In: Condensed Matter Theories, Vol. 20
Editors: J.W. Clark, R.M. Panoff and H. Li, p. 485-496

ISBN 1-59454-989-3

PAIRING MODEL SIMULATION ON A QUANTUM COMPUTER

Mark S. Byrd[*], *Lian-Ao Wu*[†], *and Daniel A. Lidar*[†],

[*]Physics Department, Southern Illinois University,
Carbondale, Illinois 62901-4401

[†]Center for Quantum Information and Quantum Control
Department of Chemistry, University of Toronto, 80 st. St. George St.
Toronto, Ontario M5S 3H6, Canada

1. INTRODUCTION

Quantum computers show promise for solving some problems much more efficiently than classical computers [1]. There are three problems in particular which have received a great deal of attention in the literature. These are i) factoring large integers [2], ii) searching an unsorted data base [3], and iii) simulating a quantum system [4]. Of these the last may well be the most important to physical scientists for at least two reasons. First, for physical scientists, more efficient simulations of quantum systems could lead to many important advances in physics, chemistry and related sciences. Second, a fully scalable quantum computing system is not necessarily required in order to simulate some properties of physical systems which are practically out of reach of present-day classical machines. For example, 100 qubits could possibly simulate interesting molecular systems and noise need not be completely removed. Therefore quantum computing devices could be useful to physical scientists before the advent of a fully scalable quantum computer.

The speed-up of a quantum computing device may be attributed to the growth of the Hilbert space as one considers more single particle states or adds particles [1, 4, 5]. When Feynman first discussed this problem, he argued that bosons could be effectively simulated on a quantum computing device, but questioned whether fermions could [4]. Abrams and Lloyd showed that indeed they can be efficiently simulated and gave a method for the anti-symmetrization required [6]. Ortiz, et al. have extended this work to include more specific mappings and an explanation of how to avoid certain sign problems associated with classical simulations of fermionic systems [7]. Other discussions of quantum simulations range from general methods

of simulating one Hamiltonian from another to specific algorithms for simulating a particular physical quantity [8-20]. This work has sparked interest in the possibility of near future simulations on quantum systems.

In order to simulate one quantum system with another one must ensure that certain mappings can be made between the two. Specifically, one must be able to represent the states of the simulated system with states in the simulating one; the system must be properly initialized; appropriate unitary operations which mimic the evolution of the simulated system must be applied; and, finally, a measurement must be performed to determine an observable which represents a quantity of interest in the simulated system. These requirements are similar to, but not identical to, those needed for a general quantum computing device. In particular, a universal quantum computer, one that could perform any computation, one would require the ability to implement an arbitrary unitary operation on a set of qubits. For a simulation, one only requires the Hamiltonian, or corresponding operations, or propagators, be able to be simulated. Thus simulations may require lesser degrees of control of the quantum system.

In the following sections, some general objectives of quantum control are discussed and proposals for how these objectives might be achieved which appear in the quantum computing literature are given (Section 2). In Section 3, these principles are applied to the simulation of a pairing model Hamiltonian on an NMR quantum computing device in Section 4. Such models are used to describe superconductivity. Finally, in Section 4 some recent results in quantum control and some recent experiments are discussed which are directly relevant to pairing model simulations.

2. CONTROL OF QUANTUM SYSTEMS

Classical control theory benefits from the theory of differentiable manifolds and Lie group theory (See for example [21]). Given the use of these subjects in quantum mechanics (e.g., Berry phases [22]), it is not surprising that much of the control theory for classical mechanical systems also applies to quantum mechanical systems. A system which has its control determined before the experiment (referred to as open-loop control) can be described quite well with classical methods. However, it is clear that the measurement and subsequent feedback (referred to as closed-loop control) of *quantum* information can be quite different. Both closed- and open-loop control techniques have now been extended to quantum systems in various circumstances. In the interest of space and time no attempt will be made to produce a thorough citation list. For an introduction, there is the book Butkovskii and Samoilenko [23]. For a textbook treatment of control of atomic and molecular processes, see [24]. Early literature on quantum control includes the work of Butkovskii and Samoilenko [25] (and references therein), Belavkin [22], and Huang, et al. [26] (See also [27] and references therein.)

In this article the focus is on the control of quantum systems for the purposes of quantum computing and the simulation of quantum systems. Therefore a brief introduction to that portion of quantum control is given which applies directly to these problems. Finite systems of qubits (two-state systems) are used, thus only Hilbert spaces of finite dimension are considered here.

