Resource requirements for fault-tolerant quantum simulation: The ground state of the transverse Ising model

Craig R. Clark,[1] Tzvetan S. Metodi,[2] Samuel D. Gasster,[2] and Kenneth R. Brown[1,*]
[1]*School of Chemistry and Biochemistry and Division of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, Georgia 30332-0400, USA*
[2]*Computer Systems Research Department, The Aerospace Corporation, El Segundo, California 90245-4691, USA*
(Received 18 February 2009; published 15 June 2009)

We estimate the resource requirements, the total number of physical qubits and computational time, required to compute the ground-state energy of a one-dimensional quantum transverse Ising model (TIM) of N spin-1/2 particles, as a function of the system size and the numerical precision. This estimate is based on analyzing the impact of fault-tolerant quantum error correction in the context of the quantum logic array architecture. Our results show that a significant amount of error correction is required to implement the TIM problem due to the exponential scaling of the computational time with the desired precision of the energy. Comparison of our results to the resource requirements for a fault-tolerant implementation of Shor's quantum factoring algorithm reveals that the required logical qubit reliability is similar for both the TIM problem and the factoring problem.

DOI: 10.1103/PhysRevA.79.062314 PACS number(s): 03.67.Ac, 03.67.Lx, 03.67.Pp

I. INTRODUCTION

The calculation of the basic properties of quantum systems (eigenstates and eigenvalues) remains a challenging problem for computational science. One of the most significant issues is the exponential scaling of the computational resource requirements with the number of particles and degrees of freedom, which for even a small number of particles ($\sim$100) exceeds the capabilities of current computer systems. In 1982 Feynman addressed this problem by proposing that it may be possible to use one quantum system as the basis for the simulation of another [1]. This was the early promise of quantum simulation, and one of the original motivations for quantum computing. Since that time, many researchers have investigated different approaches to quantum simulation [2–7]. For example, Abrams and Lloyd proposed a quantum algorithm for the efficient computation of eigenvalues and eigenvectors using a quantum computer [4]. Many of the investigations into quantum simulation have assumed ideal performance from the underlying components resulting in optimistic estimates for the quantum computer resource requirements (number of qubits and time to completion). It is well known, however, that in order to address the effects of decoherence and other sources of faults and errors in the implementation of qubits and gates, it is necessary to incorporate fault-tolerant quantum error correction into an estimate of the resource requirements.

In this paper we estimate the resource requirements for a quantum simulation of the ground-state energy for the one-dimensional (1D) quantum transverse Ising model (TIM), specifically incorporating the impact of fault-tolerant quantum error correction. We apply the general approach of Abrams and Lloyd [3,4], and compute estimates for the total number of physical qubits and computational time as a function of the number of particles (N) and required numerical precision (M) in the estimate of the ground-state energy.

We have chosen to study the resource requirements for computing the ground-state energy for the 1D quantum TIM since this model is well studied in the literature and has an analytical solution [8–10]. The relevant details of the TIM are summarized in Sec. II. In Sec. III, we map the calculation of the ground-state energy for the TIM onto a quantum phase estimation circuit that includes the effects of fault-tolerant quantum error correction. The required unitary transformations are decomposed into one-qubit gates and two-qubit controlled-NOT gates using gate identities and the Trotter formula. The one-qubit gates are approximated by a set of gates which can be executed fault tolerantly using the Solovay-Kitaev theorem [11]. In Sec. III C, the quantum circuit is mapped onto the quantum logic array (QLA) architecture model, previously described by Metodi *et al.* [12]. Our final results, utilizing the QLA architecture, are given in Sec. III D including a discussion of how improvements in the underlying technology affects the performance for executing the TIM problem. In Sec. IV, we extend our resource estimate from 1D to higher dimensions. In Sec. V, we compare our present results for the TIM quantum simulation with a previous analysis of the resource requirements for Shor's factoring algorithm [12,13]. Finally, our conclusions are presented in Sec. VI.

II. TRANSVERSE ISING MODEL

The 1D transverse Ising model is one of the simplest models exhibiting a quantum phase transition at zero temperature [8,9,14,15]. The calculation of the ground-state energy of the TIM varies from analytically solvable in the linear case [8] to computationally inefficient for frustrated two-dimensional (2D) lattices [16]. For example, the calculation of the magnetic behavior of frustrated Ising antiferromagnets requires computationally intensive Monte Carlo simulations [17]. Given the difficulty of the generic problem and the centrality of the TIM to studies of quantum phase transitions and quantum annealing, the TIM is a good benchmark model for quantum computation studies.

*ken.brown@chemistry.gatech.edu

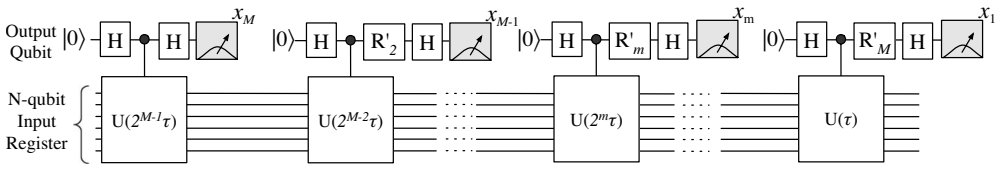

FIG. 1. The circuit for implementing the phase estimation algorithm using one continuously recycled control qubit.

The transverse Ising model consists of N spin-1/2 particles with nearest-neighbor spin-spin interactions along the z axis in the presence of an external magnetic field along the x axis. The Hamiltonian, H_I, for this system is

$$H_I = \sum_i \Gamma \sigma_i^x + \sum_{\langle i,j \rangle} J_{ij} \sigma_i^z \sigma_j^z, \tag{1}$$

where J is the spin-spin interaction energy, Γ is the energy of a spin the external magnetic field, and $\langle i,j \rangle$ implies a sum only over nearest neighbors [9]. σ_i^x and σ_i^z are the Pauli-spin operators for the ith spin, and we set $\hbar = 1$ throughout this paper.

