Physics at the Perimeter Institute: Topics in Theoretical Physics & Quantum Simulation Research

- A Quick Guide -

Huan Q. Bui

Colby College

PHYSICS & MATHEMATICS
Statistics

Class of 2021

May 21, 2020

Preface

Greetings,

This guide is my notes from Perimeter Institute of Theoretical Physics Summer School, 2020. Topics include quantum information, thermodynamics, numerical methods, condensed matter physics, path integrals, and symmetries.

Enjoy!

Contents

Part 1

Quantum Information & Thermodynamics

The aim of this course is to understand the thermodynamics of quantum systems and in the process to learn some fundamental tools in Quantum Information. We will focus on the topics of foundations of quantum statistical mechanics, resource theories, entanglement, fluctuation theorems, and quantum machines.

1.1 Foundations of Quantum Statistical Mechanics - Entanglement

1.2 Resource Theories and Quantum Information

1.3 Quantum Thermal Operations

1.4 Fluctuation Theorems and Quantum Information

1.5 Quantum Thermal Machines

Part 2

Numerical Methods & Condensed Matter Physics

This course has two main goals: (1) to introduce some key models from condensed matter physics; and (2) to introduce some numerical approaches to studying these (and other) models. As a precursor to these objectives, we will carefully understand many-body states and operators from the perspective of condensed matter theory. (However, I will cover only spin models. We will not discuss or use second quantization.)

Once this background is established, we will study the method of exact diagonalization and write simple python programs to find ground states, correlation functions, energy gaps, and other properties of the transverse-field Ising model and XXZ model. We will also discuss the computational limitations of exact diagonalization. Finally, I will introduce the concept of matrix product states, and we will see how these can be used with algorithms such as the density matrix renormalization group (DMRG) to study ground state properties for much larger systems than can be studied with exact diagonalization.

2.1 Lecture 1

2.1.1 Introduction to many-particle states and operators

2.1.2 Introduction to Ising and XXZ models

2.1.3 Programming basics

2.1.4 Finding expectation values

2.2 Lecture 2: Exact diagonalization part 1

2.2.1 Representing models

2.2.2 Finding eigenstates

2.2.3 Energy gaps

2.2.4 Phase transitions

2.3 Lecture 3: Exact diagonalization part 2

2.3.1 Limitations of the method

2.3.2 Using symmetries

2.3.3 Dynamics

2.4 Lecture 4: Matrix product states part 1

2.4.1 Entanglement and the singular value decomposition

2.4.2 What is a matrix product state and why is it useful

2.5 Lecture 5: Matrix product states part 2

2.5.1 Algorithms for finding ground states using matrix product states: iTEBD

2.5.2 Algorithms for finding ground states using matrix product states: DMRG

Part 3

Path Integrals

The goal of this course is to introduce the path integral formulation of quantum mechanics and a few of its applications. We will begin by motivating the path integral formulation and explaining its connections to other formulations of quantum mechanics and its relation to classical mechanics. We will then explore some applications of path integrals.

3.1 Introduction to path integrals and the semi-classical limit

3.2 Propagator in real and imaginary time

3.3 Perturbation theory

3.4 Non-perturbative physics and quantum tunneling

3.5 Topology and path integrals

Part 4

Symmetries

The aim of this course is to explore some of the many ways in which symmetries play a role in physics. Well start with an overview of the concept of symmetries and their description in the language of group theory. We will then discuss continuous symmetries and infinitesimal symmetries, their fundamental role in Noether's theorem, and their formalization in terms of Lie groups and Lie algebras. In the last part of the course we will focus on symmetries in quantum theory and introduce representations of (Lie) groups and Lie algebras.

4.1 Lecture 1: Overview

4.1.1 Definition of symmetry

4.1.2 Elements of group theory

4.1.3 Examples

4.2 Lecture 2

4.2.1 Continuous and discrete symmetries

4.2.2 Infinitesimal symmetries

4.2.3 Noether's theorem

4.3 Lecture 3: Lie groups and Lie algebras

4.4 Lecture 4: Symmetries in quantum mechanics

4.5 Representation theory

Part 5

Research Project: Quantum Simulation

Quantum many-body physics on quantum hardware

Project description: Recently, there have been significant advances in several quantum computing platforms, including superconducting qubits and trapped ions. At this point, nontrivial quantum operations can be implemented on tens of qubits, and the frontier continues to expand. This project will explore how these emerging technologies can assist the realization and understanding of complex many-body quantum systems, regarding both static aspects such as ground state properties and dynamical aspects such as quantum chaos and thermalization. What new physics can we learn from near-term quantum computers? The ideal outcome is an interesting and realistic proposal that can be carried out in one of the quantum platforms (see for example the interplay between theoretical proposals and experiment).

5.1 Review: The density operator

We often use the language of state vectors in quantum mechanics. An alternative language is that of *density matrices/density operators*. We will use this language extensively in quantum information/quantum computation. First, we will introduce the formulation. Second, we look at some properties of the density operator. Finally, we look at an application where the density operator really shines – as a tool for describing *individual subsystems* of a composite quantum system.

5.1.1 Ensembles of Quantum States

The density operator language provides a convenient means for describing quantum systems whose state is not completely known. Suppose a quantum system is in one of the states $|\psi_i\rangle$ where i is an index, with respective probabilities p_i. We call $\{p_i, |\psi_i\rangle\}$ an *ensemble of pure states*.

