Why we think a quantum computer could help solve a travelling salesperson problem

Newcastle computing teaching presentation

Nicholas Chancellor

April 12, 2023

(011) (010)

(111)

(000)

(001) (110)

Structure of talk

Setting the scene:

- What is quantum mechanics (very briefly)
- Quantum computing in 1996 (when this was shown)
- P versus NP

Unstructured search problem:

- Best classical algorithm
- Grover's quantum search

Limitations and applications:

- Issues with using directly
- Hybrid quantum/classical algorithms

Wrap-up, questions, and discussions

Quantum mechanics

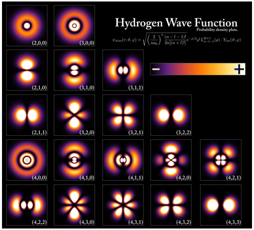


Image: public domain taken from wikimedia commons

I won't teach you all of quantum mechanics in 20 minutes, but...
there are some key facts you should know

- A linear theory $\rightarrow$ (possibly very big) matrices and vectors
- Vectors of probability amplitudes $\rightarrow$ proportional to square root of probability
- Unlike probability, amplitudes add or subtract, not just add*

*They are generally complex numbers, involving $i = \sqrt{-1}$, but that isn't important for this talk

Travelling salesperson

Prototypical example of a "hard" optimisation problem

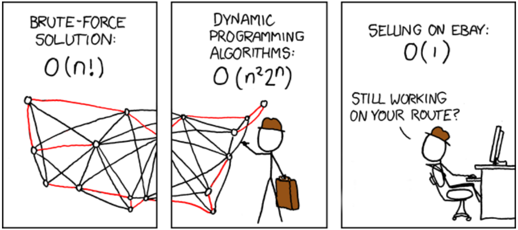


Image: XKCD comic 399 created by Randall Monroe https://xkcd.com CC attribution non-commercial (slightly modified to remove swearing)

- Our salesperson has to visit n cities, but can do so in any order
- $n!$ (valid) routes, clever algorithms can do better
- Time to find the exact solution scales (exponentially) badly for all known algorithms

Quantum computing today

Core idea:

Build a better computer by taking advantage of quantum mechanics

Image: wikimedia commons CC share alike attribution, uploaded by IBM research

- Devices exist now, lots of room for improvement, but very exciting
- We can do some experiments

... but how did we get here?

Quantum Computing in 1996

Image: copyright BBC, used under fair use

- No hope of building a device anytime soon
- Some ideas of how it might work, but is it worth it?

Any justification for building one had to be purely theoretical!

Need to show advantage for something important

Proving quantum is better for travelling salesperson?

Image: public domain taken from wikimedia commons

Any justification has to be purely theoretical!

- Need to know what the best possible classical algorithm is
- Show quantum can do better

- Oops, we don't know what the best classical is
- Hard (exponential scaling) classical optimisation problems not proven to exist
  - This is a deep question in CS: $P \stackrel{?}{=} NP$

A problem where we **do** know the best classical can do

(011) (010)

(111)

(000)

(001) (110)

Unstructured search problem:

- We can check answers with an "oracle" either tells us "right" or "wrong", but no info on how close
- No clever algorithmic tricks, either guess or check all
  - Both approaches scale like N, the number of possible solutions we could check

Quantum search in a time proportional to N^p, where $p < 1$?

Quantum search conceptually

Recall: quantum amplitudes scale as the square root of probability

- The amplitude of the solution $\frac{1}{\sqrt{N}}$ rather than $\frac{1}{N}$ for probability
- Use interference (the way amplitudes can cancel) in a clever way to exploit this fact
- End up in the solution with a high probability after a number of steps proportional to $\sqrt{N}$

Quantum search mathematically

Key trick: high degree of symmetry means we can reduce to a two dimensional subspace

- $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$ corresponds to the solution
- $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ corresponds to an unweighted sum of every state **except** the solution
- Operations consist of 2x2 matrices operating in this space

Diffusion: a quantum version of random guessing

- Amplitude goes from all states to all other states
- Total probability (sum of squares of amplitudes) must always sum to 1
- Can be compiled to quantum “‘gates”, can explain during questions if interest

Written in our two-dimensional subspace, diffusion operation becomes

$$D = \begin{pmatrix} -1 + \frac{2}{N} & \frac{-2\sqrt{N-1}}{N} \\ \frac{-2\sqrt{N-1}}{N} & 1 - \frac{2}{N} \end{pmatrix}$$

Minus signs are needed to guarantee probabilities add to 1

Applying diffusion to a quantum state

- Consider we start in a general state $\begin{pmatrix} a \\ b \end{pmatrix}$, then applying diffusion gives us:

$$\begin{pmatrix} -1+\frac{2}{N} & \frac{-2\sqrt{N-1}}{N} \\ \frac{-2\sqrt{N-1}}{N} & 1-\frac{2}{N} \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} a(\frac{2}{N}-1) - 2b\frac{\sqrt{N-1}}{N} \\ b(1-\frac{2}{N}) - 2a\frac{\sqrt{N-1}}{N} \end{pmatrix}$$

- Exercise: show that if $a = \sqrt{\frac{N-1}{N}}$ and $b = \sqrt{\frac{1}{N}}$, than $D\begin{pmatrix} a \\ b \end{pmatrix} = -\begin{pmatrix} a \\ b \end{pmatrix}$

