

# Quantum Computing (CST Part II)

## Lecture 11: Application 2 of QFT / QPE: Quantum Chemistry

*Feynman's 1982 conjecture, that quantum computers can be programmed to simulate any local quantum system, is shown to be correct.*

**Seth Lloyd**

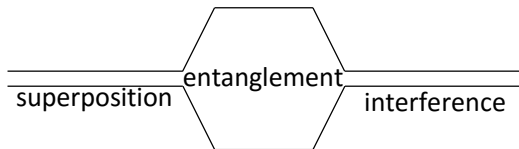
## Resources for this lecture

**Nielsen and Chuang p204-211** covers quantum simulation.

For the interested student **Quantum computational chemistry** (McCardle *et al*, <https://arxiv.org/abs/1808.10402>) provides a comprehensive overview of quantum chemistry – but much of this is well beyond the scope of this course.

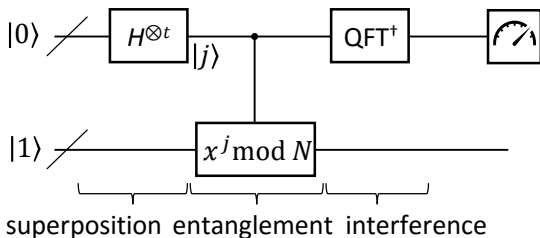
# The typical structure of quantum algorithms

Quantum algorithms typically consist of three stages:



1. Initially we must put the state in a **superposition** – however if the quantum algorithm is to be efficient overall then this cannot incur too many operations.
2. Next, we use **entanglement** to “search” a vast solution space.
3. Finally, we must extract some data which is truly a “global” property of the state, but which is compact in some sense (otherwise it may take an exponential amount of time to extract this data). To do so, we **interfere** the final entangled superposition.

## Shor's algorithm as an example of this typical structure



1. We put the first register in the superposition  $\frac{1}{\sqrt{2^t}} \sum_{x \in \{0,1\}^t} |x\rangle$ , which can be achieved with a single layer of Hadamard gates (the second register's initial state,  $|1\rangle$ , can also easily be prepared).
2. The controlled- $U^{2^j}$  operations are entangling.
3. Finally, the inverse QFT interferes the entangled state in such a way that we can extract the phase.

# The computational power of entanglement

It is not fully understood precisely how entanglement apparently yields exponential speed-ups, but one important factor is that **entangled spaces are exponentially larger than their unentangled counterparts in some sense**. To see this, consider three alternatives:

1. **A classical  $n$ -bit binary number**,  $x$ , thus  $|x\rangle$  will be a vector of length  $2^n$  with exactly one 1. So we have  $2^n$  possibilities for  $|x\rangle$ .
2. **A  $n$ -qubit product state**,  $|x\rangle$ , in which each qubit is either  $|0\rangle$ ,  $|1\rangle$  or the superposition  $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ . So we have  $3^n = (2^n)^{\log_2 3}$  possibilities for  $|x\rangle$ . That is,  $|x\rangle$  is a *product state* such that, when expressed as a superposition (over the standard basis states), each term in the superposition has a co-efficient of  $+1$  (and the entire sum is normalised accordingly).
3. **All  $n$ -qubit states** such that when expressed as a superposition (over the standard basis states), each term in the superposition has a co-efficient of  $+1$  (and the entire sum is normalised accordingly). A state,  $|x\rangle$  can thus be such that  $x$  is any binary string  $\{0,1\}^{2^n}$  (appropriately normalised), i.e., the set of all  $2^n$ -bit binary vectors, so we have  $2^{(2^n)}$  possibilities for  $|x\rangle$ .

# Quantum simulation

Suppose we want to know the state,  $|\psi_t\rangle$  of a quantum system at a time,  $t$  in the future, given its Hamiltonian,  $\mathbf{H}$  and its current state,  $|\psi_0\rangle$ , then we must solve the Schrödinger equation:

$$|\psi_t\rangle = e^{-i\mathbf{H}t} |\psi_0\rangle$$

However, exponentiating  $\mathbf{H}$  is not, in general, tractable. Moreover, first-order approximations,  $|\psi_{t+\Delta t}\rangle \approx (I - i\mathbf{H}\Delta t) |\psi_t\rangle$  are usually unsatisfactory.