Definition 5.1.1 (Density Operator)**.**

$$\boxed{\rho \equiv \sum_i p_i |\psi_i\rangle \langle\psi_i|} \tag{5.1}$$

Suppose we let a unitary $\mathcal{U}$ act on a closed system. If the system was initially in the state $|\psi_i\rangle$ with probability p_i then we get $\mathcal{U}|\psi_i\rangle$ with probability p_i. The density operator evolves as

$$\rho = \sum_i p_i |\psi_i\rangle \langle\psi_i| \stackrel{\mathcal{U}}{\to} \sum_i p_i \mathcal{U}|\psi_i\rangle \langle\psi_i|\mathcal{U}^\dagger = \mathcal{U}\rho\mathcal{U}^\dagger. \tag{5.2}$$

Suppose we have a measurement gate $\mathcal{M}_m$. If the intial state is $|\psi_i\rangle$ then the probability of getting result m is

$$p(m|i) = \langle\psi_i| \mathcal{M}_m^\dagger \mathcal{M}_m |\psi_i\rangle = \mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \langle\psi_i|\psi_i\rangle\right) \tag{5.3}$$

where we have used the identity:

$$\mathrm{tr}(A|\psi\rangle\langle\psi|) = \sum_i \langle i|A|\psi\rangle\langle\psi|i\rangle = \langle\psi|A|\psi\rangle. \tag{5.4}$$

With this, the total probability of measuring m is

$$p(m) = \sum_i p(m|i)p_i = \sum_i p_i \,\mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m |\psi_i\rangle\langle\psi_i|\right) = \mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho\right). \tag{5.5}$$

The state after the measurement is thus

$$|\psi_i^m\rangle = \frac{\mathcal{M}_m|\psi_i\rangle}{\sqrt{\langle\psi_i|\mathcal{M}_m^\dagger\mathcal{M}_m|\psi_i\rangle}}. \tag{5.6}$$

The corresponding density operator for this state is thus

$$\rho_m = \sum_i p(i|m) \left|\psi_i^m\right\rangle \left\langle\psi_i^m\right| = \sum_i p(i|m) \frac{\mathcal{M}_m \left|\psi_i\right\rangle \left\langle\psi_i\right| \mathcal{M}_m^\dagger}{\left\langle\psi_i\right| \mathcal{M}_m^\dagger \mathcal{M}_m \left|\psi_i\right\rangle}. \tag{5.7}$$

From probability theory we have $p(i|m) = p(m|i)p_i/p(m)$, so:

$$\rho_m = \sum_i p_i \frac{\mathcal{M}_m \left|\psi_i\right\rangle \left\langle\psi_i\right| \mathcal{M}_m^\dagger}{\mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho\right)} = \frac{\mathcal{M}_m \rho \mathcal{M}_m^\dagger}{\mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho\right)}. \tag{5.8}$$

Finally, suppose we have a quantum system in state ρ_i with probability p_i. The system might be described by the density matrix

$$\rho = \sum_i p_i \rho_i. \tag{5.9}$$

Here's why:

$$\rho = \sum_{i,j} p_i p_{ij} \left|\psi_{ij}\right\rangle \left\langle\psi_{ij}\right| = \sum_i p_i \rho_i \tag{5.10}$$

where we have used the definition

$$\rho_i = \sum_j p_{ij} \left|\psi_{ij}\right\rangle \left\langle\psi_{ij}\right|. \tag{5.11}$$

We call ρ the *mixture* of the states ρ_i with probabilities p_i. This concept of mixture comes up up repeatedly in the analysis of problems like quantum noise, where the effect of the noise is to introduce ignorance into our knowledge of the quantum state. A simple example is provided by the measurement scenario. Imagine that for some reason out record of the result m of the measurement was lost. We would have a quantum system in the state ρ_m with probability $p(m)$, but would no longer know the actual value of m. The state of such a quantum system would therefore by described by the density operator

$$\rho = \sum_m p(m)\rho_m = \sum_m \mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho\right) \frac{\mathcal{M}_m \rho \mathcal{M}_m^\dagger}{\mathrm{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho\right)} = \sum_m \mathcal{M}_m \rho \mathcal{M}_m^\dagger. \tag{5.12}$$

5.1.2 General Properties of the Density Operator

Theorem 5.1.1 (Characterization of density operators)**.** An operator ρ is the density operator to some ensemble $\{p_i, |\psi_i\rangle\}$ if and only if it statisfies the conditions

- **Trace condition:** $\mathrm{tr}(\rho) = 1$.
- **Positivity:** ρ is a positive operator, i.e., $\langle\varphi| \rho |\varphi\rangle \geq 0 \forall \varphi$.

With this definition we can reformulate the postulates of quantum mechanics in the language of density operators as

- **Postulate 1:** Associated to any isolated physical system is a Hilbert space known as the *state space* of the system. The system is completely described by its *density operator*, which is a positive operator ρ with trace one, acting on the state space of the system. If a quantum system is in the state ρ_i with probability p_i then the density operator for the system is $\sum_i p_i \rho_i$.

- **Postulate 2:** The evolution of a *closed* quantum system is described by a *unitary transformation*:

$$\rho' = \mathcal{U}\rho\mathcal{U}^\dagger. \tag{5.13}$$

- **Postulate 3:** Measurements are described by a collection $\{\mathcal{M}_m\}$ of *measurement operators*. The index m refers to the measurement outcomes that may occur. If the state of the quantum system is ρ immediately before the measurement then the probability that result m occurs is given by

$$p(m) = \operatorname{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho_m\right) \tag{5.14}$$

  and the state of the system after the measurement is

$$\frac{\mathcal{M}_m \rho \mathcal{M}_m^\dagger}{\operatorname{tr}\left(\mathcal{M}_m^\dagger \mathcal{M}_m \rho\right)}. \tag{5.15}$$

  The measurement operators satisfy the completeness equation

$$\sum_m \mathcal{M}_m^\dagger \mathcal{M}_m = \mathcal{I}. \tag{5.16}$$

- **Postulate 4:** The state space of a composite physical system is the tensor product of the state spaces of the component physical systems. E.g. the joint state of the total system can be written as $\rho_1 \otimes \cdots \otimes \rho_n$.

