Solving Graph Problems with Photonic Quantum Computing

Alexander Makarovskiy

August 2024

My Internship Project

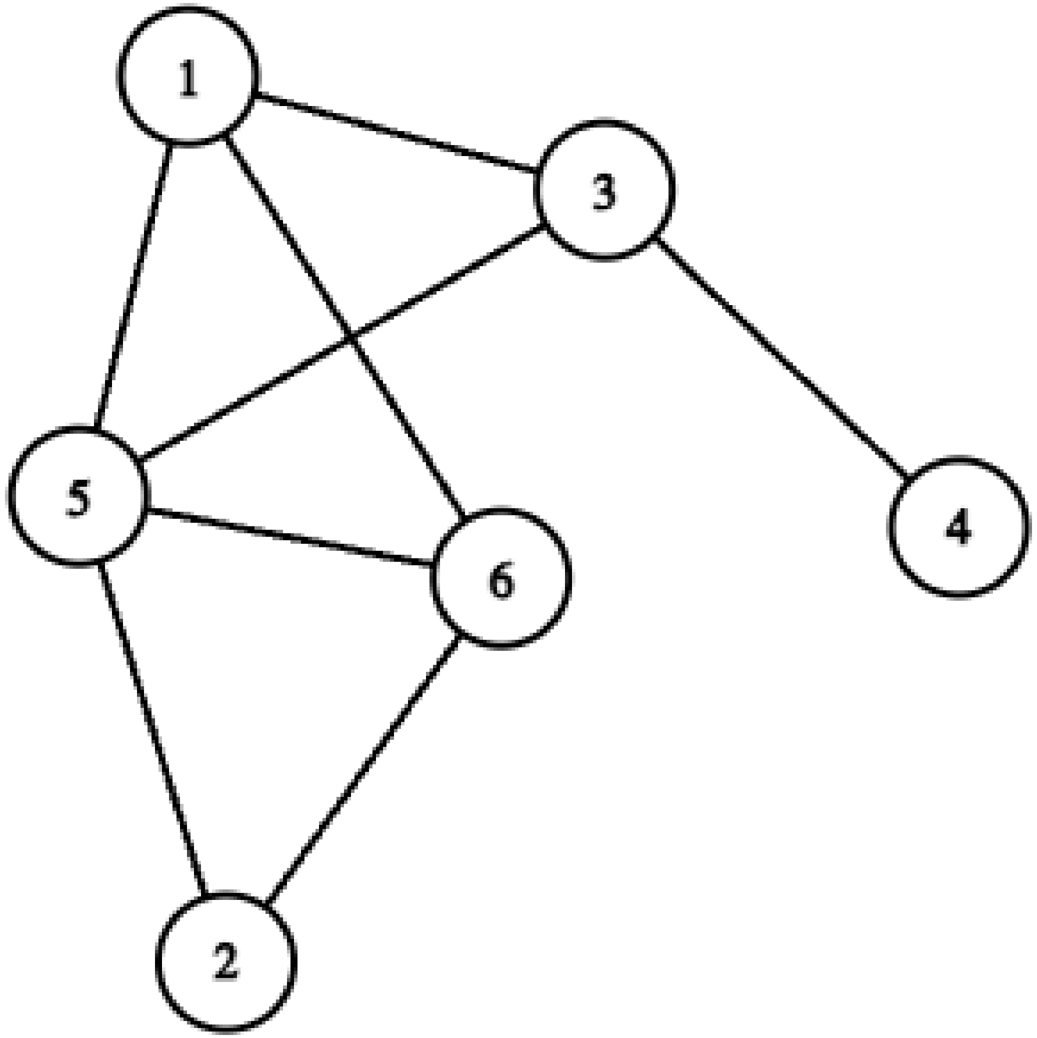
I studied the viability of using a type of photonic quantum computer to solve problems in graph theory

Some problems in graph theory are very computationally complex: it takes a classical computer a long time to solve them

There have been propositions about how we could perhaps solve these problems quicker using quantum computers

