Lecture 2: Efficient quantum simulations
Reading: Universal quantum simulators, S. Lloyd, Science (1996).

1 Efficiency of quantum simulation

Let's begin with a discussion of the goal of our application of a QC for quantum simulation. For this application to be useful, we need to be able to implement the simulation efficiently, meaning that to simulate a given Hamiltonian we should need only a polynomial number of gates as a function of the simulation time. If we need exponentially many gates to perform the time evolution, the ability to store the system state in an exponentially growing Hilbert space would not be very helpful because we wouldn't be able to apply operators to the state!

Consider the most straightforward transient simulation with a time-independent Hamiltonian. In that case, the wavefunction at time t is related to the initial wavefunction at $t = 0$ by a propagator $U(t)$ as:

$$|\psi(t)\rangle = U(t)\,|\psi(t=0)\rangle = e^{-iHt}\,|\psi(t=0)\rangle \tag{1.1}$$

Consider for now that the gate set we have available consists of single qubit rotations and a two-qubit entangling gate of some type (e.g. CNOT, CZ,...it turns out that nearly any two-qubit gate along with single qubit gates form a universal set [1, 2]). Any unitary can be decomposed into $d(d-1)/2$ two-qubit operations, where d is the dimension of the Hilbert space acted on by the unitary (N&C p189). Here we see a problem: as we know, the dimension d grows exponentially with the number of particles or quantum degrees of freedom in the system, with the result that the gate depth required to implement the propagator exactly grows exponentially.

So it is clear that exactly simulating a given Hamiltonian is in general not possible. However, Lloyd described how we can obtain an efficient simulation by relaxing the requirement for exact simulation and by employing knowledge of the physical nature of most Hamiltonians we care about. In other words, a quantum system can be programmed to simulate the behavior of arbitrary quantum systems for which dynamics are determined by local interactions. The programming is accomplished by inducing interactions between variables of the simulator that imitate interactions with the variables of the actual system. Further, the simulator can perform this imitation to any desired accuracy using polynomial resources and in linear time with the real time of the system.

Let's make this discussion more precise by recalling the definition of simulation from the last lecture. In Lloyd's definition, a simulation is where one system is made to mimic another by appropriately adjusting variables. Quantum systems can be controlled by external laser pulses, magnetic fields, and so on. An experimenter can turn on and off Hamiltonian terms governing interactions between quantum degrees of freedom with these controls, exactly as occurs in e.g. cold atom experiments with light pulses.

Therefore, with this control we can create any unitary operator as $U = \exp(-iAt)$ where A is a Hermitian operator. We string these operations together to get the overall unitary we want by multiplying individual unitaries and using the Baker-Campbell-Hausdorff (BCH) formula:

$$e^X e^Y = e^{X+Y+[X,Y]/2+\ldots} \tag{1.2}$$

where $[.,.]$ denotes the commutator. As discussed above, the number of terms in the product required to produce an arbitrary unitary operator exactly is $O(d^2)$. Therefore, if we let different

quantum variables interact to make a logic gate, we can construct arbitrary unitary transforms of a 2^N dimensional Hilbert space, but not efficiently. This decomposition is therefore not useful.

However, physical systems that we care about in physics are not arbitrary but have a very important constraint: locality. Locality is a strict required of relativity, but in non-relativistic settings there is also an important constraint known as the Lieb-Robinson bound, which loosely states that information in local quantum spin systems propagates with a finite group velocity [3]. Therefore, all physical systems of which I am aware cannot contain arbitrary interactions but rather must consist of local terms. More precisely, a k-local Hamiltonian consists of:

$$H = \sum_{i=1}^{l} H_i \tag{1.3}$$

where each H_i acts on a space of dimension m_i with at most k variables (for example, in a quantum spin system at most k sites). Any Hamiltonian with local interactions can be represented in this way. Now for any Hamiltonian that is a sum of terms, we can approximate the BCH expansion by retaining only the first order terms as:

$$e^{iHt} \approx (e^{iH_1 t/n} ... e^{iH_l t/n})^n + \sum_{i>j} [H_i, H_j] t^2/2n \tag{1.4}$$

Therefore, we can approximate the actual unitary evolution by repeatedly applying the individual unitaries above to any desired accuracy by making n large enough. This expansion is known as Trotterization and is accurate to $O(\Delta t^2)$. Higher order expansions are also known that have better error scalings [4].