Theorem 5.1.2 (Pure states)**.** $\operatorname{tr}\left(\rho^2\right) \leq 1$. Equality occurs if and only if ρ is a pure state.

Theorem 5.1.3 (Unitary freedom in the ensemble for density matrices)**.** The sets $|\tilde{\varphi}_i\rangle$ and $\left|\tilde{\psi}_j\right\rangle$ generate the same density matrix if and only if they are unitarily similar, i.e.,

$$|\tilde{\varphi}_i\rangle = \sum_j u_{ij} \left|\tilde{\psi}_j\right\rangle \tag{5.17}$$

where the u_{ij} are entries of a unitary $\mathcal{U}$.

5.1.3 The Reduced Density Operator

Te deepest application of the density operator is as a descriptive tool for *subsystems* of a composite quantum system. Such a description is given by the *reduced density operator*.

Suppose we have physical systems A and B whose state is described by the density operator ρ^{AB}. The reduced density operator for A is then given by

$$\rho^A = \mathrm{tr}_B\left(\rho^{AB}\right) \tag{5.18}$$

where tr_B is the *partial trace* over system B, defined by

$$\mathrm{tr}_B\left(|a_1\rangle\langle a_2| \otimes |b_1\rangle\langle b_2|\right) \equiv |a_1\rangle\langle a_2|\,\mathrm{tr}\left(|b_1\rangle\langle b_2|\right) = |a_1\rangle\langle a_2|\langle b_2|b_1\rangle \tag{5.19}$$

where $|a_1\rangle, |a_2\rangle$ are any two vectors in the state space of A, and $|b_1\rangle, |b_2\rangle$ are any two vectors in the state space of B.

Example 5.1.1. To understand this construction, consider an example where $\rho^{AB} = \rho \otimes \sigma$ where ρ, σ describe A, B respectively. Then

$$\rho^A = \mathrm{tr}_B\left(\rho \otimes \sigma\right) = \rho\,\mathrm{tr}(\sigma) = \rho \tag{5.20}$$

as expected.

5.2 Quantum Computational Complexity: An Introduction

In this section we look at quantum computational complexity, as introduced in this paper by John Watrous at the IQC, the University of Waterloo.

5.2.1 Introduction

- **P**
- **NP**
- **BPP**
- **PP**
- **MA**
- **AM**
- **SZK**
- **PSPACE**
- **EXP**
- **NEXP**
- **PL**
- **NC**

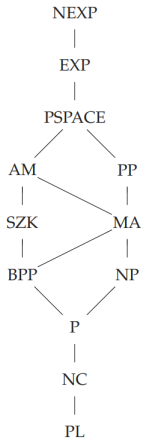

Figure 1: A diagram illustrating known inclusions among most of the classical complexity classes discussed in this paper. Lines indicate containments going upward; for example, AM is contained in PSPACE.

5.3 Quantum Simulation: A Primer

Here's what Feynman said in 1982:

"Can physics be simulated by a universal computer? [...] the physical world is quantum mechanical, and therefore the proper problem is the simulation of quantum physics [...] the full description of quantum mechanics for a large system with R particles [...] has too many variables, it cannot be simulated with a normal computer with a number of elements proportional to R [... but it can be simulated with] quantum computer elements. [...] Can a quantum system be probabilistically simulated by a classical (probabilistic, I'd assume) universal computer? [...] If you take the computer to be the classical kind I've described so far [..] the answer is certainly, No!"

5.3.1 Simulation in action

The heart of simulation is the solution of differential equations which capture the physical laws governing the dynamical behavior of a system. The goal is generally: given an initial state of the system, what is the state at some other time and/or position? Solutions are usually obtained by *approximating* the state with a digital representation, then *discretizing* the differential equation in space and time such that an iterative application of a procedure carries the state from the initial to the final conditions.

The error in this procedure is *bounded*, and known not to grow faster than some small power of the number of iterations. Furthermore, not all dynamical systems can be simulated efficiently: generally, only those systems which can be

described efficiently can be simulated efficiently.

Simulating quantum systems using classical computers is usually inefficient. The dynamical behavior of many simple quantum systems is governed the SE:

$$i\hbar\partial_t |\psi\rangle = \mathcal{H} |\psi\rangle . \tag{5.21}$$

We will set $\hbar = 1$. For real particles in space, we often have

$$i\partial_t \psi(x) = \left[-\frac{1}{2m}\partial_x^2 + V(x) \right] \psi(x). \tag{5.22}$$

The key challenge in simulating quantum systems is the *exponential* number of differential equations which must be solved. For 1 qubit evolving according to the SE, a system of 2 differential equations must be solved; for 2 qubits, 4 equations; and for n qubits, 2^n equations. Sometimes, insightful approximations can be made which reduce the effective number of equations involved, but there are many physically interesting quantum systems for which no such approximations are known.

There are many important quantum systems for which classical simulation is intractable. For example:

$$\text{Hubbard model: } \mathcal{H} = \sum_{k=1}^{n} V_0 n_{k\uparrow} n_{k\downarrow} + \sum_{k,j \text{ neighbors},\sigma} t_0 c_{k\sigma}^* c_{j\sigma}. \tag{5.23}$$

$$\text{Ising mode: } \mathcal{H} = \sum_{k=1}^{n} \vec{\sigma_k} \cdot \vec{\sigma}_{k+1}. \tag{5.24}$$

Quantum computers can efficiently simulate quantum systems for which there is no known efficient classical simulation.