## Quantum simulation (continued)

One helpful property of many real quantum systems of interest is that they can be decomposed:

$$H = \sum_{k=1}^K H_k$$

where  $K$  is sufficiently small, and the physical nature of the system is such that each  $H_k$  can be exponentiated (this is because the physical system is dominated by “local” few-body interactions). So we have that:

$$|\psi_t\rangle = e^{-iHt} |\psi_0\rangle = e^{-i\sum_k H_k t} |\psi_0\rangle$$

However, notice we are dealing with *matrix exponentiation*, and in general:

$$e^{-i\sum_k H_k t} \neq \prod_k e^{-iH_k t}$$

so it appears that we cannot directly use the fact that  $H_k$  can be exponentiated tractably.

# Trotterisation

However,  $H_k$  can still be useful for simulating the quantum system, because of the **Trotter formula**, which is at the heart of quantum simulation:

$$\lim_{n \rightarrow \infty} \left( e^{iH_1 t/n} e^{iH_2 t/n} \right)^n = e^{i(H_1 + H_2)t}$$

To see this, consider that, by definition:

$$\begin{aligned} e^{iH_1 t/n} &= I + \frac{1}{n} iH_1 t + \mathcal{O}\left(\frac{1}{n^2}\right) \\ \implies e^{iH_1 t/n} e^{iH_2 t/n} &= I + \frac{1}{n} i(H_1 + H_2)t + \mathcal{O}\left(\frac{1}{n^2}\right) \\ \implies \left( e^{iH_1 t/n} e^{iH_2 t/n} \right)^n &= I + \sum_{l=1}^n \binom{n}{l} \frac{1}{n^l} (i(H_1 + H_2)t)^l + \mathcal{O}\left(\frac{1}{n}\right) \end{aligned}$$



## Proof of the Trotter formula (continued)

Noticing that:

$$\binom{n}{l} \frac{1}{n^l} = \frac{n!}{l!(n-l)!} \frac{1}{n^l} = \frac{n(n-1)(n-2)\cdots(n-l+1)}{n^l} \frac{1}{l!} = \frac{1}{l!} \left(1 + \mathcal{O}\left(\frac{1}{n}\right)\right)$$

We get that:

$$\begin{aligned} \lim_{n \rightarrow \infty} \left( e^{iH_1 t/n} e^{iH_2 t/n} \right)^n &= I + \lim_{n \rightarrow \infty} \sum_{l=1}^n \frac{(i(H_1 + H_2)t)^l}{l!} \left(1 + \mathcal{O}\left(\frac{1}{n}\right)\right) + \mathcal{O}\left(\frac{1}{n}\right) \\ &= \lim_{n \rightarrow \infty} \sum_{l=0}^n \frac{(i(H_1 + H_2)t)^l}{l!} \\ &= e^{i(H_1 + H_2)t} \end{aligned}$$

Following analysis similar to that above, with a small finite time  $\Delta t$ , we get that:

$$e^{i(H_1 + H_2)\Delta t} = e^{iH_1 \Delta t} e^{iH_2 \Delta t} + \mathcal{O}(\Delta t^2)$$

Therefore, if we divide up the duration of the evolution into sufficiently short intervals, we can accurately approximate the overall evolution by evolving each  $H_k$  in turn.

# Quantum simulation algorithm

For simulating  $|\tilde{\psi}_t\rangle \approx |\psi_t\rangle = e^{iHt} |\psi_0\rangle$ :

1. Initialise  $|\tilde{\psi}_0\rangle = |\psi_0\rangle$ ;  $j = 0$
2.  $|\tilde{\psi}_{j+1}\rangle \leftarrow U_{\Delta t} |\tilde{\psi}_j\rangle$
3.  $j \leftarrow j + 1$ ; if  $j\Delta t < t$  goto step 2
4. Output  $|\tilde{\psi}_t\rangle = |\tilde{\psi}_j\rangle$

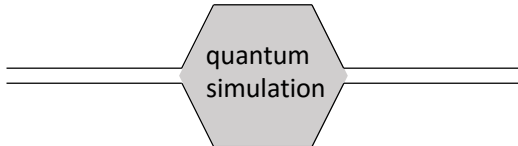
Where  $H = \sum_{k=1}^K H_k$ , and:

$$U_{\Delta t} = e^{iH_1\Delta t} e^{iH_2\Delta t} \dots e^{iH_K\Delta t}$$

for  $\Delta t$  chosen to be suitably small to achieve the overall desired accuracy.

## Quantum simulation: a solution in search of a problem?

As Feynman asserted, simulation of quantum systems is classically intractable in a fundamental way: because of entanglement it may take an exponential amount of classical memory to even represent the state. So the significance of the discovery of efficient quantum simulation on quantum computers should not be understated. However, from an algorithmic point of view, this really only corresponds to the middle of the double-necked bottle:



In order to find useful application, we need to use quantum simulation to solve a problem with a compact input and output.