Let us define a control system according to the following. A state $\psi(t)$ of a quantum system evolves according to

$$i\frac{d}{dt}\psi(t) = \left[H_0 + \sum_{m=1}^{M} u_m(t)H_m\right]\psi(t), \tag{1}$$

where $\hbar = 1$, H_m, $m = 0, 1, ..., M$ are constant Hermitian operators (Hamiltonians) and the $u_m(t)$, $m = 1, ..., M$ are controllable parameters. There are many questions control theory seeks to answer. For example given an initial state, can a particular final state be reached? Since quantum computers will seek to answer unknown questions, a better statement for those considerations is, given an initial state, what are all the final states that can be reached? For simulations, one may ask, what are all the particular Hamiltonians that can be generated from a given set?

The most obvious way in which one would consider quantum control for quantum computing is the use of experimental controls to implement a universal gate set. This is a set of gates which, when combined, can implement any unitary transformation on a set of qubits. One universal gate set is the controlled not (CNOT) and the set of single qubit gates [28]. It was later shown, (a result from classical control theory [21]), that almost any two two-qubit Hamiltonians will provide a universal gate set [29, 30]. This result concerns controllability and, while useful, says nothing about the ability of an experimenter to create a unitary transformation on an appropriate time scale. The main obstacle to building a quantum computing device is noise in the quantum systems. Thus accuracy depends on the time required to implement transformations.

To produce a reliable quantum computing output, quantum error correcting codes were developed [31, 32]. Quantum error correcting codes are also quantum control scenarios where an error is detected and corrected. This closed loop control mimics the redundancy, or repetition codes from classical error correction. In this case, the control parameters are associated with transformations on a set of logical, or encoded qubits. These provide transformations which implement any error-free unitary transformation on the set of logical qubits. These methods are often referred to as active error correction methods since they actively detect and correct errors. (For a review of these and other error prevention methods for quantum computing see [33].)

Decoherence-free, or noiseless, subspaces [34-39] are often referred to as passive error prevention codes. In this case the information is again stored in logical qubits which are designed to take advantage of a system-bath interaction symmetry. If the logical qubits are chosen properly, the information will evolve without being adversely affected by the interaction with the bath. Since, after proper preparation, the system can evolve decoherence-free, or noiselessly, this is an open loop type of control.

Dynamical decoupling operations [40, 41], which are generalizations of average Hamiltonian theory used in nuclear magnetic resonance (NMR), are used to suppress errors. In an ideal limit of very fast "bang-bang" type operations, they can eliminate noises in a quantum systems' evolution. Such operations are also open-loop, using only a predetermined pulse set to eliminate noises. These controls are used to average away noises by repeated application of fast pulsing controls and have been shown to provide noise reduction in a variety of proposed experimental situations.

Finally, since not one of these methods has been completely effective at eliminating noise, combinations have been used to provide control techniques which use one or more of these techniques. These combinations may provide the most promising avenues yet for the reduction and/or elimination of noises in order to achieve quantum computing [33].

3. SIMULATING A BCS-TYPE MODEL

A universal quantum computing device could serve as a universal quantum simulator. However, one need not have a working quantum computing device with a large number of qubits in order to simulate some physical systems which are out of reach of a classical device. In addition, some parts of a system which are not necessarily controllable may have analogues in the system to be simulated. In fact, not only may we simulate closed quantum systems with a quantum simulator, it is possible to simulate open quantum system with a quantum simulator thus some noises need not be prevented.

The quantity simulated and its relation to the physical system may have special characteristics not inherent in the quantum simulation. Therefore care must be taken when comparing the simulated system to the one simulated. For example, the success dynamical decoupling controls discussed in the previous section are dependent upon the details of the system-bath interaction (see [33] for a review). This is something which may not be able to be accurately simulated except in special cases. However, for testing physical models, quantum simulation shows great promise.

In this section we will discuss the simulation BCS-like Hamiltonian and more general pairing models. However, we will limit the discussion to this particularly interesting case of quantum simulation and discuss the initialization, simulation and readout of the simulating system. Before such details are presented, we will first examine the salient features of the algorithm to simulate the BCS Hamiltonian which was first presented in [42] (see also [43]).