5.3.2 The quantum simulation algorithm

The quantum simulation is concerned with the solution of $i\partial_t |\psi\rangle = \mathcal{H} |\psi\rangle$, which, for a time-independent $\mathcal{H}$, is just

$$|\psi(t)\rangle = e^{-i\mathcal{H}t} |\psi(0)\rangle . \tag{5.25}$$

$\mathcal{H}$ is in general difficult to exponentiate, but a good beginning is the first order solution

$$|(\psi(t + \Delta t))\rangle \approx (\mathcal{I} - i\mathcal{H}\Delta t) |\psi(t)\rangle . \tag{5.26}$$

This is tractable, but often is not very satisfactory.

Efficient approximations to higher oder is possible for many *classes* of Hamiltonians. For example, in most physical systems, the Hamiltonian can be written as a sum over many *local* interactions. For a system of n particles:

$$\mathcal{H} = \sum_{k=1}^{L} \mathcal{H}_k \sim \bigoplus_{k=1}^{L} \mathcal{H}_k \tag{5.27}$$

where each $\mathcal{H}_k$ acts on at most a constant c number of systems, and L is a polynomial in n. For instance, the terms $\mathcal{H}_k$ are often just 2-body interactions such as $X_i X_j$ and 1-boy Hamiltonians such as X_i. Note that both the Hubbard and Ising models have Hamiltonians of this form. There are sometimes additional global symmetry constraints such as particle statistics.

Note that although $e^{-i\mathcal{H}t}$ is difficult to compute, $e^{-i\mathcal{H}_k t}$ acts on a much smaller subsystem, and is straightforward to approximate using quantum circuits. However, because $[\mathcal{H}_j, \mathcal{H}_k] \neq 0$ in general (obviously, they don't commute in general), $e^{-i\mathcal{H}t} \neq \prod_k e^{-i\mathcal{H}_k t}$. So, we might want to ask how $e^{-i\mathcal{H}_k t}$ be useful in constructing $e^{-i\mathcal{H}t}$.

Remarks:

- If the $\mathcal{H}_i$'s commute, then we do have $e^{-i\mathcal{H}t} = \prod_k e^{-i\mathcal{H}_k t}$.
- The restriction of $\mathcal{H}_k$ to involve at most c particles *implies* that in the sum of the Hamiltonians, L is upper bounded by a polynomial in n.

The heart of quantum simulation algorithms is the following asymptotic approximation theorem:

Theorem 5.3.1 (Trotter formula)**.** Let A and B be Hermitian operators. Then for any real t,

$$\lim_{n\to\infty} \left(e^{-iAt/n} e^{iBt/n}\right) = e^{i(A+B)t}. \tag{5.28}$$

□

Of course, the theorem holds even if A and B don't commute. For now, we only consider the cases where A, B are Hermitian. But A, B can actually be generators of certain kinds of semigroups, which correspond to general quantum operations.

Using the theorem (and the proof of the theorem, which I won't show here), we can get

$$e^{i(A+B)\Delta t} = e^{iA\Delta t} e^{iB\Delta t} + \mathcal{O}(\Delta t^2). \tag{5.29}$$

Similarly,

$$e^{i(A+B)\Delta t} = e^{iA\Delta t/2} e^{iB\Delta t} e^{iA\Delta t/2} + \mathcal{O}(\Delta t^3). \tag{5.30}$$

5.3.3 The algorithm

- **Inputs:** (1) A Hamiltonian $\mathcal{H} = \sum_k \mathcal{H}_k$ acting on an N-dimensional system, where each $\mathcal{H}_k$ acts on a small subsystem of size independent of N, (2) an initial state $|\psi_0\rangle$, of the system at $t = 0$, (3) a positive, non-zero accuracy δ, and (3) a time t_f at which the evolved state is desired.
- A state $\left|\tilde{\psi}(t_f)\right\rangle$ such that

$$\left|\left\langle\tilde{\psi}(t_f)\right| e^{-i\mathcal{H}t_f} \left|\psi_0\right\rangle\right|^2 \geq 1 - \delta. \tag{5.31}$$

- Choose a representation such that the state $\left|\tilde{\psi}\right\rangle$ of $n = \text{poly}(\log N)$ qubits approximates the system and the operators $e^{-\mathcal{H}_k \Delta t}$ have efficient quantum circuit approximations. Selection an approximation method and Δt such that the expected error is acceptable, construct the corresponding quantum circuit $\mathcal{U}_{\Delta t}$ fo the iterative step, and do the following:
  - Initialize state: $\left|\tilde{\psi}_0\right\rangle \leftarrow |\psi_0\rangle\,; j = 0$.
  - Iterative update: $\rightarrow \left|\tilde{\psi}_{j+1}\right\rangle = \mathcal{U}_{\Delta t}\left|\tilde{\psi}_j\right\rangle$.
  - Loop: $\rightarrow j = j + 1$; go to the second step until $j\Delta t \geq t_f$.
  - Final result: $\rightarrow \left|\tilde{\psi}(t_f)\right\rangle = \left|\tilde{\psi}\right\rangle$.

Theorem 5.3.2 (Baker-Campbell-Hausdorf formula)**.**

$$e^{(A+B)\Delta t} = e^{A\Delta t} e^{B\Delta t} e^{-(1/2)[A,B]\Delta t^2} + \mathcal{O}(\Delta t^3). \tag{5.32}$$

5.3.4 An illustrative example

We have seen the case where the Hamiltonian is a sum of local interactions. However, this is not a fundamental requirement. Efficient quantum simulations are possible even for Hamiltonians which act non-trivially on all or nearly all parts of a large system.