In present work we focus on the 1D linear chain TIM of N spins with constant Ising interaction energy $J_{ij} = -J$. The ground state of the system is determined by the ratio of $g = \Gamma / J$. For the large magnetic field case, $g \gg 1$ the system is paramagnetic with all the spins aligned along the $\hat{x}$ axis, and in the limit of small magnetic field, $g \ll 1$, the system has two degenerate ferromagnetic ground states, parallel and antiparallel to the $\hat{z}$ axis. In the intermediate range of magnetic field strength the linear 1D TIM exhibits a quantum phase transition at $g = 1$ [9].

The TIM Hamiltonian in Eq. (1), for the 1D case with constant coupling can be rewritten as

$$H_I = -J \left(\sum_{j=1}^{N} g X_j + \sum_{j=1}^{N-1} Z_j Z_{j+1} \right), \tag{2}$$

where the Pauli-spin operators, σ_j^x and σ_j^z, are replaced with their corresponding matrix operators X_j and Z_j. For the 1D TIM, the ground-state energy can be calculated analytically in the limit of large N [8]. In the case of a finite number of spins with nonuniform spin-spin interactions (J is not constant), it is possible to efficiently simulate the TIM using either the Monte Carlo method [18] or the density-matrix renormalization-group approach [10]. The challenge for classical computers comes from the 2D TIM on a frustrated lattice where the simulation scales exponentially with N. Applying the quantum phase estimation circuit to calculate the ground-state energy of the TIM requires physical qubit resources, which scale polynomially with N, and the number of computational time steps is also polynomial in N. In addition, just as the complexity of the problem is independent of the lattice dimension and layout when applying classical brute force diagonalization, the amount of resources required to apply the quantum phase estimation circuit is largely independent of the dimensionality of the TIM Hamiltonian.

III. TIM QUANTUM SIMULATION RESOURCE ESTIMATES

Our approach to estimating the resource requirements for the ground-state energy calculation with Hamiltonian H_I involves two steps. First, we follow the approach of Abrams and Lloyd and map the problem of computing the eigenvalues of the TIM Hamiltonian in Eq. (2) onto a phase estimation quantum circuit [3,4]. Second, we decompose each operation in the phase estimation circuit into a set of universal gates that can be implemented fault tolerantly within the context of the QLA architecture. This allows us an accurate estimate of the resources in a fault-tolerant environment.

A. Phase estimation circuit

The phase estimation algorithm calculates an M-bit estimate of the phase ϕ of the eigenvalue $e^{-i2\pi\phi}$ of the time evolution unitary operator $U(\tau) = e^{-iH_I\tau}$ for a fixed τ given an eigenvector of H_I. $\phi < 1$ and can be represented by the binary fraction $0.x_1 \ldots x_M$ [3,4]. The energy eigenvalue $E = \frac{2\pi\phi}{\tau}$ when $E\tau < 2\pi$. Calculation of the ground-state energy $|E_g|$ requires that $\tau < 2\pi/|E_g|$. For the 1D TIM, the magnitude of the ground-state energy $|E_g|$ is bounded by $NJ(1+g)$ [8]. In the region near the phase transition $g \approx 1$, we choose $\tau = (10JN)^{-1}$.

The quantum circuit for implementing the phase estimation algorithm is shown in Fig. 1. The circuit consists of two quantum registers: an N-qubit input quantum register prepared in an initial quantum state $|\Psi\rangle$ and an output quantum register consisting of a single-qubit recycled M times [19,20]. Each of the N qubits in the input register corresponds to one of the N spin-1/2 particles in the TIM model [21]. At the beginning of each of the M steps in the algorithm, the output qubit is prepared into the state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ using a Hadamard (H) gate. The H gate is followed by a controlled power of $U(\tau)$, denoted with $U(2^m\tau)$, applied on the input register, where $0 \le m \le M-1$.

Letting j denote the jth step in the circuit, each time the output qubit is measured (meter symbols) the result is in the mth bit in the estimate of ϕ, following the rotation of the output qubit via the gate,

$$R_j = |0\rangle\langle 0| + \exp\left(i\pi \sum_{m=M+2-j}^{M} \frac{2^{M+1} x_m}{2^{m+j}} \right) |1\rangle\langle 1|, \tag{3}$$

where the gate R_j corresponds to the application of the quantum Fourier transform on the output qubit at each step [19,20]. The result after each of the M measurements is an M-bit binary string $\{x_1 x_2 \ldots x_M\}$, which corresponds the M-bit approximation of ϕ given by $0.x_1 \ldots x_M$. Using this estimate

of ϕ, the corresponding energy eigenvalue $E=\frac{2\pi\phi}{\tau}$ will be the ground-state energy E_g with a probability equal to $|\langle\Psi|\Psi_g\rangle|^2$ [3], where $|\Psi_g\rangle$ is the ground eigenstate of H_I.

To maximize the probability of success $|\langle\Psi|\Psi_g\rangle|^2$, the initial quantum state $|\Psi\rangle$ should be an approximation of the ground state $|\Psi_g\rangle$. For arbitrary Hamiltonians the preparation of an approximation to $|\Psi_g\rangle$ is generally computationally difficult [22,23]. For certain cases, the preparation can be accomplished using classical approximation techniques to calculate an estimated wave function or adiabatic quantum state preparation techniques [6,21]. If the state can be prepared adiabatically, the resource requirements for preparing $|\Psi\rangle$ are comparable in complexity to the resource requirements for implementing the circuit for the phase estimation algorithm shown in Fig. 1 [21]. For this reason, we focus our analysis on estimating the number of computational time steps and qubits required to implement the circuit, assuming that the input register has been already prepared in the N-qubit quantum state $|\Psi\rangle$.

B. Decomposition of the TIM quantum circuit into fault-tolerant gates

Figure 1 in Sec. III A shows the TIM circuit at a high level, involving $N+1$ unitary operators. In this section, each unitary operation of the circuit is decomposed into a set of basic one- and two-qubit gates which can be implemented fault tolerantly using the QLA architecture. The set of basic gates used is

$$\{X,Z,H,T,S,\text{CNOT},\text{MEASURE}\}, \tag{4}$$

where MEASURE is a single-qubit measurement in the $\hat{z}$ basis, CNOT denotes the two-qubit controlled-NOT gate, and T and S gates are single-qubit rotations around the $\hat{z}$ axis by $\pi/4$ and $\pi/2$ radians, respectively. The high-level circuit operations which require decomposition are the controlled-$U(2^m\tau)$ gates and each R_j gate.