# Quantum chemistry

Quantum chemistry provides one such application of quantum simulation, the general set-up being:

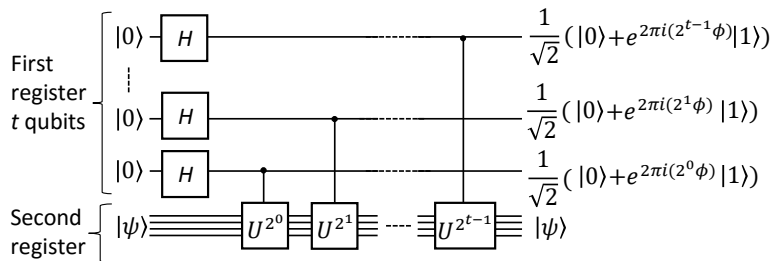
- A system Hamiltonian,  $H_s$  is encoded as a qubit Hamiltonian  $H_q$ .
- An important property in computational chemistry is the *ground state energy*, which can be found by estimating the minimum eigenvalue phase of the unitary  $U = e^{-Ht}$ : **that is, a compact output.**
- We defer discussion of a suitable input state, and instead consider a general input,  $|\psi\rangle$ , expressed as a superposition of eigenvectors of  $U$ :

$$|\psi\rangle = \sum_i a_i |u_i\rangle$$

i.e., noting that the eigenvectors,  $|u_i\rangle$ , of the unitary form an orthonormal basis.

# Quantum phase estimation for quantum chemistry

We can now see how quantum simulation enables QPE for quantum chemistry, as the controlled-unitaries represent evolutions of the Hamiltonian for discrete amounts of time, which we have seen can be efficiently simulated.



# Quantum phase estimation algorithm

Thus QPE for ground-state energy estimation can be summarised thus:

1. Initialise the second register in the state  $|\psi\rangle = \sum_i a_i |u_i\rangle$
2. Perform QPE, using the quantum simulation algorithm to evolve  $U$  accordingly.
3. Because of linearity, the final state is therefore:

$$\sum_i a_i |\text{bin}(\phi_i)\rangle |u_i\rangle$$

where  $|\text{bin}(\phi_i)\rangle$  is a binary estimate of the phase if the  $i$ th eigenvalue.

So we can see that there is a  $|a_0|^2$  probability of collapsing into the desired state  $|u_0\rangle$  (and so reading out the eigenvalue phase associated with the ground-state energy). This in turn tells us that **we should prepare the initial state such that its sufficiently dominated by  $|u_0\rangle$** . There are various ways to do this, and later in the course we will study one of them: adiabatic state preparation.

## Near-term quantum chemistry

Performing QPE for quantum chemistry requires a *full-scale fault-tolerant quantum computer*, however even in the absence of such a device the principle that quantum simulation on classical computers is intractable still holds. Therefore much current research concerns **hybrid quantum-classical algorithms** which aim to execute only **shallow-depth quantum circuits**, in which an **unmanageable amount of error is not expected to occur**.

The most famous and promising of these hybrid algorithms is the **variational quantum eigensolver**.

# The variational quantum eigensolver

VQE relies on the Rayleigh-Ritz variational principle:

$$\langle \psi(\theta) | H | \psi(\theta) \rangle \geq E_0$$

where  $|\psi(\theta)\rangle$  is a quantum state parameterised by  $\theta$  (suitably initialised), and  $E_0$  is the lowest energy eigenvalue. This implies we can find the ground-state energy by finding the value of parameters that minimise  $\langle \psi(\theta) | H | \psi(\theta) \rangle$ .

VQE then iterates the following:

1. Run a shallow-depth quantum circuit  $U(\theta) : |0\rangle \rightarrow |\psi(\theta)\rangle$  to prepare  $|\psi(\theta)\rangle$ .
2. Measure to get  $E(\theta)$  – an estimate of the lowest energy eigenvalue.
3. Perform classical optimisation to update the parameter values  $\theta$ .

Fig. 6 of **Quantum computational chemistry** (McCardle *et al*) gives a good illustration of the operation of the variational quantum eigensolver.



# Summary

In this lecture we have covered:

- The typical structure of quantum algorithms and the computational significance of entanglement.
- Quantum simulation, including Trotterisation.
- Quantum phase estimation for ground-state energy estimation in quantum chemistry.
- The variational quantum eigensolver: a near-term quantum chemistry algorithm.