Consider the Hamiltonian

$$\mathcal{H} = Z_1 \otimes Z_2 \otimes \cdots \otimes Z_n \tag{5.33}$$

which acts on an n qubit system. This can be simulated efficiently. What we desire is a simple quantum circuit which implements $e^{-i\mathcal{H}\Delta t}$, for arbitrary values of Δt. A circuit that does this is given by

Even though the Hamiltonian involves all the qubits in the system, it does so in a *classical* manner: the phase shift applied to the system is $e^{-i\Delta t}$ if the *parity* of the n qubits in the computational basis is even. else the phase shift would

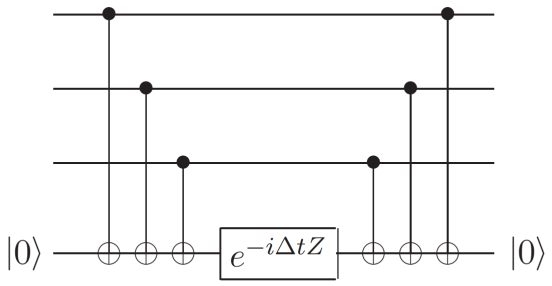

Figure 4.19. Quantum circuit for simulating the Hamiltonian $H = Z_1 \otimes Z_2 \otimes Z_3$ for time Δt.

be $e^{i\Delta t}$. Thus, a simple simulation of $\mathcal{H}$ is possible by first classical computing the parity, then applying the appropriate phase shift conditioned on the parity, then uncomputing the parity.

Extending the same procedure allows us to simulate more complicate extended Hamiltonians. Specifically, we can efficiently simulate any Hamiltonian of the form

$$\mathcal{H} = \bigotimes_{k=1}^{n} \sigma_{c(k)}^{k} \tag{5.34}$$

where $\sigma_{c(k)}^{k}$ is a Pauli matrix (or identity) acting on the kth qubit, with $c(k) \in \{0, 1, 2, 3\}$ specifying one of $\{I, X, Y, Z\}$. The qubits upon which the identity acts can be disregarded, and X or Y terms can be transformed by single qubit gates to Z operations. This leaves a Hamiltonian of the form given above, which can be efficiently simulated.

Using this procedure allows us to simulate a wide class of Hamiltonians containing terms which are not local. It is possible to simulate a Hamiltonian of the form $\mathcal{H} = \sum_{k=1}^{L} \mathcal{H}_k$ where we only require that the individual $\mathcal{H}_k$ have a tensor product structure, and that L is polynomial in the total number of particles n.

5.3.5 Perspectives on quantum simulation

Quantum simulation is similar to classical methods, but differs fundamentally in many aspects. Each iteration of the quantum algorithm must completely replace the old state with a new one and there is no way to obtain any nontrivial information from an intermediate step without changing the algorithm (due to the no-cloning theorem). Furthermore, the final measurement must be chosen to provide the desired result because it collapses the entire state.

Another difficult problem is the simulation of equilibration processes. A system with Hamiltonian $\mathcal{H}$ in contact with an environment at temperate T

will come to thermal equilibrium in a state known as the *Gibbs* state,

$$\rho_{\text{therm}} = \frac{e^{-\mathcal{H}/k_B T}}{\mathcal{Z}} \equiv \frac{e^{-\mathcal{H}/k_B T}}{\text{tr}\left(e^{-\mathcal{H}/k_B T}\right)} \tag{5.35}$$

where of course $\mathcal{Z}$ is the partition function. We can check that $\text{tr}(\rho) = 1$. For now, we don't know how a quantum computer can simulate this.

Indistinguishability of particles also places a constraint on the state vector of a system which manifests itself in two ways: bosons and fermions. The state vector of a system of bosons remains unchanged under permutation of any two constituents. System of fermions experience a sign change in their state vector under interchange of any two constituents. Both kinds of systems can be simulate efficiently on a quantum computer.

5.4 Problems in Quantum Simulations and Quantum Algorithms

In this section, we will be looking at the following review on quantum algorithms for quantum chemistry and materials science.

5.4.1 Quantum Simulation: An Overview

Introduction

The interest in quantum computing for quantum simulations of molecules and materials stems from the fact that in many cases, the chemistry and physics of molecules and materials is best described using quantum mechanics. In the worst case, quantum simulation is exponentially hard on classical computers. Quantum computers have advantages over classical computers, but these advantages are problem-specific.

As we have seen before, the natural problem to solve on a quantum computer is the time evolution of a quantum system given some initial state:

$$i\partial_t \left|\psi(t)\right\rangle = \mathcal{H}\left|\psi(t)\right\rangle . \tag{5.36}$$

This problem, as we have pointed out, is of polynomial cost on a quantum computer, which can offer exponential speedup over classical computers. However, it is necessary to prepare the initial state, which may be difficult. In particular, preparing a low-energy state may be challenging, which naturally leas to considering other important problems:

$$\text{Ground state:} \quad \mathcal{H}\left|\psi_0\right\rangle = E_0\left|\psi_0\right\rangle ; \quad E_0 = \min_{\left|\psi\right\rangle} \left\langle\psi\right| \mathcal{H}\left|\psi\right\rangle \tag{5.37}$$

$$\text{Thermal averages:} \quad \langle\mathcal{A}\rangle = \frac{\operatorname{tr}\left(\mathcal{A}e^{-\beta\mathcal{H}}\right)}{\operatorname{tr}\left(e^{-\beta\mathcal{H}}\right)} \tag{5.38}$$

Ground state determination lies in complexity class QMA, a class of problems *not known to be efficiently solvable in general on a quantum computer*. This means that thermal averages cannot in general be computed efficiently on a quantum computer, since in the limit of zero temperature, this problem reduces to ground-state determination.

To understand quantum advantage in chemistry, condensed matter physics, and quantum materials science, we must be guided by actual empirical data in the form of numerical and theoretical experiments with quantum algorithms and quantum devices on simulation problems of interest. This requires progress, of course.