Pairing Hamiltonians are typically expressed in terms of fermionic or bosonic creation (annihilation) operators, $c_m^\dagger$ (c_m) and $b_m^\dagger$ (b_m), respectively, where $|m| = 1, 2, \ldots, N$ denotes all relevant quantum numbers. For example, the BCS Hamiltonian has the form:

$$H_{\mathrm{BCS}} = \sum_{m=1}^{N} \frac{\epsilon_m}{2}(n_m^F + n_{-m}^F) + \sum_{m,l=1}^{N} V_{ml}^{+} c_m^\dagger c_{-m}^\dagger c_{-l} c_l$$

where $n_{\pm m}^F \equiv c_{\pm m}^\dagger c_{\pm m}$ is the number operator, and the matrix elements

$$V_{ml}^{+} \equiv \langle m, -m|V|l, -l\rangle$$

(we impose no restriction on m, l) are real and can be calculated, e.g., for superconductors, in terms of the Coulomb force and the electron-phonon interaction [44, 45]. Pairs of fermions are labeled by the quantum numbers m and $-m$, according to the Cooper pair situation where paired electrons have equal energies but opposite momenta and spins: $m = (p, \uparrow)$ and $-m = (-\mathrm{p}, \downarrow)$. These are degenerate, time-reversed

partners whose energies are considered phenomenological parameters [46]. The same idea is applicable to nuclei, where effective pairings occur between nucleons in time-reversed partners [44, 45]. N is an effective state number, which equals the number of qubits in the algorithm below. E.g., in the case of metallic grains N is twice the the Debye frequency in units of the average level spacing (inversely proportional to volume of the grain). For nuclear pairing models, N could be the number of states in one or more major energy shells.

To make a connection to quantum algorithms we map the fermionic or bosonic operators to qubit operators. We denote the raising and lowering operators for the m^{th} qubit by the Pauli matrices $\sigma_m^{\pm}$, acting non-trivially only on the m^{th} qubit. A "number operator" is $n_m = (\sigma_m^z + 1)/2$, where $n_m = 1$ (0) if the m^{th} qubit is in state $|1\rangle$ ($|0\rangle$); $n = \sum_m n_m$ is the number of 1's in a computational basis state (a ket of a single bit-string), and will correspond, e.g., to the number of Cooper pairs in our applications below. The computational ground state $|0\rangle = |0_1 0_2 \cdots 0_N\rangle$ acts as a vacuum state: $n|0\rangle = \sigma_m^-|0\rangle = 0$. Now we can consider three generic pairing cases and map them to qubits. In each case we identify fermionic or bosonic operator pairs that satisfy the commutation rules of $sl(2) = \{\sigma_m^+, \sigma_m^-, \sigma_m^z\}$ (see [47] for details). These cases are: (i) *Fermionic particle-particle pairs* (e.g., Cooper pairs): $sl(2) = \{c_{-m}c_m, c_m^\dagger c_{-m}^\dagger, n_m^F + n_{-m}^F - 1\}$, provided $n_m^F = n_{-m}^F$ (a condition satisfied by H_{BCS}), and $|0\rangle = |0\rangle_F$. (ii) *Fermionic particle-hole pairs* (e.g., excitons): $sl(2) = \{c_{-m}^\dagger c_m, c_m^\dagger c_{-m}, n_m^F - n_{-m}^F\}$, provided $n_m^F + n_{-m}^F = 1$ and $|0\rangle = c_{-N}^\dagger \cdots c_{-2}^\dagger c_{-1}^\dagger |0\rangle_F$. (iii) *Bosonic 'particle-hole' pairs* (e.g., dual-rail photons in the optical quantum computer proposal [48]): $sl(2) = \{b_{-m}^\dagger b_m, b_m^\dagger b_{-m}, n_m^B - n_{-m}^B\}$, provided $n_m^B + n_{-m}^B = 1$ and $|0\rangle = b_{-N}^\dagger \cdots b_{-2}^\dagger b_{-1}^\dagger |0\rangle_B$. The three conditions above each restrict the dynamics to a different subspace of the entire Hilbert space. The conditions play the role of conserved quantities and only Hamiltonians that satisfy them preserve such subspaces.

We may now express H_{BCS} in terms of qubit operators. In fact, a more general Hamiltonian, that is applicable to all cases (i)-(iii) is:

$$H_{\text{p}} = \sum_{m=1}^{N} \frac{\varepsilon_m}{2}\sigma_m^z + \sum_{r=\pm}\sum_{l>m=1}^{N} \frac{V_{ml}^r}{2}(\sigma_m^x\sigma_l^x + r\sigma_m^y\sigma_l^y), \tag{2}$$

where $\varepsilon_m = \epsilon_m + V_{mm}^+$ and $V_{ml}^- = 0$ for H_{BCS}; l, m now denote both state indices and qubit indices. Further, in the BCS case the qubit state space

$$\mathcal{H}_{\mathcal{P}} = \text{Span}\{|0\rangle, \sigma_{\text{m}}^+|0\rangle, \sigma_{\text{l}}^+\sigma_{\text{m}}^+|0\rangle, \cdots\}$$

is mapped into a subspace of the total fermionic Hilbert space where $n_m^F = n_{-m}^F$. H_{BCS} conserves the total number operator n (the number of Cooper pairs). In terms of qubits, this means that the number of $|1\rangle$'s in a general N-qubit state is fixed by H_{BCS}. Thus the Hilbert space splits into invariant subspaces with dimension $\binom{N}{n}$ for fixed n. The problem is reduced to diagonalizing separate blocks of size $\binom{N}{n}$. For half-filled states in a system with $N = 100$, an exact solution could require

diagonalizing a $10^{29} \times 10^{29}$-dimensional matrix. Such a task is clearly unfeasible on a classical computer if one does not use approximations such as a perturbation calculation or DMRG methods.