I spent the summer investigating some of these propositions

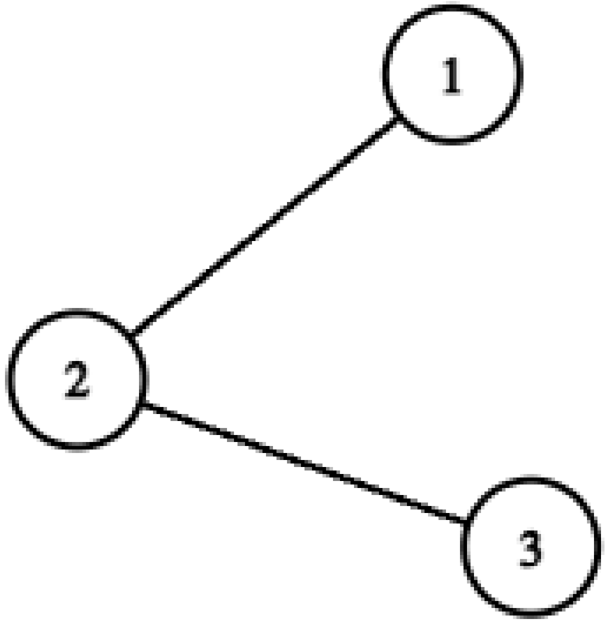
A Quick Intro. to Graph Theory



A Quick Intro. to Graph Theory

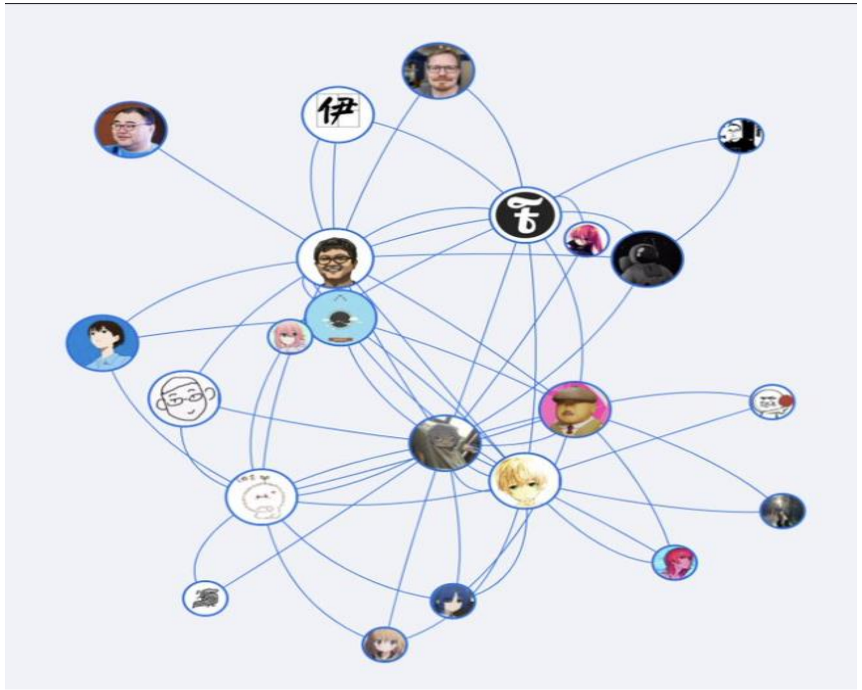
$$A = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