Current quantum architectures

5.4.2 Simulation challenges in molecular and materials science

Quantum chemistry

Quantum molecular spectroscopy

Chemical quantum dynamics

Correlated electronic structure in materials

Dynamical quantum effects in materials

5.4.3 Challenges for quantum algorithms in quantum simulation

Overview of algorithms

Qubit representation of many-body systems

Quantum algorithms for ground and excited states

Preparing ground states along a prescribed path

Variational state preparation and variational quantum eigensolver

Excited states

Phase estimation

Quantum algorithms for time evolution

5.4.4 Reading out results

Equal-time measurements

Dynamical properties and Green's functions

5.5 Some Models in Quantum Statistical & Condensed-Matter Physics

5.5.1 The Hubbard Model

5.5.2 The Ising Model

5.6 Quantum Phase Estimation (QPE)

One of the most important applications of the Quantum Fourier Transform is the Quantum Phase Estimation. Phase estimation is the key for many quantum algorithms. Suppose a unitary operator $\mathcal{U}$ has an eigenvector $|u\rangle$ with eigenvalue $2^{2\pi i\varphi}$, where the value of φ is unknown. The goal of the phase estimation algorithm is to estimate $0 \leq \varphi \leq 1$ with high probability within additive error ϵ. The algorithm uses $\mathcal{O}(\log(1/\epsilon))$ qubits and $\mathcal{O}(1/\epsilon)$ controlled-$\mathcal{U}$ operations.

To perform the estimation we assume that we have available *oracles* capable of preparing the state $|u\rangle$ (the eigenstate) and performing the controlled-$\mathcal{U}^{2j}$ operation, for suitable nonnegative integers j. Since preparing eigenstates is not necessarily easy, the phase estimation algorithm is not a complete algorithm by itself. Rather, we should view it as a "subroutine" or a "module."

The QPE uses two registers. The first register contains t qubits initially in $|0\rangle$. The choice of t depends on two things: the number of digits of accuracy we wish to have in our estimate of φ, and with what probability we wish the phase estimation procedure to be successful. The second register begins in the eigenstate $|u\rangle$, and contains as many qubits as necessary to store $|u\rangle$.

QPE occurs in two main stages. First, the circuit applies an $\mathcal{H}$ to each of the t qubits. Then, the circuit applies controlled-$\mathcal{U}$ operators on the second register, with $\mathcal{U}$ raised to successive powers of two. This stage looks like the following: Recall that immediately after $\mathcal{H}^{\otimes t}$ the state of the upper register looks like

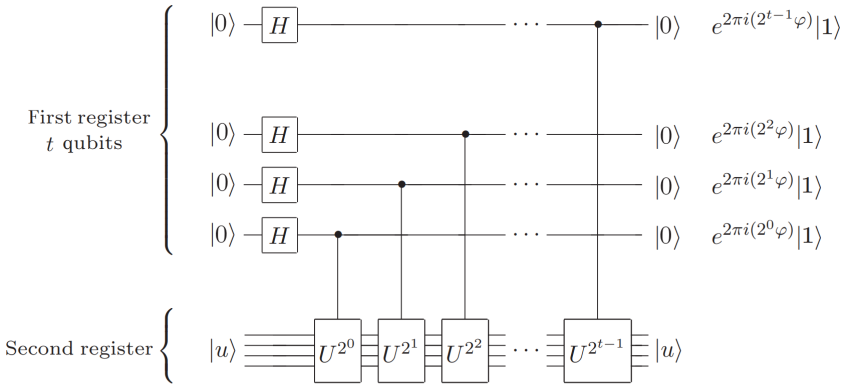

Figure 5.2. The first stage of the phase estimation procedure. Normalization factors of $1/\sqrt{2}$ have been omitted, on the right.

Figure 5.1: Source: Mike and Ike

$$\mathcal{H}|0\rangle = \frac{1}{2^{1/2}}\left(|0\rangle + |1\rangle\right) \tag{5.39}$$

The controlled-$\mathcal{U}$ acts on $|u\rangle$ only if the control qubit is in $|1\rangle$. With this, we see that the final state of the first register is given by

$$\begin{aligned}
&\frac{1}{2^{t/2}}\left(|0\rangle + e^{2\pi i 2^{t-1}\varphi}|1\rangle\right)\otimes\left(|0\rangle + e^{2\pi i 2^{t-2}\varphi}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{2\pi i 2^{0}\varphi}|1\rangle\right)\\
&\quad = \dots\\
&\quad = \frac{1}{2^{t/2}}\sum_{0\le k<2^t} e^{2\pi i\varphi k}|k\rangle_t\,. \qquad (5.40)
\end{aligned}$$

Example 5.6.1. For $t = 2$, we have

$$\begin{aligned}
&\frac{1}{2}\left(|0\rangle + e^{4\pi i\varphi}|1\rangle\right)\otimes\left(|0\rangle + e^{2\pi i\varphi}|1\rangle\right)\\
&\quad = \frac{1}{2}\left(|0\rangle_2 + e^{1\cdot 2\pi i\varphi}|1\rangle_2 + e^{2\cdot 2\pi i\varphi}|2\rangle_2 + e^{3\cdot 2\pi i\varphi}|3\rangle_2\right)\\
&\quad = \frac{1}{2^{2/2}}\sum_{0\le k<2^2} e^{2\pi i\varphi\cdot 2}|k\rangle_2 \qquad (5.41)
\end{aligned}$$

□

Note that we're leaving out the qubits carrying the $|u\rangle$ state because we don't do anything to it (because $|u\rangle$ is an eigenstate of $\mathcal{U}$). We should also note that the effect of the sequence of controlled-$\mathcal{U}$ operations is to take $|j\rangle|u\rangle$ to $|j\rangle\mathcal{U}^j|u\rangle$. We should also note that this does not depend on the fact that $|u\rangle$ is an eigenstate of $\mathcal{U}$.