The controlled-$U(2^m\tau)$ gate can be decomposed using the second-order Trotter formula [24,25]. First, H_I is broken into two terms: $H_X=\Sigma_{j=0}^{N} g X_j$, representing the transverse magnetic field, and $H_{ZZ}=\Sigma_{j=0}^{N-1} Z_j Z_{j+1}$, representing the Ising interactions. By considering the related unitary operators

$$U_x(2\tau)=\prod_{j=1}^{N}\exp(-igX_j), \tag{5}$$

$$U_{zz}(2\tau)=\prod_{j=1}^{N-1}\exp(-iZ_jZ_{j+1}), \tag{6}$$

and setting $g=1$ (as discussed in Sec. II), we can construct the Totter approximation of $U(2^m\tau)$, denoted by $\tilde{U}(2^m\tau)$ as

$$U(2^m\tau)=[U_x(\theta)U_{zz}(2\theta)U_x(\theta)]^k+\epsilon_T=\tilde{U}(2^m\tau)+\epsilon_T, \tag{7}$$

where $\theta=(2^m\tau/k)$ and ϵ_T is the Trotter approximation error, which scales as $O[(2^m\tau)^3/k^2]$ [24]. The Trotter approximation error can be made arbitrarily small by increasing the integer Trotter parameter k. Since the controlled-$U(2^m\tau)$ corresponds to the $(M-m)$th bit, ϵ_T must be less than $1/2^{M-m}$, which is the precision of the $(M-m)$th measured bit in the binary fraction for the phase ϕ. Thus, when approximating $U(2^m\tau)$, k is increased until ϵ_T is less than $1/2^{M-m}$. For a given M, we estimate a numerical value for the Trotter parameter $k(m=0)=k_0$ as a function of $N\leq 10$, with the constraint that $\epsilon_T<1/2^M$. We thus find that for fixed M, k_0 scales as $1/N$. We extrapolate k_0 for larger N based on a power-law fit of $N\leq 10$. For $m>0$, we set $k=2^m k_0$, which will satisfy the error bound based on the scaling of ϵ_T with k.

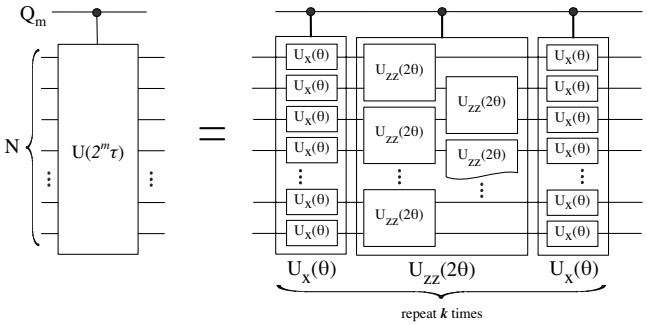

FIG. 2. Circuit for the controlled unitary operation $U(2^m\tau)$ approximated using the Trotter formula.

The circuit corresponding to the Trotter approximation of $U(2^m\tau)$ is shown in Fig. 2, where it can be seen that the controlled-$U(2^m\tau)$ is composed of two controlled-$U_x(\theta)$ operations and a controlled-$U_{zz}(\theta)$ operation, repeated k times and controlled on the mth instance of the output qubit denoted with Q_m. Expanding the circuit in Fig. 2, we can express $\tilde{U}(2^m\tau)$ as

$$\tilde{U}(2^m\tau)=U_x(\theta)[U_{zz}(2\theta)U_x(2\theta)]^{k-1}U_{zz}(2\theta)U_x(\theta), \tag{8}$$

which shows that approximating $U(2^m\tau)$ will require the sequential implementation of k controlled-$U_{zz}(2\theta)$ gates, $(k-1)$ controlled-$U_x(2\theta)$ gates, and two instances of controlled-$U_x(\theta)$ gates, all controlled on the mth instance of the output qubit.

The quantum circuits for the decomposition of the controlled-$U_x(2\theta)$ and controlled-$U_{zz}(2\theta)$ gates are shown in Figs. 3 and 4, respectively. The gates are decomposed into rotations about the $\hat{z}$ axis, $R_z(\theta)=\exp(-i\frac{\theta}{2}Z)$, and CNOT gates. $(N-1)$ additional qubits are used to prepare an N-qubit cat state in order to parallelize each of the N $R_z(\theta)$ gates. The preparation of an N-qubit cat state requires $(N-1)$ CNOT gates, which can be implemented in $O(N)$ time steps in parallel with the $R_z(\theta/4)$ gates in Fig. 3 and in parallel with the $R_z(\theta/2)$ gates in Fig. 4.

The three single-qubit R_z gates $[R_z(\theta)$, $R_z(\theta/2)$, and $R_z(\theta/4)]$ can be approximated using $O[\log^{3.97}(1/\epsilon_{sk})]$ basics gates (H,T,S) by the Solovay-Kitaev theorem [11,26]. The Solovay-Kitaev error (ϵ_{sk}) is equivalent to a small rotation applied to the qubit. The algorithm of Dawson and Nielsen [26] is used to compute the sequence of H, T, and S gates required to approximate each of the three R_z gates for $\theta=\frac{2^m\tau}{k}$. We define S_R as the length of the longest of these three sequences. For $M=30$, for example, we find that $S_R=4\times 10^5$, requiring a sixth-order Solovay-Kitaev approximation [26]. The results of this calculation show that the Solovay-

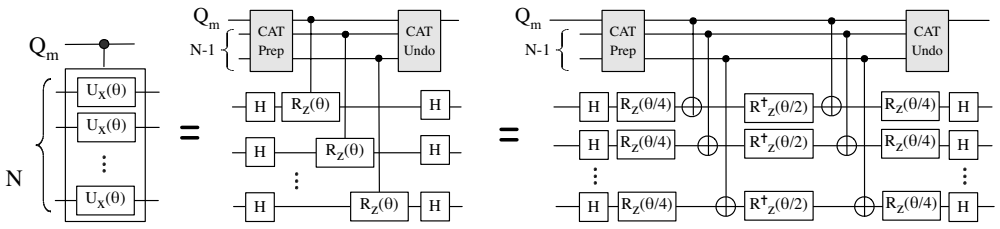

FIG. 3. The decomposition of the controlled unitary operation $U_x(\theta)$ into single-qubit R_z gates and CNOT gates.