Adjacency Matrix

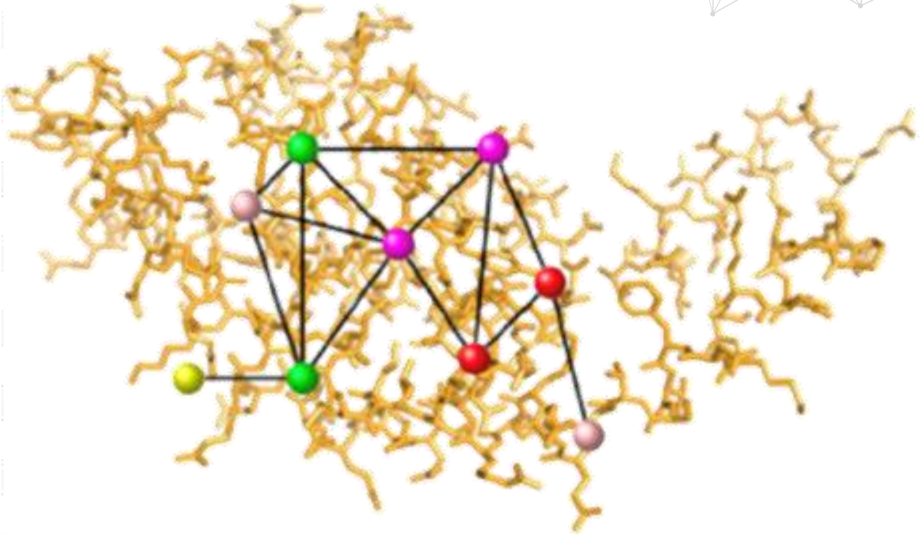


Why Solve Graph Problems?

Social Network Analysis



Protein Structure Analysis



Protein graph 1
(1allA)

What is a Permanent

$$\mathrm{Det}(A) = \sum_{\sigma \in S_n} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} a_{\sigma(i)i}$$

$$\mathrm{Per}(A) = \sum_{\sigma \in S_n} 1 \prod_{i=1}^{n} a_{\sigma(i)i}$$

What is a Permanent

$$\mathrm{Det}(A) = \sum_{\sigma \in S_n} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} a_{\sigma(i)i}$$

$$O(n^3)$$

$$\mathrm{Per}(A) = \sum_{\sigma \in S_n} 1 \prod_{i=1}^{n} a_{\sigma(i)i}$$

$$O(n2^n)$$

What is a Permanent

$$\mathrm{Det}(A) = \sum_{\sigma \in S_n} \mathrm{sgn}(\sigma) \prod_{i=1}^{n} a_{\sigma(i)i}$$