In the second state, the algorithm applies the inverse quantum Fourier transform on the first register (which is just the QFT, applied in reverse order of operations). This can be done efficiently in $\mathcal{O}(t^2)$ steps, as we have seen the Quantum Information notes. Since the input in this case is the maximal superposition in the t-vector space, the output of the inverse QFT is going to be a single state: $|\tilde{\varphi}\rangle$, which we hope is approximately $|\varphi\rangle$. We will see how close $|\tilde{\varphi}\rangle$ can get to $|\varphi\rangle$.

The third and final stage of the algorithm is to read out the state of the first register by doing a measurement in the computational basis.

Example 5.6.2. To make sure we see what is going on, suppose φ is expressed exactly in t qubits, as $\varphi = 0, \varphi_1\varphi - 2\dots\varphi_t$ (note that $\varphi \in [0,1]$). Then the state of the input register after the first stage is given by

$$\frac{1}{2^{t/2}}\left(|0\rangle + e^{2\pi i 0.\varphi_t}|1\rangle\right)\otimes\left(|0\rangle + e^{2\pi i 0.\varphi_{t-1}\varphi_t}|1\rangle\right)\otimes\cdots\otimes\left(|0\rangle + e^{2\pi i 0.\varphi_1\varphi_2\dots\varphi_t}|1\rangle\right). \qquad (5.42)$$

Applying the inverse QFT to this, we see that output state from the second state is the product state $|\varphi_1\dots\varphi_t\rangle$. A measurement in the computational basis gives φ exactly. □

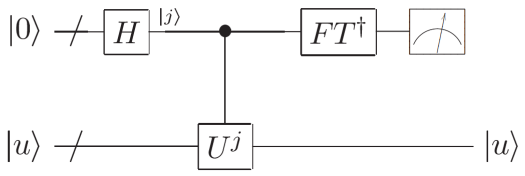

Figure 5.3. Schematic of the overall phase estimation procedure. The top t qubits (the '/' denotes a bundle of wires, as usual) are the first register, and the bottom qubits are the second register, numbering as many as required to perform U. $|u\rangle$ is an eigenstate of U with eigenvalue $e^{2\pi i\varphi}$. The output of the measurement is an approximation to φ accurate to $t - \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil$ bits, with probability of success at least $1 - \epsilon$.

Figure 5.2: Source: Mike and Ike

Schematically, the algorithm for QPE looks like the following:
In summary, what we basically did was using the inverse QFT to get

$$\frac{1}{2^{t/2}} \sum_{0 \le k < 2^t} e^{2\pi i \varphi j} |j\rangle |u\rangle \to |\tilde{\varphi}\rangle |u\rangle \tag{5.43}$$

where $|\tilde{\varphi}\rangle$ is a good estimator for φ when measured.

5.6.1 Performance and Requirements

What happens when φ cannot be written exactly with a t-bit expansion? We will now show that the procedure we described will produce a good approximation to φ with high probability.

Let $b \in [0, 2^t - 1]$ an integer be given such that $b/2^t = 0.b_1 b_2 \cdots_t$ is the best t bit approximation to φ which is less than φ, i.e.,

$$0 \le \delta \equiv \varphi - b/2^t \le 2^{-t}. \tag{5.44}$$

(This is getting pretty close to an analysis proof – because it *is* an analysis proof.) Applying the inverse QFT to the state immediately after the first stage gives

$$\begin{aligned} \mathcal{F}^{-1}\left[\frac{1}{2^{t/2}} \sum_{0 \le k < 2^t} e^{2\pi i \varphi k} |k\rangle\right] &= \frac{1}{2^t} \sum_{0 \le k,l < 2^t} e^{\frac{-2\pi i k l}{2^t}} e^{2\pi i \varphi k} |l\rangle \\ &= \frac{1}{2^t} \sum_{0 \le k,l < 2^t} \left(e^{2\pi i (\varphi - l/2^t)}\right)^k |l\rangle \end{aligned} \tag{5.45}$$

by definition. Note that when φ can be expressed using exactly t bits we are reduced to the simpler version of the problem described above.

Let α_l be the amplitude of $|(b+l) \mod 2^t\rangle$, i.e.,

$$\alpha_l \equiv \frac{1}{2^t} \sum_{0 \le k < 2^t} \left(e^{2\pi i(\varphi - (b+l)/2^t)} \right)^k. \tag{5.46}$$

This is a partial sum of a geometric series. It is straightforward to show that

$$\alpha_l = \frac{1}{2^t} \left(\frac{1 - e^{2\pi i(2^t\varphi - (b+l))}}{1 - e^{2\pi i(\varphi - (b+l)/2^t)}} \right) = \frac{1}{2^t} \left(\frac{1 - e^{2\pi i(2^t\delta - l)}}{1 - e^{2\pi i(\delta - l/2^t)}} \right). \tag{5.47}$$

Suppose the outcome of the final measurement is m. We want to bound the probability that $|m-b| > \epsilon$ (where ϵ is the tolerance), i.e., we don't want the probably that m is far from b to be large. Well, the probability of observing such an m is given by

$$P(|m-b| > \epsilon) = \sum_{-2^{t-1} < l \le -(\epsilon+1)} |\alpha_l|^2 + \sum_{e+1 \le l \le 2^{t-1}} |\alpha_l|^2. \tag{5.48}$$

For any real θ, we have $|1 - \exp(i\theta)| \le 2$, so

$$|\alpha_l| \le \frac{2}{2^t \left| 1 - e^{2\pi i(\delta - l/2^)} \right|}. \tag{5.49}$$