Kitaev error $\epsilon_{sk} < \frac{\epsilon_T}{k}$, in order that the total error, ϵ_T, is less than the required precision $(1/2^{M-m})$, when we approximate $U(2^m\tau)$. As a result S_R scales as $O[\log^{3.97}(k/\epsilon_T)] = O(M^{3.97})$.

We now have a complete decomposition of the controlled-$U(2^m\tau)$ into the basic gates given in Eq. (4). As a function of S_R, the number of time steps required to implement controlled-$U_x(\theta)$ and $U_{zz}(\theta)$ is equal to $(3S_R+4)$ and $(6S_R+7)$, respectively. Following Eq. (8), the number of time steps required to implement the entire controlled-$U(2^m\tau)$ is $k(9S_R+11)+3S_R+4$, where $k=2^m k_0$. Each R_j gate in Fig. 1 is equivalent to at most a rotation by $R_z(\theta)$ and requires less than S_R gates.

Putting all of the above together, the total number of time steps (K) required to implement the TIM circuit as a function of S_R, k_0, and M is given by

$$K = \sum_{m=0}^{M-1} [2^m k_0(9S_R + 11) + 3S_R + 4 + S_R] = O(2^M)S_R. \quad (9)$$

Since S_R scales as $O(M^{3.97})$, the total number of time steps is dominated by the exponential dependence on the precision (M). The number of qubits Q required to implement the circuit is $2N$, since N qubits are needed for the input register $|\Psi\rangle$, one qubit is needed for the output register, and $N-1$ qubits are needed for the cat state.

In the next section we include fault-tolerant QEC into our circuit model and determine the resulting resource requirements, K and Q. We also provide an estimate on how long it could take to implement the TIM problem in real time by taking into account the underlying physical implementation of each gate and qubit in the context of the QLA architecture.

C. Mapping onto the QLA architecture

Incorporating quantum error correction and fault tolerance [27–30] into the TIM circuit design will impact the resource requirements in two ways. First, each of the qubits becomes a *logical qubit*, which is encoded into a state using a number of lower-level qubits. Second, each gate becomes a *logical gate*, realized via a circuit composed of lower-level gates applied on the lower-level qubits that make up a logical qubit. Each lower-level qubit may itself be a logical qubit all the way down to the physical level. Thus, quantum error correction and fault tolerance increase the number of physical time steps and qubits required to implement each basic gate and may even require additional logical qubits, depending on how each gate is implemented fault tolerantly and the choice of error correcting code. The resource requirements necessary to implement encoded logical qubits and gates will depend on the performance parameters of the underlying physical technology, the type of error correcting code used, and the level of reliability required per logical operation. The physical technology performance parameters that are taken into account in the design of the QLA architecture are the physical gate implementation reliability, time to execute a physical gate, and the time it takes for the state of the physical qubits to decohere.

The QLA architecture [12] is a tile-based homogeneous quantum computer architecture based on ion-trap technology, employing 2D surface electrode trap structures [31–33]. Each tile represents a single computational unit capable of storing two logical qubits and executing fault tolerantly any logical gate from the basic gate set given in Eq. (4). One of the key features of the QLA architecture is the teleportation-based logical interconnect which enables logical qubit exchange between any two computational tiles. The interconnect uses the entanglement-swapping protocol [34] to enable logical qubit communication without adding any overhead to the number of time required to implement a quantum circuit [12].

The QLA was originally designed to factor 1024 bit integers [12]. This requirement resulted in the need to employ

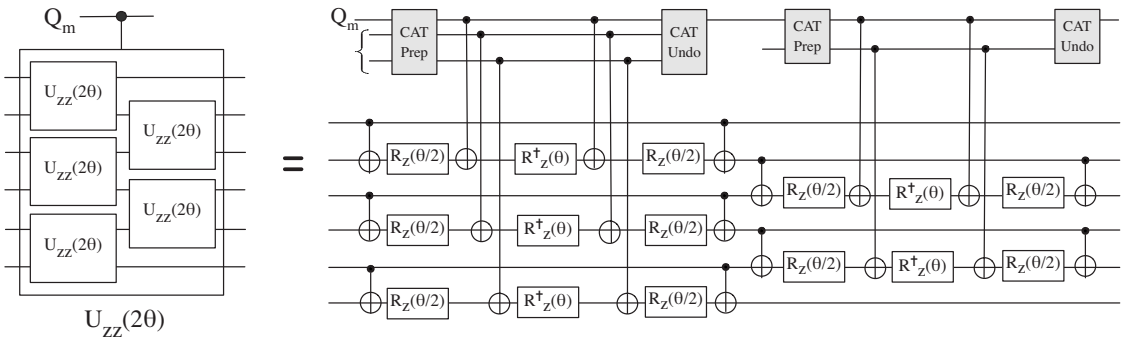

FIG. 4. The decomposition of the controlled unitary operation $U_{zz}(2\theta)$ gate into single-qubit R_z gates and CNOT gates.

the second-order concatenated Steane $[\![7,1,3]\!]$ quantum error correcting code [35]. Second-order concatenation means that each logical qubit is a level 2 qubit, composed of seven level 1 logical qubits each encoded into the state of seven physical ion-trap qubits.

To estimate the reliability for executing each of the basic gates fault tolerantly, a lower bound of 3.1×10^{-6} for the fault-tolerant threshold of the $[\![7,1,3]\!]$ code. This value was derived by Metodi *et al.* [36], by analysis of the ion-trap-based geometrical layout of each logical qubit tile. The $[\![7,1,3]\!]$ code threshold value used in the current research differs from the previously published estimate of 1.8×10^{-5} [37] due to our more detailed account of the operations specific to the ion-trap technology in the implementation of each logical qubit [36]. These threshold results are combined with Gottesman's method for including qubit movement [38] to estimate the reliability for each logical operation at levels 1 and 2.