$$O(n^3)$$

$$50^3$$

$$= 125000$$

$$\mathrm{Per}(A) = \sum_{\sigma \in S_n} 1 \prod_{i=1}^{n} a_{\sigma(i)i}$$

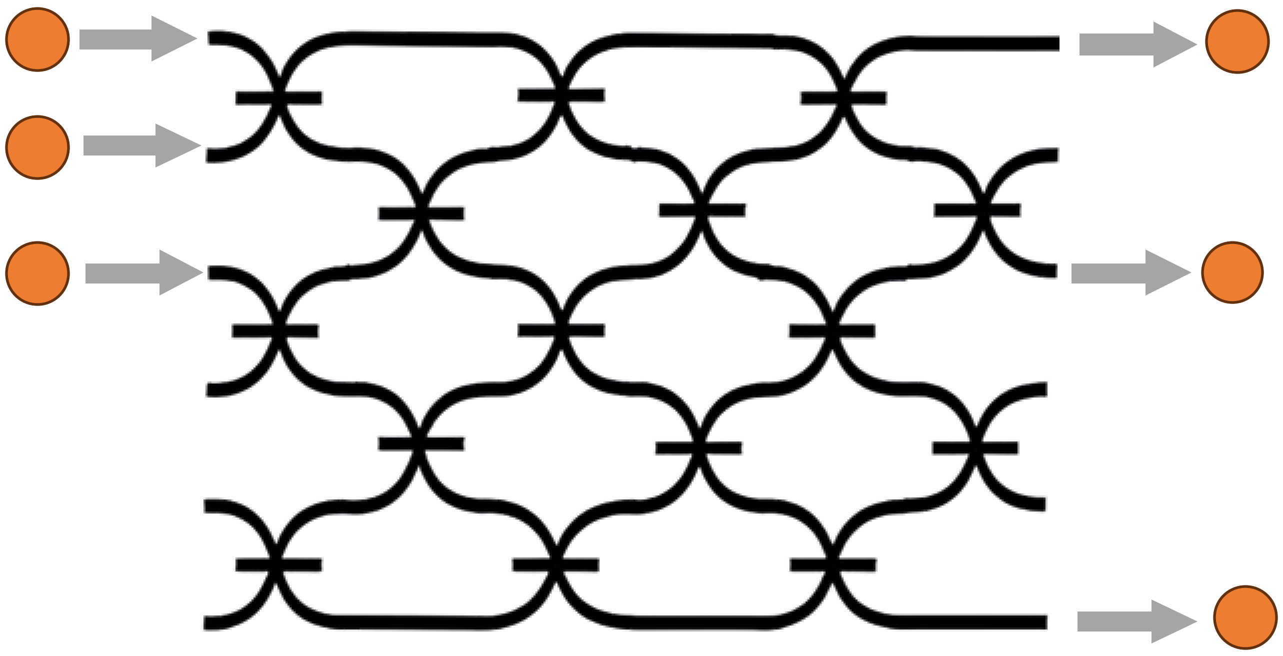
$$O(n2^n)$$

$$50 \cdot 2^{50}$$

$$= 56294995342131200$$

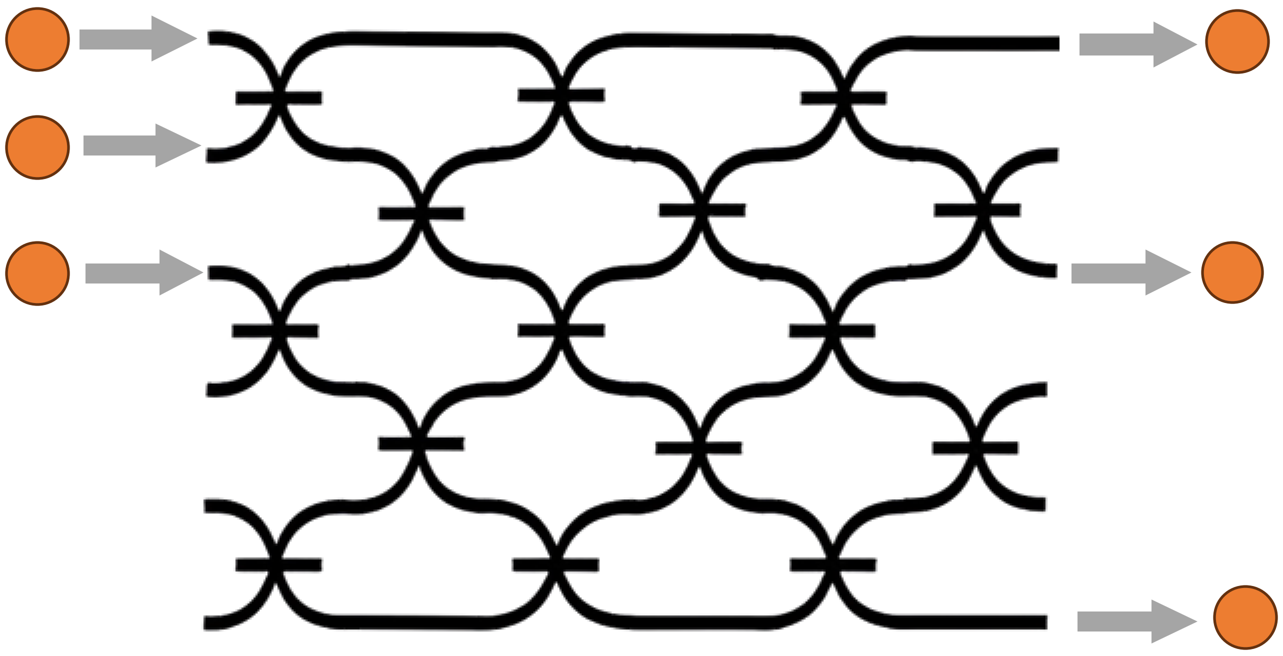
How can we use Photonic Quantum Computers?

