] to avoid performing uncontrolled extrapolations.

---

[2]The effect of making a different choice for m is explored in previous work [8].

Contrary to what we found in Ref. [8], the results presented in this paper were never flagged by this procedure.

b. Combining different extrapolation results

At the end of the consistency check procedure described in the previous section, we collect the non-*failed* results into a list of triples (O_k^e, E_k^e, C_k) containing the best value O_k^e and its error estimate E_k^e, together with the total error count C_k obtained for the given extrapolation method. Note that, as mentioned above, both the polynomial and exponential extrapolation strategies can have higher error counts than the Richardson strategy since we are also checking for χ^2 compatibility (in the polynomial case) and sign changes (in the exponential case).

A final unique result is then obtained as follows: if the global linear fit was successful then we use that as our best result; if not, then we average together the results with the smallest error count C_k. This allows us to avoid problems in regimes where Richardson is well behaved but the exponential is not and vice versa.

3. Error propagation

The results reported in the main text contain also an error which we use to quantify the effect of statistical fluctuations.[3] In this work we obtained these errors by using a simple resampling technique: we first obtain the readout-mitigated results $\{O_k^r\}$ at all the N_{ex}^{max} noise levels with their corresponding errors $\{O_k^r\}$, we then associate to the kth result a Gaussian distribution of mean O_k^r and variance E_k^r and obtain M samples from this distribution. For each of the M sets of N_{ex}^{max} resampled observables we perform a full extrapolation neglecting the resulting error estimates. We finally compute the mean and 68% probability confidence interval from the collection of M extrapolated results.

Results obtained by explicitly propagating the errors throughout the calculation (as done in Ref. [8]) produces results that are very similar but with a mildly increased dispersion. This is likely an effect generated by both our choice of parameters for the consistency check and of the procedure to combine results together: when two results are only barely failing a consistency check this might result in an exclusion of that extrapolation method in the final average, but a small fluctuation can change this and possibly considerably shift the final optimal result. The resampling procedure we adopt ensures that these effects are smoothed out.

———————

[3]Since we are combining together results from different approximations the error bar contains information about the systematic errors, too.

[1] A. Steane, Quantum computing, Rep. Prog. Phys. **61**, 117 (1998).

[2] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura, C. Monroe, and J. L. O'Brien, Quantum computers, Nature (London) **464**, 45 (2010).

[3] E. A. Martinez, C. A. Muschik, P. Schindler, D. Nigg, A. Erhard, M. Heyl, P. Hauke, M. Dalmonte, T. Monz, P. Zoller, and R. Blatt, Real-time dynamics of lattice gauge theories with a few-qubit quantum computer, Nature (London) **534**, 516 (2016).

[4] D. B. Kaplan, N. Klco, and A. Roggero, Ground states via spectral combing on a quantum computer, arXiv:1709.08250.

[5] E. F. Dumitrescu, A. J. McCaskey, G. Hagen, G. R. Jansen, T. D. Morris, T. Papenbrock, R. C. Pooser, D. J. Dean, and P. Lougovski, Cloud Quantum Computing of an Atomic Nucleus, Phys. Rev. Lett. **120**, 210501 (2018).

[6] N. Klco, E. F. Dumitrescu, A. J. McCaskey, T. D. Morris, R. C. Pooser, M. Sanz, E. Solano, P. Lougovski, and M. J. Savage, Quantum-classical computation of Schwinger model dynamics using quantum computers, Phys. Rev. A **98**, 032331 (2018).

[7] A. Roggero and J. Carlson, Dynamic linear response quantum algorithm, Phys. Rev. C **100**, 034610 (2019).

[8] A. Roggero, A. C. Y. Li, J. Carlson, R. Gupta, and G. N. Perdue, Quantum computing for neutrino-nucleus scattering, Phys. Rev. D **101**, 074038 (2020).

[9] E. Caurier, G. Martínez-Pinedo, F. Nowack, A. Poves, and A. P. Zuker, The shell model as a unified view of nuclear structure, Rev. Mod. Phys. **77**, 427 (2005).

[10] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

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

[12] 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 (London) **567**, 491 (2019).

[13] E. G. Adelberger, A. García, R. G. Hamish Robertson, K. A. Snover, A. B. Balantekin, K. Heeger, M. J. Ramsey-Musolf, D. Bemmerer, A. Junghans, C. A. Bertulani, J.-W. Chen, H. Costantini, P. Prati, M. Couder, E. Uberseder, M. Wiescher, R. Cyburt, B. Davids, S. J. Freedman, M. Gai *et al.*, Solar fusion cross sections. II. The *pp* chain and CNO cycles, Rev. Mod. Phys. **83**, 195 (2011).