Since each qubit in the $[\![7,1,3]\!]$ code moves an average of ten steps during error correction [36], we find that each level 1 gate has a failure probability of 3.2×10^{-10} and each level 2 gate has a failure probability of 3.5×10^{-14}. In our failure probability estimates, we have assumed optimistic physical ion-trap gate error probabilities of 10^{-7} per physical operation, consistent with recent ion-trap literature [39]. We also determine the physical resources required for each logical qubit. Each level 1 qubit requires 21 ion-trap qubits (7 data qubits and 14 ancilla to facilitate error correction) and each level 2 qubit requires 21 level 1 qubits. Given that the duration of each physical operation on an ion-trap device is currently of the order of 10 μs [40,41], the time required to complete a single error correction step is approximately 1.6 ms at level 1 and 0.26 s at level 2.

The number of logical qubits Q directly maps to the number of computational tiles required by the QLA, allowing us to estimate the size of the physical system. Similarly, the number of time steps K maps directly to the time required to implement the application since the duration of a single time step in the QLA architecture is defined as the time required to perform error correction, as discussed in Ref. [12]. We define an aggregated metric KQ called the problem size equal to $K\times Q$, which is an upper bound on the total number of logical gates executed during the computation [42]. The inverse of the problem size, $1/KQ$, is the maximum failure probability allowed in the execution of a logical gate [42], which ensures that the algorithm completes execution at least 36% of the time. Taking into consideration the failure probabilities per logical gate, the maximum problem size KQ which can be implemented in the QLA architecture is 3.1×10^{9} at level 1 error correction, 3×10^{13} at level 2, and 2.8×10^{20} at level 3. Level 3 error correction is not described in the design of the QLA architecture; however, its implementation is possible since a level 3 qubit is simply a collection of level 2 qubits and the architecture design does not change. The estimated failure probability for each level 3 logical gate is 3.6×10^{-21}.

The parameters K and Q for the TIM problem were estimated in Sec. III B, where Q was found to be $2N$ and K is $O(2^{M})\times S_R$. The fault-tolerant implementation of the T gate, however, requires an auxiliary logical qubit prepared into the state $T|+\rangle$ for one time step followed by four time steps composed of H, CNOT, S, and Measure gates [43], causing the value of K and Q to increase. Since many of the gates in the Solovay-Kitaev sequences approximating the R_z gates are T gates, when calculating K using Eq. (9), the value of S_R must take into consideration the increased number of cycles for each T gate. All other basic gates are implemented transversally and require only one time step.

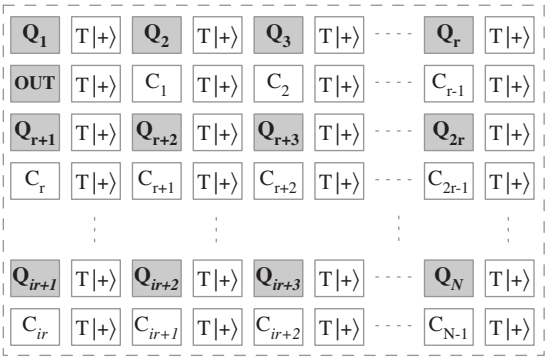

FIG. 5. QLA architecture for the TIM problem.

The resulting functional layout for the QLA architecture for the TIM problem is shown in Fig. 5. The architecture consists of $4N$ logical qubit tiles. The tiles labeled with Q_1 through Q_N are the data tiles which hold the logical qubits used in the N-qubit input register $|\Psi\rangle$ and the "OUT" tile is for the output register. The tiles labeled with C_1 through C_{N-1} are the $N-1$ qubit tiles for the cat state. The $T|+\rangle$ tiles are for the preparation of the auxiliary states in the event that T gates are applied on any of the data qubits. All tiles are specifically arranged as shown in Fig. 5 in order to minimize the communication required for each logical CNOT gate between the control and target qubits. For example, when preparing the cat state using all C_i tiles and the OUT tile, CNOT gates are required only between the OUT tile, C_1, and C_r. Similarly, C_1 interacts via a CNOT gate only with C_2, while C_2 interacts only with Q_3, during the cat state preparation.

D. Resource estimates for the 1D TIM problem

The resource requirements for implementing the 1D TIM problem using the QLA architecture are given in Fig. 6, where we show a logarithmic plot of the number of time steps K [calculated using Eq. (9)] as a function of the energy precision $M\leq20$, assuming $N=100$. The figure clearly shows K's exponential dependence on M. The dependence of K on the number of spins (N) is negligible and appears only in the k_0 term in Eq. (9) as $O(1/N)$, as discussed in Sec. III B. In fact, since $Q=4N$, we expect very little increase in the value of the total problem size KQ as N increases.

We see that for $M\leq8$ no error correction is required. This is because the required reliability per gate of $1/KQ$ is still below the physical ion-trap gate reliability of 1×10^{-7}. Without error correction, the architecture is composed entirely of physical qubits and all gates are physical gates. This means that each single-qubit R_z gate can be implemented directly without the need to approximate it using the Solovay-Kitaev theorem, resulting in $S_R=1$ in Eq. (9), and the total number

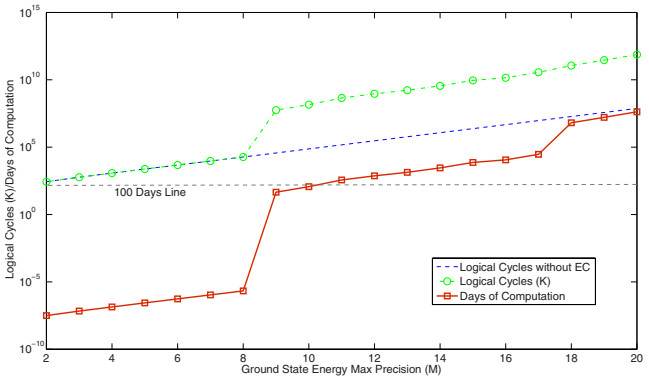

FIG. 6. (Color online) Numerical calculations for the number of logical cycles K (solid line) and days of computation necessary assuming $N=100$ spin TIM problem as a function of the desired maximum precision $M \leq 20$.