Sample outcome probabilities



Sample outcome probabilities

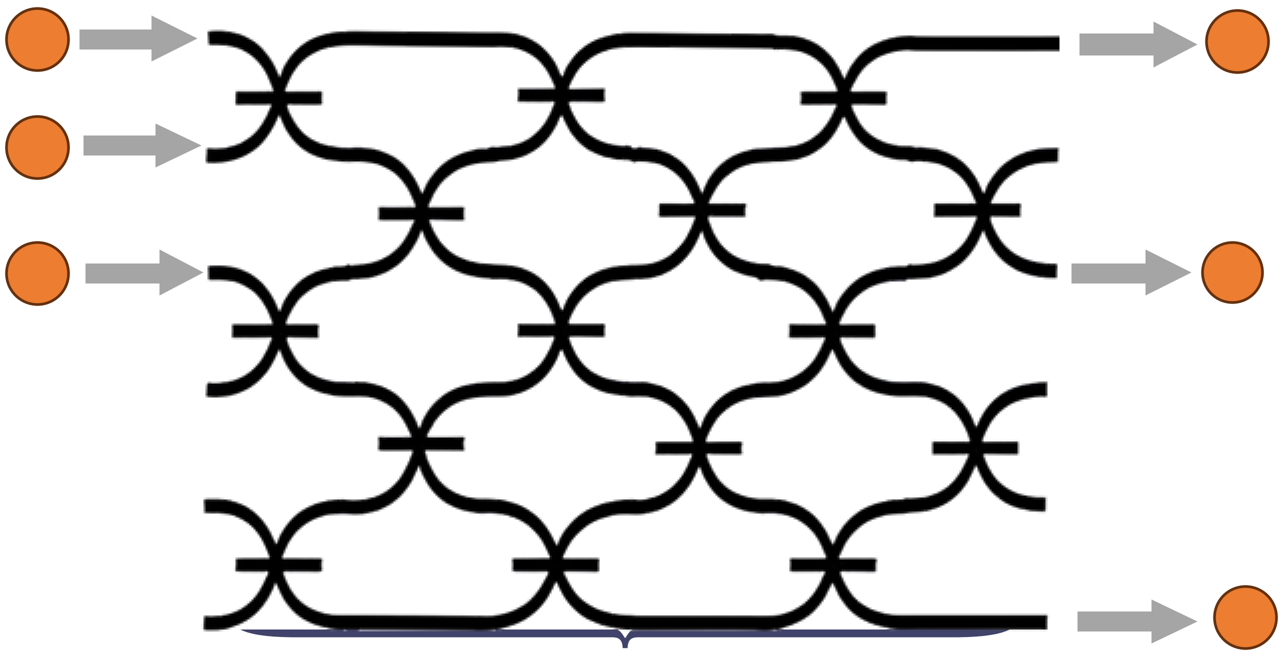
$$|\underline{n}_{in}\rangle = \begin{pmatrix}1\\1\\1\\0\\0\\0\end{pmatrix}$$



$$\begin{pmatrix}1\\0\\1\\0\\0\\1\end{pmatrix} = |\underline{n}_{out}\rangle$$

Sample outcome probabilities

$$|\underline{n}_{in}\rangle = \begin{pmatrix}1\\1\\1\\0\\0\\0\end{pmatrix}$$



$$\begin{pmatrix}1\\0\\1\\0\\0\\1\end{pmatrix} = |\underline{n}_{out}\rangle$$

U

$$p(\underline{n}_{out}|\underline{n}_{in}) = \left|Per\left(U_{\underline{n}_{in},\underline{n}_{out}}\right)\right|^2$$