[14] T.-S. Park, K. Kubodera, D.-P. Min, and M. Rho, Effective field theory for low-energy two-nucleon systems, Phys. Rev. C **58**, R637(R) (1998).

[15] J.-W. Chen, G. Rupak, and M. J. Savage, Nucleon-nucleon effective field theory without pions, Nucl. Phys. A **653**, 386 (1999).

[16] J.-W. Chen and M. J. Savage, $n \vec{p} \, d\gamma$ for big-bang nucleosynthesis, Phys. Rev. C **60**, 065205 (1999).

[17] S. B. Bravyi and A. Yu. Kitaev, Fermionic quantum computation, Ann. Phys. (NY) **298**, 210 (2002).

[18] J. D. Whitfield, V. Havlíček, and M. Troyer, Local spin operators for fermion simulations, Phys. Rev. A **94**, 030301(R) (2016).

[19] K. Setia, S. Bravyi, A. Mezzacapo, and J. D. Whitfield, Superfast encodings for fermionic quantum simulation, Phys. Rev. Research **1**, 033033 (2019).

[20] M. Steudtner and S. Wehner, Quantum codes for quantum simulation of fermions on a square lattice of qubits, Phys. Rev. A **99**, 022308 (2019).

[21] C. Derby and J. Klassen, Low weight fermionic encodings for lattice models, arXiv:2003.06939.

[22] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, UK, 2010).

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

[24] G. Mazzola, P. J. Ollitrault, P. Kl. Barkoutsos, and I. Tavernelli, Nonunitary Operations for Ground-State Calculations in Near-Term Quantum Computers, Phys. Rev. Lett. **123**, 130501 (2019).

[25] H. Terashima and M. Ueda, Nonunitary quantum circuit, Int. J. Quantum Inform. **3**, 633 (2005).

[26] A. M. Childs and N. Wiebe, Hamiltonian simulation using linear combinations of unitary operations, Quantum Inf. Comput. **12**, 901 (2012).

[27] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, Simulating Hamiltonian Dynamics with a Truncated Taylor Series, Phys. Rev. Lett. **114**, 090502 (2015).

[28] G. H. Low and I. L. Chuang, Optimal Hamiltonian Simulation by Quantum Signal Processing, Phys. Rev. Lett. **118**, 010501 (2017).

[29] A. M. Childs, R. Kothari, and R. D. Somma, Quantum algorithm for systems of linear equations with exponentially improved dependence on precision, SIAM J. Comput. **46**, 1920 (2017).

[30] G. Vidal and C. M. Dawson, Universal quantum circuit for two-qubit transformations with three controlled-not gates, Phys. Rev. A **69**, 010301(R) (2004).

[31] F. Vatan and C. Williams, Optimal quantum circuits for general two-qubit gates, Phys. Rev. A **69**, 032315 (2004).

[32] Héctor Abraham *et al.*, "Qiskit: An open-source framework for quantum computing" (2019) .

[33] 5-qubit backed: IBM Q team, " IBM Vigo backend specification v1.0.2" (2020), retrieved from https://quantum-computing.ibm.com.

[34] Y. Li and S. C. Benjamin, Efficient Variational Quantum Simulator Incorporating Active Error Minimization, Phys. Rev. X **7**, 021050 (2017).

[35] S. Endo, S. C. Benjamin, and Y. Li, Practical Quantum Error Mitigation for Near-Future Applications, Phys. Rev. X **8**, 031027 (2018).

[36] V. V. Shende and I. L. Markov, On the CNOT-cost of Toffoli gates, Quantum Info. Comput. **9**, 461 (2009).

[37] G. H. Low and I. L. Chuang, Hamiltonian simulation by qubitization, Quantum **3**, 163 (2019).

[38] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics, in *STOC 2019: Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019 (Association for Computing Machinery, New York, 2019), p. 193–204.

[39] A. Roggero, Spectral-density estimation with the Gaussian integral transform, Phys. Rev. A **102**, 022409 (2020).

[40] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, Elementary gates for quantum computation, Phys. Rev. A **52**, 3457 (1995).

[41] R. Babbush, D. W. Berry, and H. Neven, Quantum simulation of the Sachdev-Ye-Kitaev model by asymmetric qubitization, Phys. Rev. A **99**, 040301(R) (2019).

[42] A. He, B. Nachman, W. A. de Jong, and C. W. Bauer, Zero-noise extrapolation for quantum-gate error mitigation with identity insertions, Phys. Rev. A **102**, 012426 (2020).

[43] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, Nature (London) **549**, 242 (2017).

[44] S. Endo, Q. Zhao, Y. Li, S. Benjamin, and X. Yuan, Mitigating algorithmic errors in a Hamiltonian simulation, Phys. Rev. A **99**, 012334 (2019).