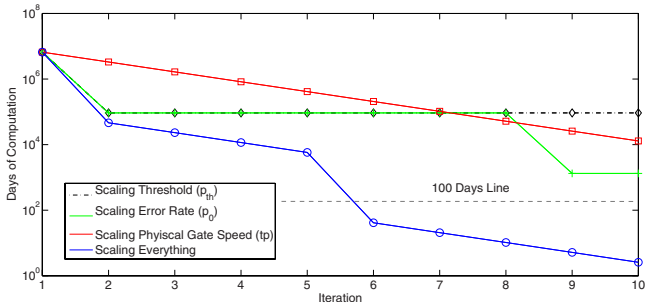

FIG. 7. (Color online) The total computation time in days as we vary the physical cycle time t_p (square markers), physical failure probability p_0 (starred markers), threshold failure probability p_{th} (diamond markers), and all together (circular markers) by a factor of 2 over ten iterations.

of qubits becomes $2N$ instead of $4N$. For $M \geq 9$ error correction is required, resulting in a sudden jump in the number of time steps at $M=9$, with an additional scaling factor of $O(M^4)$ in K due to S_R's dependence on M. In fact, K increases so quickly that at $M=9$ that level 2 error correction is required instead of level 1. At $M \geq 18$ level 3 error correction is required and while there is no increase in K, each time step is much longer, so there is a jump in the number of days of computation. The Solovay-Kitaev order [26] for $M=9$ is three and increases to order five for $M=20$.

E. Discussion of the resource estimates

Our resource estimates for the 1D TIM problem indicate that multiple levels of error correction, even for modest precision requirements, result in long computational times. As shown in Fig. 6, it takes longer than 100 days for $M>10$ with level 2 error correction. When level 3 error correction is required the estimated time is greater than 7.5×10^3 years.

The number of logical cycles K, which grows exponentially with M, contributes to the long computational times. However, the primary factor contributing to the long computational time is the time it takes to implement a single logical gate using error correction. Presently, it is difficult to see how one might reduce the value of K short of implementing a different approach for solving quantum simulation problems. On the other hand, the logical gate time can be improved by implementing small changes in three parameters: decreasing the physical gate time t_p, increasing the threshold failure probability p_{th}, and decreasing the underlying physical failure probability p_0.

The effect of these three parameters on the overall computational time for the 1D TIM problem is shown in Fig. 7. The figure shows how the total time, in days, for $M=18$ varies as we improve each of the three parameters by a factor of 2 during each of the ten iterations shown. The starting values for each parameter in the figure are 3.1×10^{-6} for p_{th}, 10^{-7} for p_0, and 10 μs for t_p. Decreasing the physical failure probability and increasing the threshold values by a factor of 2 during each iteration cause the number of days to decrease quadratically whenever lower error correction level is required, otherwise the number of days remains constant from one iteration to the next. A single change in the error correction level from level 3 to level 2 occurs by increasing p_{th} by a factor of 2 but there is no gain from additional increases in the threshold alone. Decreasing only p_0 by a factor of 512 yields a reduction of two levels of error correction.

From this analysis, we see that in order to reach a computational time on the order of 100 days with only level 1 error correction, we need to achieve parameter values of $p_{\text{th}}=1 \times 10^{-4}$, $p_0=3 \times 10^{-9}$, and $t_p=300$ ns or better. This provides goals for the improvement in the device technologies necessary for quantum simulation. It should also be noted that these parameters are not completely independent and improvements in one of them may result in improvements in the others. For example, improving the physical failure probability may lead to better threshold failure probability by allowing some of the underlying operations to be weighted against one another. Similarly, improving the threshold failure probability may require choosing a more efficient quantum error correcting code which could have a fundamentally shorter logical time step.

IV. GENERALIZING TO HIGHER SPATIAL DIMENSIONS

The ground-state energy of the 1D TIM can be efficiently computed using classical computing resources by taking advantage of the linear geometry of the spin configuration and significantly reducing the effective state space to a polynomial in N [10]. A 2D TIM with ferromagnetic and antiferromagnetic Ising couplings can be difficult to solve due to spin frustration. Many reductions to this problem still yield an exponential number of states with near degenerate energy [16]. As a result, the problem size scales exponentially with the size of the lattice. In contrast, the implementation of the quantum phase estimation circuit in Fig. 1 is independent of the values of Γ_i and J_{ij} and is weakly dependent on the geometry of the N spin system. This suggests that, given an approximation of the ground state, the ground-state energy of the TIM can be calculated for systems with random couplings [44] or higher dimensions with similar computational resources. Consider, for example, the calculation of the ground-state energy for the 2D Villain's model [45] using the phase estimation circuit.

Villain's model is a 2D square lattice Ising model with N^2 spin sites in which the rows have all ferromagnetic coupling and the columns alternate between ferromagnetic and antiferromagnetic. Each of the N^2 sites in Villain's model is represented by N^2 qubits in a $N \times N$ grid. The only change to the circuit for the phase estimation algorithm is the application of the U_{zz} Ising interaction, which must be decomposed into two successive steps. First the rows of spin states are treated as the 1D TIM problem in parallel, followed by the columns. Since the U_{zz} operations within each step are done in parallel, we still require $N/2$ additional qubits for the cat states. Given that the remaining operations, including the quantum Fourier transform implementation, remain the same, the increase in the number of time steps to implement an N^2-spin 2D TIM problem, compared to the 1D TIM problem, is by less than a factor of 2. Similarly, the increase in the resource requirements between a 1D and a three-dimensional TIM problem will be by less than a factor of 3.

V. COMPARISON WITH FACTORING

Since the QLA architecture was initially evaluated in the context of Shor's quantum factoring algorithm [13], it would be interesting to consider how the resource requirements for implementing the TIM problem compare to those for implementing the factoring algorithm. In this section, we compare the implementation of the two applications on the QLA architecture and highlight some important differences between each application.