Encoding the Graph Theory Problem

- Have a size n graph G
- Have it's $n \times n$ adjacency matrix A
- We set up a circuit (pick a U) and pick $\underline{n}_{in}$ input distribution such that for some output $\underline{n}_{out}$:

$$p(\underline{n}_{out}|\underline{n}_{in}) = \left|Per\left(U_{\underline{n}_{in},\underline{n}_{out}}\right)\right|^2 \propto |Per(A)|^2$$

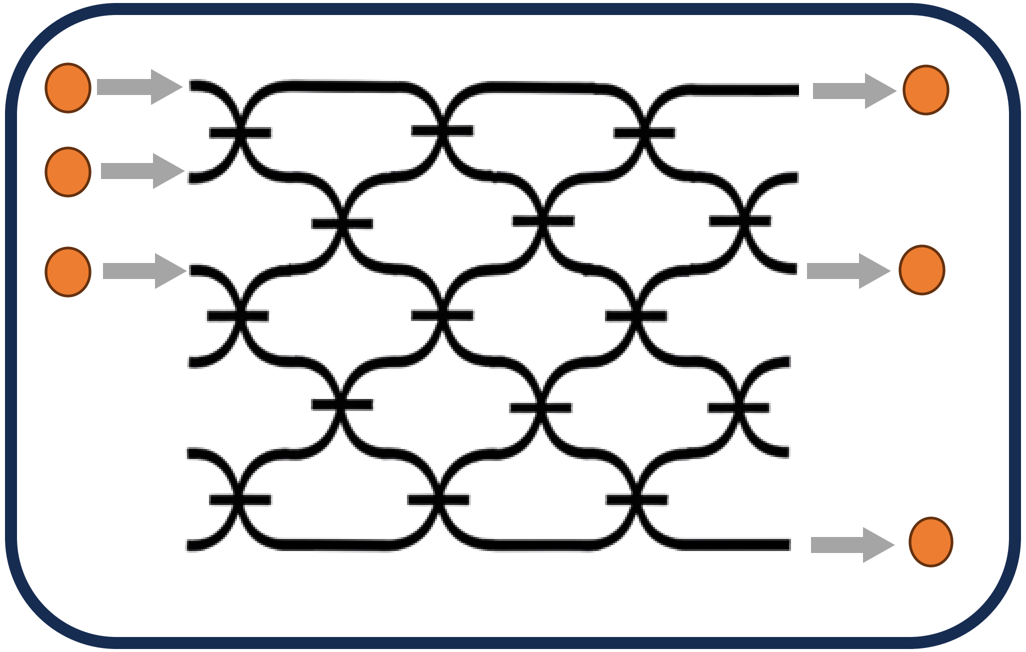
Our final probability

With this setup, we have an outcome probability:

$$p = p(\underline{n}_{out}|\underline{n}_{in}) = |Per(A_s)|^2 = \frac{1}{\lambda_{max}^{2n}}|Per(A)|^2$$

If we estimate p, we can calculate the permanent of our graph!

Estimating the Permanent



Photonic circuit with graph encoded

$$\frac{\#\, \underline{n}_{out}\; samples}{\#\, samples} = p_{est}$$

$$Per_{est} = \lambda_{max}^{n} \sqrt{p_{est}}$$

As $\#\, samples \rightarrow \infty$:

$$Per_{est} \rightarrow Per$$

The Complete Algorithm

- Have a size n graph G
- We want to calculate $Per(A)$, the value of its permanent
- Set up a photonic quantum circuit such that the probability of a certain outcome is:

$$p(\underline{n}_{out}|\underline{n}_{in}) \propto |Per(A)|^2$$

- We estimate this value, and we get useful information about the graph!