The final consideration is how many terms compose the Hamiltonian, for if this number diverges exponentially we still have a problem. Here is where we use the locality constraint that enforces that all physical Hamiltonians must operate only locally within a certain neighborhood. In practice, many relevant Hamiltonians operator only over nearest-neighbor sites - consider the generic Heisenberg and Ising spin models. As an example, consider a Hamiltonian that operates over only two sites. There can be at most $\binom{N}{2}$ terms, where N is the number of particles (or quantum degrees of freedom but for simplicity we just say particles). This number grows polynomially with N and so we have an efficient simulation that can be carried out on a quantum computer. Therefore, we can generally say that local Hamiltonians, which comprise essentially all physically relevant Hamiltonians, can indeed be simulated to within arbitrary accuracy using a quantum computer.

In a bit more detail, say that each H_j operates on a Hilbert space of dimension m_j (e.g. 2 for a spin-1/2 system). Therefore the number of operations required for this term is $O(m_j^2)$. Each local operator is simulated n times, and so the total number of operations required for U is

$$N_{ops} \approx n \sum_{i=1}^{l} m_i^2 \leq nlm^2 \tag{1.5}$$

where m is the maximum of the m_i. To obtain an overall error $< \epsilon$, the error for an elementary operation must be $< \epsilon/nlm^2$.

Now, a simulation is efficient if simulating a system with N variables takes time that is polynomial in N. Therefore, as long as $l = l(N)$ is polynomial in N, the simulation is indeed efficient. For

local Hamiltonians, that requirement is indeed met! For instance, for nearest-neighbor interactions, $l \sim N$.

Let's examine the simulation time required in more detail. The error in the Trotter approximation for the entire unitary evolution of time t is $O(t^2/n)$ as from the BCH formula. To get $O(\epsilon)$ error, we need $n \sim t^2/\epsilon$ steps. However, each coherent operation that is actually implemented on the QC is implemented in real time and therefore takes time $\sim t/n$. Therefore, the total time for the overall simulation is simply the number of steps times the time per step, or $n \times t/n \sim t$, or linear!

In practice, we often can do better than the worst-case scaling described above. For instance, for 2-local nearest-neighbor terms H_l, terms that are more than one site away do commute and so the decomposition in Eq. 1.4 is exact. Further, gates that do not share common sites can be applied simultaneously, leading to parallelization such that the number of terms depend only on the locality of the interaction, not the number of particles N! In that case, the time required to simulate the system does not depend on N, while the classical cost is still exponential in N.

As an example, a 2-local nearest-neighbor Hamiltonian can be divided as $H = H_{even} + H_{odd}$ where even and odd denote gates over sites starting with an even or odd number, respectively, and all the gates with the even or odd sum commute and can be applied simultaneously. In that case, each Trotter step takes only two levels of gates to even and odd sites, and the two level decomposition does not depend on the number of particles.

The conclusion is that quantum computers can indeed efficiently simulate physical quantum systems - not all systems, but for all physical systems that respect locality of interactions.

References

[1]S. Lloyd, "Almost any quantum logic gate is universal.", Phys Rev Lett **75**, 346–349 (1995).

[2]D. Deutsch, A. Barenco, and A. Ekert, "Universality in quantum computation", Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences **449**, 669–677 (1995).

[3]E. H. Lieb and D. W. Robinson, "The finite group velocity of quantum spin systems", Commun.Math. Phys. **28**, 251–257 (1972).

[4]M. Suzuki, "Fractal decomposition of exponential operators with applications to many-body theories and monte carlo simulations", Physics Letters A **146**, 319–323 (1990).