Even though both applications employ the phase estimation algorithm, there are several important differences. First, the precision requirements are different. For Shor's quantum factoring algorithm, the precision M must scale linearly with the size N of the N-bit number being factored [13], where $N \geq 1024$ for modern cryptosystems. For quantum simulations, the desired precision is independent of the system size N, and the required M is small compared to factoring. The second difference lies in the implementation cost of the repeated powers of the controlled-$U(\tau)$ gates for each application. In Shor's algorithm, the gate is defined as $U(\tau)|x\rangle = |ax \bmod N\rangle$. Higher-order powers of the unitary can be generated efficiently via modular exponentiation [13]. The result is that the implementation of $U(2^m\tau)$ requires $2m$ times the number of gates used for $U(\tau)$. For generic quantum simulation problems, the implementation cost of $U(2^m\tau)$ equals 2^m times the cost of $U(\tau)$ because of the Trotter parameter k. The implementation of the control unitary gates for quantum simulation is not as efficient as that for the modular exponentiation unitary gates. The third difference lies in the preparation of the initial N-qubit state $|\Psi\rangle$. The preparation of $|\Psi\rangle$ for the TIM problem by adiabatic evolution is comparable in resource requirements to the phase estimation circuit. For Shor's quantum factoring algorithm $|\Psi\rangle = |1\rangle$ in the computational basis and is easily prepared.

Finally, factoring integers large enough to be relevant for modern cryptanalysis requires several orders of magnitude more logical qubits than the scale of quantum simulation problems considered in this paper. At minimum, the factoring of an N-bit number requires $2N+3$ qubits using the same one-control qubit circuit given in Fig. 1 [46]. As shown later in this section, choosing to use only the minimum number of qubits required for factoring leads to very high error correction overhead. A more reasonable implementation of the factoring algorithm requires $O(N^2)$ number of logical qubits, which corresponds to millions of logical qubits for factoring a 1024 bit number. Quantum simulation problems require significantly less computational space and the problems considered in this paper require less than 500 logical qubits.

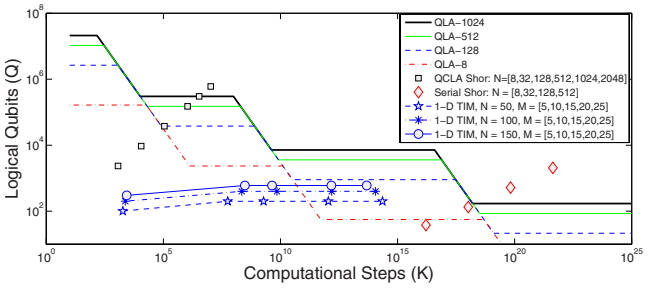

FIG. 8. (Color online) Performance characteristics of different QLA-based quantum computers in KQ space with fixed amount of physical resources. The binary precision for the Ising problem of $M=\{5,10,15,20,25\}$ corresponds to decimal precision of $\{1,3,4,6,7\}$ digits, respectively.

We examine how these differences affect the relative size of the QLA architecture required to implement each application. In particular, Fig. 8 shows the performance of QLA-based quantum computers in KQ space with fixed physical resources. Each horizontal line corresponds to the KQ limit for a QLA-based architecture modeled for factoring a 1024 bit number (topmost horizontal dashed line), a 512 bit number, a 128 bit number, and an 8 bit number, respectively. The physical resources for each QLA-N quantum computer (where $N=\{1024,512,128,8\}$ bits) are determined by how many logical qubits at level 2 error correction are required to implement the quantum carry look-ahead adder factoring circuit [12,47], which requires $O(N^2)$ logical qubits and $O(N \log^2 N)$ logical cycles. The plateaus in each QLA-N line of Fig. 8 represent using all of the qubits at a specific level of encoding, with the topmost right-hand plateau representing level 1. Where the lines are sloped, the model that is only a certain number of the lower-level encoded qubits can be used. Once this reaches the number of qubits that can be encoded at the next level, the quantum computer is switched from encoding level L to $L+1$ by using all the available level L qubits.

A QLA-N quantum computer is capable of executing an application using level L encoded qubits if the application instance is mapped *underneath* the line representing the computer at level L in Fig. 8. Factoring a 1024 bit number, for example, falls directly on the level 2 portion of the QLA-1024 line (see the square markers). Anything above that line cannot be implemented with the QLA-1024 computer. Similarly, factoring a 128 bit number maps under the QLA-128 line but can be accomplished using level 1 qubits. The TIM problem is mapped onto Fig. 8 for $N=50$, 100, 150, and several binary precision instances: $M=\{5,10,15,20,25\}$. As expected, factoring requires many more logical qubits; however, both applications require similar levels of error correc-

tion. A decimal precision of up to four digits of accuracy ($M=15$) can be reached by using a quantum computer capable of factoring an 8 bit number at level 2 error correction; however higher precision quickly requires level 3 error correction.

The resources for implementing quantum factoring with one-control qubit were calculated following the circuit in Fig. 1, where the unitary gates are replaced with the unitary gates corresponding to modular exponentiation, as discussed in Ref. [46]. The results are shown with the diamond-shaped markers in Fig. 8. While this particular implementation is the least expensive factoring network in terms of logical qubits, the high-precision requirement of $M=O(N)$ makes this network very expensive in terms of time steps. In fact, the number of time steps required pushes the reliability requirements into level 4 error correction for factoring even modestly sized numbers.

VI. CONCLUSION

In this paper, the TIM quantum simulation circuit was decomposed into fault-tolerant operations and we estimated the circuit resource requirements and number of logical cycles K as a function of the desired precision M in the estimate of the ground-state energy. Our resource estimates were based on the QLA architecture and underlying technology parameters of trapped ions allowing us to estimate both K, as a function of the level of the error correction level, and the total length of the computation in real time.

Our results indicate that even for small precision requirements K is large enough to require error correction. The growth of K is due to its linear dependence on the Trotter parameter k, which scales exponentially with the maximum desired precision M. In order for K to scale polynomially with the precision, new quantum simulation algorithms are required or systems must be chosen where the phase estimation algorithm can be implemented without the Trotter formula. The linear dependence of the number of time steps on k is due to the fact that U_x and the U_{zz} do not commute. However, there are some physical systems, whose Hamiltonians are composed of commuting terms, such as the nontransversal classical Ising model, which has a solution to the partition function in two dimensions but is intractable for higher dimensions [48]. In those cases, Trotterization is unnecessary. In a future work, we intend to generalize the calculations of the resource requirements to other physical systems and consider different ways to implement the phase estimation algorithm that limit its dependence on the Trotter formula.