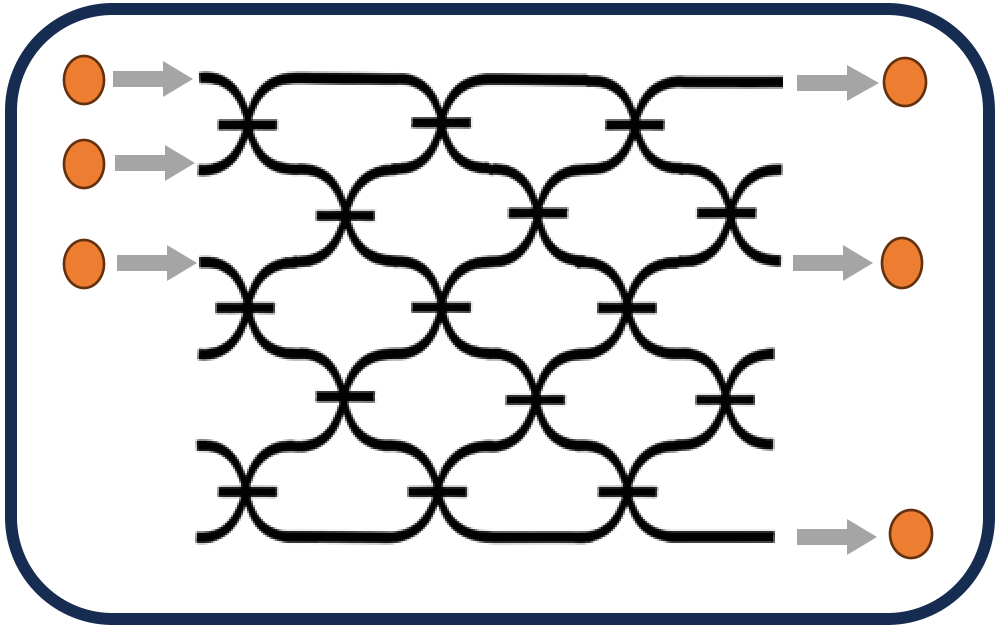
Proposed by R. Mezher, A.F. Carvalho & S. Mansfield:

“Solving graph problems with single-photons and linear optics”

My Analysis of the Algorithm

How many samples do we need?

We want to calculate the value Per within some percentage error of the true value. It turns out, we need:


Photonic circuit with graph encoded

$$N_{samples} \geq \frac{1}{p}$$

We want this to be a lot smaller than the classical complexity of $n2^n$ to get 'quantum advantage'

Lower bounding sample number

$$N_{samples} \geq \frac{1}{p} = \frac{1}{|Per(A_s)|^2} = \frac{\lambda_{max}^{2n}}{|Per(A)|^2}$$

We want to lower bound this in terms of n: the size of the graph. With (lots of) work, we get to the following bound:

$$N_{samples} \geq \frac{\lambda_{max}^{2n}}{|Per(A)|^2} \geq 2^{2n}$$

What can we conclude?

Calculating permanents for general graphs with this proposition isn't very efficient.

We need to run our circuit 2^{2n} times to do this for a size n graph! Classical computers only need $n2^n$ operations to calculate the value exactly.

What can we conclude?

Calculating permanents for general graphs with this proposition isn’t very efficient.

We need to run our circuit 2^{2n} times to do this for a size n graph! Classical computers only need $n2^n$ operations to calculate the value exactly.

$$50 \cdot 2^{50} = 56294995342131200$$

What can we conclude?

Calculating permanents for general graphs with this proposition isn't very efficient.

We need to run our circuit 2^{2n} times to do this for a size n graph! Classical computers only need $n2^n$ operations to calculate the value exactly.

$$50 \cdot 2^{50} = 56294995342131200$$

$$2^{2 \cdot 50} = 1267650600228229400000000000000$$

Photonic Quantum Computers <u>are</u> Useful

With some slight tweaking of our input state, we can use our setup to efficiently sample a distribution of subgraphs with bias towards those with more edges. It takes classical computers a long time to do this.

There’s lots of promising research into the application of photonic quantum computers for:

- optimisation problems
- machine learning
- simulating physical phenomena

ORCA PT-series device

Thank you very much to Sally Baume for coordinating the CMP program!

Thank you very much to my supervisors William Clements, Thorin Farnsworth, Hugo Wallner and the rest of the team at ORCA!

Thanks for listening!

Please do ask any questions you have