We will develop an algorithm for the simulation of

$$U_p(k\tau) = \exp(-iH_p k\tau),$$

for $k = 1, ..., T/\tau$, where T is ... and τ is ... This will be a polynomial-time algorithm using, for concreteness, the NMR Hamiltonian. We will use the standard convention that a spin $+1/2$ state represents $|0\rangle$ and a -1/2 state will represent $|1\rangle$ so that our computational basis states (tensor products of zeros and ones) are represented by tensor products of the spin 1/2 states.

The algorithm is broken into five parts. (1) Initialize the system in a computational basis state $|x_n\rangle$ with a fixed n (number of $|1\rangle$s). (2) Evolve $|x_n\rangle$ quasi-adiabatically to $|\psi(0)\rangle = |g_n\rangle + \theta\,|e_n\rangle$ where $|g_n\rangle$ is the ground state of H_p, $|e_n\rangle$ is the first excited state, and $\theta \ll 1$. (3) Transform $|\psi(0)\rangle_0$ to $|\psi(0)\rangle = |g_{n,n\pm1}\rangle + \theta'|e_{n,n\pm1}\rangle$, a state that includes contributions from $n \pm 1$. (4) Implement $U_{\rm p}(t) = \exp(-iH_{\rm p}t)$ on $|\psi(0)\rangle$. (5) Measure. Repeat steps (1)-(5) while increasing t in step (5). The Hamiltonian simulation part of the algorithm has been broken up into parts to detail the quasi-adiabatic evolution.

Step (1)–Initialization: The initialization of a liquid state NMR system for the purposes of quantum information processing has been discussed in several places in the literature, notably [49, 50].

Step (2)– Adiabatic Evolution: Let 2Δ be the gap between the ground and the first excited states, and let $0 \le c(t) \le 1$, $c(0) = 0$, $c(T) = 1$, be a slowly varying function, i.e., $2\pi/T \ll 2\Delta$ (e.g., $c(t) = t/T$). Consider the time-ordered evolution $U_{\rm ad}(t) = \mathcal{T}\exp(-)\int_{\sqcup}^{\sqcup}\mathcal{H}(\int)\int)$ under a time-dependent Hamiltonian $H(t) = H_0 + c(t)H_I$. For sufficiently small τ this factors into a product

$$U_{\rm ad}(k\tau) \approx e^{-iH(k\tau)\tau} \cdots e^{-iH(2\tau)\tau} e^{-iH(\tau)\tau} + O(\tau^2), \tag{3}$$

where $\exp(-iH(j\tau)\tau) \approx \exp(-iH_0\tau)\exp(-ic(j\tau)H_I\tau)$ $(j = 1, ..., k)$, and now we choose times $\tau_{ml}(j)$ (for turning on $-J_l\sigma_l^z\sigma_m^z$) such that $2J_l\tau_{ml}() = |V_{ml}|\tau\, c(j\tau)$. Since $c(t)$ is slow, $U_{\rm ad}(k\tau)$ will represent an *adiabatic evolution*. Adiabatic evolution usually ensures that the system will be in an eigenstate of $H_{\rm p} = H(T)$ at $T = k\tau$, provided the initial state is in an eigenstate of H_0. Moreover, this will be a ground state $|g_n\rangle$ of $H_{\rm p}$ (a state with fixed n) if the initial state is the ground state of H_0 (a computational basis state $|x_n\rangle$). In order to probe the low-lying spectrum we may slightly relax the adiabatic condition $\pi/T \ll \Delta$, or $k \gg \pi/(\tau\Delta)$. This can be defined in terms of the adiabatic expansion where the first order constraint is the usual adiabatic assumption. Here we only wish to satisfy the second order condition [51].Then we obtain a state $|\psi(0)\rangle_0 \approx |g_n\rangle + \theta|e_n\rangle$ which contains a small component $|e_n\rangle$ of some of the low-lying excited states of $H_{\rm p}$ (with the same n).

Step (3)– Transformation: To probe states with different n we transform to

$$e^{-i\omega\sigma_\alpha^y}\,|\psi(0)_0\rangle \approx |g_n\rangle + \theta'\,|e_{n,n\pm1}\rangle\,,$$