For $N \gg 1$ this is approximately $\begin{pmatrix} -a - b\frac{2}{\sqrt{N}} \\ b - a\frac{2}{\sqrt{N}} \end{pmatrix}$

- Addition in complement of the solution, subtraction in solution

Adding a way to tell which state is the solution

- To make our guessing useful, we need to do something to tell us when we got the answer "right"
- We ask our "oracle" to multiply by -1 if we have found the solution, we call this the "marking" operation

$$m = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Applying both

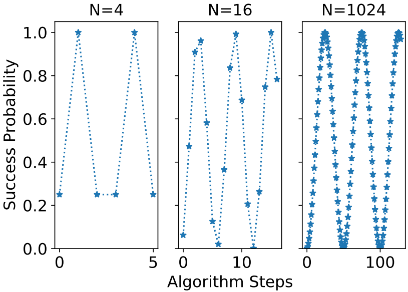
$$Dm \begin{pmatrix} a \\ b \end{pmatrix} = \begin{pmatrix} a(\frac{2}{N} - 1) + b\frac{2\sqrt{N-1}}{N} \\ b(\frac{2}{N} - 1) - a\frac{2\sqrt{N-1}}{N} \end{pmatrix} \approx \begin{pmatrix} -a + b\frac{2}{\sqrt{N}} \\ -b - a\frac{2}{\sqrt{N}} \end{pmatrix}$$

This operation **adds** to the solution and **subtracts** elsewhere

The bottom line

- Every application of *Dm* increases the amplitude to be in the solution by an amount proportional to $\frac{1}{\sqrt{N}}$
- Therefore applying this operation a number of times proportional to $\sqrt{N}$ gives us an amplitude of order 1

Classical computers are very good at multiplying 2x2 matrices (and even were in 1996), here are some examples*



*it is a good exercise to reproduce these, hint probability is the (absolute value of) amplitude squared

Stepping back

We just showed that a quantum computer can search faster than a classical computer ever could*...

What does this mean in practice?

- Motivation that they might be fundamentally better at tasks like solving travelling salesperson
- Might even be directly useful as part of a bigger algorithm

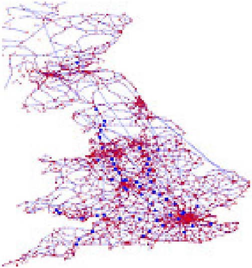


Image: public domain taken from wikimedia commons

*The algorithm we demonstrated is called "Grover's algorithm"

Probably don't want to just apply directly

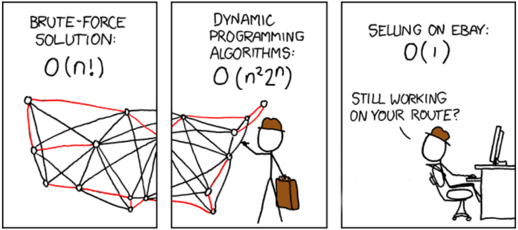


Image: XKCD comic 399 created by Randall Monroe https://xkcd.com CC attribution non-commercial (slightly modified to remove swearing)

- Often classical algorithms can scale better than $\sqrt{N}$ for real (structured) problems
- The dynamical programming algorithm given in this example scales better than $\sqrt{n!}$ for example*
- ...also the issue of how to know if a route is the shortest

*The algorithm being referenced here in particular (Held-Karp) does have some unfortunate scaling in memory usage, but the larger point still stands

Use within classical algorithms

- Find classical algorithm with stages which look like unstructured search and replace with quantum search
- Example Montanero, Phys. Rev. Research 2, 013056 (2020):
  - Classical branch-and-bound solved their problem* in a time which scales as $2^{0.451n}$, where n is the number of variables
  - Showed that the quantum version would scale in $2^{0.226n}$
- Took a classical algorithm which was already faster than unstructured search, and got an additional speedup

---

*Not travelling salesperson but a different hard optimisation problem

The importance of encoding

Classical computers are already very good...

- Only worth using quantum if we are searching over a large number of configurations
  - A laptop can easily solve an optimisation problem where $N \approx 1,000,000$ just by checking every possibility
- Need an efficient encoding, physical size of device scales as $n \propto \log(N)$

(011) (010)

(111)

(000)

(001) (110)

- Usual approach is to encode into quantum bits, $N = 2^n$, but other ways exist
- Algorithm could be encoded into a quantum circuit, on quantum bits, but that is beyond the scope of this lecture

The big picture

The theory presented here motivated the use of quantum computers for one important type of problem

Other key early factors:

- Quantum error correction was shown to be theoretically feasible $\rightarrow$ hardware doesn't have to be "perfect"
- Quantum computers could factor numbers very fast (Shor's algorithm)
- Simulating quantum systems is very hard classically, quantum computers could be good for simulating quantum systems

Summary and key points

- Early justification for quantum computers had to be purely theoretical
- Unstructured searching provided a way to do this

- Quantum amplitudes scale as the square root of probability
- Unlike probabilities they can be positive or negative (or involve $\sqrt{-1}$)
- Controlling how these add or cancel is at the heart of quantum algorithms

- The search algorithm we showed is not useful directly but...
- It can be used within other algorithms
- Quantum computing is only useful with a good encoding