Further, provided $-\pi \le \theta \le \pi$, we have $|1 - \exp(i\theta)| \ge 2|\theta|/\pi$. Also, when $-2^{t-1} < l \le 2^{t-1}$ we have $-\pi \le 2\pi(\delta - l/2^t) \le \pi$, so we have

$$|\alpha_l| \le \frac{1}{2^{t+1}(\delta - l/2^t)}. \tag{5.50}$$

With this we have

$$\begin{aligned}
p(|m-b| > \epsilon) &\le \frac{1}{4} \left(\sum_{l=-2^{t-1}+1}^{-(\epsilon+1)} \frac{1}{(l - 2^t\delta)^2} + \sum_{l=\epsilon+1}^{2^{t-1}} \frac{1}{(l - 2^t\delta)^2} \right) \\
&\le \frac{1}{4} \left(\sum_{l=-2^{t-1}+1}^{-(\epsilon+1)} \frac{1}{(l-1)^2} + \sum_{l=\epsilon+1}^{2^{t-1}} \frac{1}{(l-1)^2} \right), \quad 0 \le 2^t\delta < 1 \\
&\le \frac{1}{2} \sum_{l=\epsilon}^{2^{t-1}-1} \frac{1}{l^2} \\
&\le \frac{1}{2} \int_{\epsilon-1}^{2^{t-1}-1} \frac{1}{l^2}\, dl \\
&= \frac{1}{2(\epsilon-1)}.
\end{aligned} \tag{5.51}$$

Suppose we wish to approximate φ to an accuracy 2^{-n}, i.e., we choose $\epsilon = 2^{t-n} - 1$. By using $t = n + p$ qubits, we see that the probability of obtaining an

approximation correct to this accuracy is at least

$$1 - \frac{1}{2(\epsilon - 1)} = 1 - \frac{1}{2(2^p - 2)}. \tag{5.52}$$

So, to successfully obtain φ accurate to n bits with probability of success at least $1 - \epsilon$ we choose

$$t = n + \left\lceil \log\left(2 + \frac{1}{2\epsilon}\right)\right\rceil. \tag{5.53}$$

Now here's a catch: to use QPE, we need the eigenstate $|u\rangle$. What if we don't know how to prepare such a state? Suppose that we prepare some other state $|\psi\rangle$ in place of $|u\rangle$. We can write

$$|\psi\rangle = \sum_u c_u |u\rangle \tag{5.54}$$

where $|u\rangle$ are the eigenstates of U. Suppose the eigenstate $|u\rangle$ has the eigenvalue $e^{2\pi i \varphi_u}$. Running the QPE on $|\psi\rangle$ will give

$$|\psi\rangle \overset{\text{QPE}}{\rightarrow} \sum_u c_u |\tilde{\varphi_u}\rangle |u\rangle, \tag{5.55}$$

where $\tilde{\varphi_u}$ is an estimation for φ_u. Reading out the first register gives us a good approximation for φ_u, where u is chosen at random with probability $|c_n|^2$. This procedure allows us to avoid preparing a possibly unknown eigenstate, but at the cost of introducing some additional randomness into the algorithm. It turns out that if t is chosen according to the minimization procedure above, then the probability for measuring φ_u accurate to n bits at the conclusion of the phase estimation algorithm is at least $|c_u|^2(1 - \epsilon)$.

5.6.2 The algorithm

QPE solves a problem which is both non-trivial and interesting from a physical point of view. It allows us to estimate the eigenvalue associated to a given eigenvector of a unitary operator. It turns our that other interesting problems can be reduced to phase estimation, which makes QPE particularly useful.

Here's the algorithm:

- **Inputs:** (1) An oracle which performs a controlled-$\mathcal{U}^j$ operation, (2) an eigenstate $|u\rangle$ of $\mathcal{U}$ with eigenvalue $e^{2\pi i \varphi_u}$, and (3) $t = n + \lceil \log(2 + 1/2\epsilon)\rceil$ qubits initialized to $|0\rangle$.
- **Outputs:** An n-bit approximation $\tilde{\varphi}_u$ of φ_u.
- **Runtime:** $\mathcal{O}(t^2)$ operations (due to th inverse QFT) and one call to controlled-$\mathcal{U}^j$ oracle. Succeeds with probability at least $1 - \epsilon$.

- **Procedure:**
  - Initial state: $|0\rangle |u\rangle$
  - Create superposition: $\rightarrow (1/2^{t/2}) \sum_{0 \leq j < 2^t} |j\rangle |u\rangle$.
  - Apply the oracle:
    $\rightarrow (1/2^{t/2}) \sum_{0 \leq j < 2^t} |j\rangle \mathcal{U}^j |u\rangle = (1/2^{t/2}) \sum_{0 \leq j < 2^t} e^{2\pi i j \varphi_u} |j\rangle |u\rangle$
  - Apply inverse QFT: $\rightarrow |\tilde{\varphi}_u\rangle |u\rangle$.
  - Measure the first register: $\rightarrow \tilde{\varphi}_u$.

The algorithm has a wide range of applications, including order-finding and factorization.

5.7 Quantum Expectation Estimation (QEE)

In this and the following section we will be looking at two quantum algorithms that are alternatives/improvements to QPE.

5.8 Quantum Variational Eigensolver (QVE)

5.9 Quantum Approximate Optimization Algorithm (QAOA)

5.10 Quantum Variational Factorization (QVF)

5.11 Variational Thermal Quantum Simulation via Thermofield Double States

5.11.1 Thermofield Double States (TFD)

5.12 Generation of TFD and Critical Ground States with a Quantum Computer

5.13 Efficient variational simulation of non-trivial quantum states