[1] R. Feynman, Int. J. Theor. Phys. **21**, 467 (1982).
[2] S. Lloyd, Science **273**, 1073 (1996).
[3] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **79**, 2586 (1997).
[4] D. S. Abrams and S. Lloyd, Phys. Rev. Lett. **83**, 5162 (1999).
[5] C. Zalka, Proc. R. Soc. London, Ser. A **454**, 313 (1998).
[6] A. Aspuru-Guzik, A. Dutoi, P. Love, and M. Head-Gordon, Science **309**, 1704 (2005).
[7] I. Kassal, S. P. Jordan, P. J. Love, M. Mohseni, and A. Aspuru-Guzik, Proc. Natl. Acad. Sci. U.S.A. **105**, 18681 (2008).
[8] P. Pfeuty, Ann. Phys. **57**, 79 (1970).
[9] S. Sachdev, *Quantum Phase Transitions* (Cambridge University Press, Cambridge, England, 1999).
[10] A. Juozapavicius, S. Caprara, and A. Rosengren, Phys. Rev. B **56**, 11097 (1997).
[11] A. Y. Kitaev, A. H. Shen, and M. N. Vyalyi, *Classical and Quantum Computation*, Graduate Studies in Mathematics Vol. 47 (American Mathematical Society, Providence, 2002).
[12] T. S. Metodi, D. D. Thaker, A. W. Cross, F. T. Chong, and I. L. Chuang, Proceedings of the 38th Annual IEEE/ACM International Symposium on Microarchitecture (IEEE Computer Society, Washington, D.C., 2005), pp. 305–318.
[13] P. W. Shor, Proceedings of the 35th Annual Symposium on Foundations of Computer Science (IEEE, Los Alamitos, 1994), pp. 124–134.
[14] R. J. Elliott, P. Pfeuty, and C. Wood, Phys. Rev. Lett. **25**, 443 (1970).
[15] R. Jullien, P. Pfeuty, J. N. Fields, and S. Doniach, Phys. Rev. B **18**, 3568 (1978).
[16] R. Moessner and S. L. Sondhi, Phys. Rev. B **68**, 054405 (2003).
[17] Y. Hu and A. Du, J. Phys.: Condens. Matter **20**, 125225 (2008).
[18] M. Santos, Phys. Rev. E **61**, 7204 (2000).
[19] S. Parker and M. B. Plenio, Phys. Rev. Lett. **85**, 3049 (2000).
[20] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, Proc. R. Soc. London, Ser. A **454**, 339 (1998).
[21] K. R. Brown, R. J. Clark, and I. L. Chuang, Phys. Rev. Lett. **97**, 050504 (2006).
[22] J. Kempe, A. Kitaev, and O. Regev, SIAM J. Comput. **35**, 1070 (2006).
[23] J. D. Biamonte and P. J. Love, Phys. Rev. A **78**, 012352 (2008).
[24] M. Suzuki, Phys. Lett. A **165**, 387 (1992).
[25] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2001).
[26] C. M. Dawson and M. A. Nielsen, Quantum Inf. Comput. **6**, 81 (2006).
[27] P. Shor, Proceedings of the 37th Annual Symposium on the Foundations of Computer Science (IEEE, Los Alamitos, 1996), pp. 56–65.
[28] D. Aharonov and M. Ben-Or, Proceedings of the 29th Annual ACM Symposium on Theory of Computing (ACM, New York, 1997), Vol. 29, pp. 176–188.
[29] A. Y. Kitaev, Proceedings of the 3rd International Conference of Quantum Communication and Measurement, edited by O. Hirota, A. S. Holevo, and C. M. Caves (Plenum, New York, 1997), pp. 181–188.
[30] A. M. Steane, Fortschr. Phys. **46**, 443 (1999).
[31] J. I. Cirac and P. Zoller, Phys. Rev. Lett. **74**, 4091 (1995).

[32] D. J. Wineland *et al.*, Philos. Trans. R. Soc. London, Ser. A **361**, 1349 (2003).
[33] J. Chiaverini, R. B. Blakestad, J. Britton, J. D. Jost, C. Langer, D. Leibfried, R. Ozeri, and D. J. Wineland, Quantum Inf. Comput. **5**, 419 (2005).
[34] W. Dur, H. J. Briegel, J. I. Cirac, and P. Zoller, Phys. Rev. A **59**, 169 (1999).
[35] A. M. Steane, Phys. Rev. Lett. **77**, 793 (1996).
[36] T. M. Metodi, D. D. Thaker, and F. T. Chong, Proceedings of the 5th Workshop on Non-Silicon Computing (NSC-5), 2007 (unpublished).
[37] K. M. Svore, D. P. DiVincenzo, and B. M. Terhal, Quantum Inf. Comput. **7**, 297 (2007).
[38] D. Gottesman, J. Mod. Opt. **47**, 333 (2000).
[39] R. Ozeri *et al.*, Phys. Rev. Lett. **95**, 030403 (2005).
[40] S. Seidelin *et al.*, Phys. Rev. Lett. **96**, 253003 (2006).
[41] R. Reichle, E. Knill, J. Britton, R. Blakestad, J. Jost, C. Langer, R. Ozeri, S. Seidelin, and D. Wineland, Nature (London) **443**, 838 (2006).
[42] A. M. Steane, Phys. Rev. A **68**, 042322 (2003).
[43] P. Aliferis, D. Gottesman, and J. Preskill, Quantum Inf. Comput. **6**, 97 (2006).
[44] K. Walasek and K. Lukierska-Walasek, Phys. Rev. B **34**, 4962 (1986).
[45] R. Moessner and S. L. Sondhi, Phys. Rev. B **63**, 224401 (2001).
[46] S. Beauregard, Quantum Inf. Comput. **3**, 175 (2003).
[47] R. VanMeter and K. M. Itoh, Phys. Rev. A **71**, 052320 (2005).
[48] S. Istrail, Proceedings of the 32nd Annual ACM Symposium on Theory of Computing (ACM, New York, 2000), p. 87.