Quantum Algorithms

Shelby Kimmel
Department of Computer Science
Middlebury College

Middlebury

The Quantum Question

- If I have a quantum computer, could I solve this difficult problem in my field faster/more accurately?

The Quantum Question

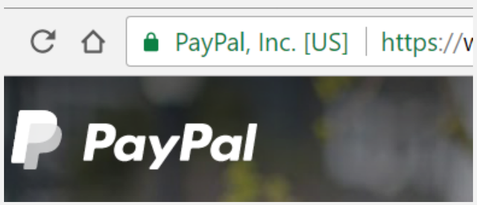
- If I have a quantum computer, could I solve this difficult problem in my field faster/more accurately?

The Answer:

- We don't know yet (for many problems)

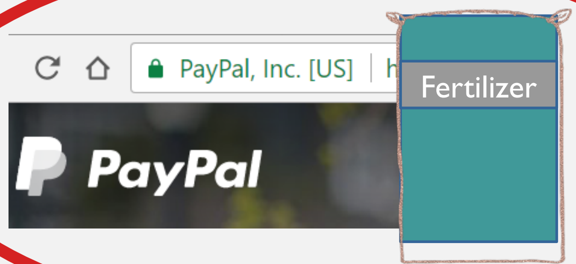
Quantum Algorithm Landscape

Large Advantage Over Best
Known Classical Algorithm



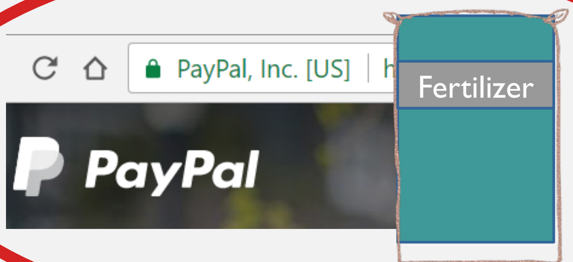
Quantum Algorithm Landscape

Large Advantage Over Best
Known Classical Algorithm



Quantum Algorithm Landscape

Large Advantage Over Best Known Classical Algorithm



Small/No Advantage Over Best Classical Algorithms

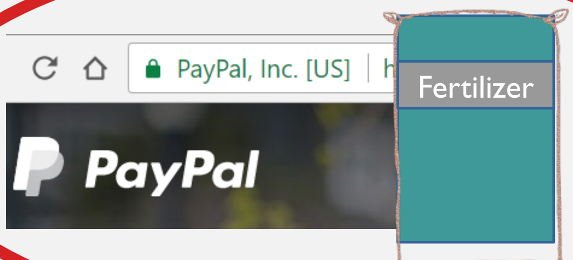
PARITY

SEARCH

COMPLETE GRAPH CONNECTIVITY

Quantum Algorithm Landscape

Large Advantage Over Best Known Classical Algorithm



Small/No Advantage Over Best Classical Algorithms

PARITY

SEARCH

COMPLETE GRAPH CONNECTIVITY

Quantum Advantage Unknown

Lots…

Quantum Algorithms: Promise and Perspective

1. **Temper expectations:**

2. **Build excitement:**

Quantum Algorithms: Promise and Perspective

1. **Temper expectations:** quantum computers are sometimes, but not always, helpful

2. **Build excitement:**

Quantum Algorithms: Promise and Perspective

1. **Temper expectations:** quantum computers are sometimes, but not always, helpful
2. **Build excitement:** quantum computing devices provide an unprecedented tool for designing and studying quantum algorithms

Quantum Advantage

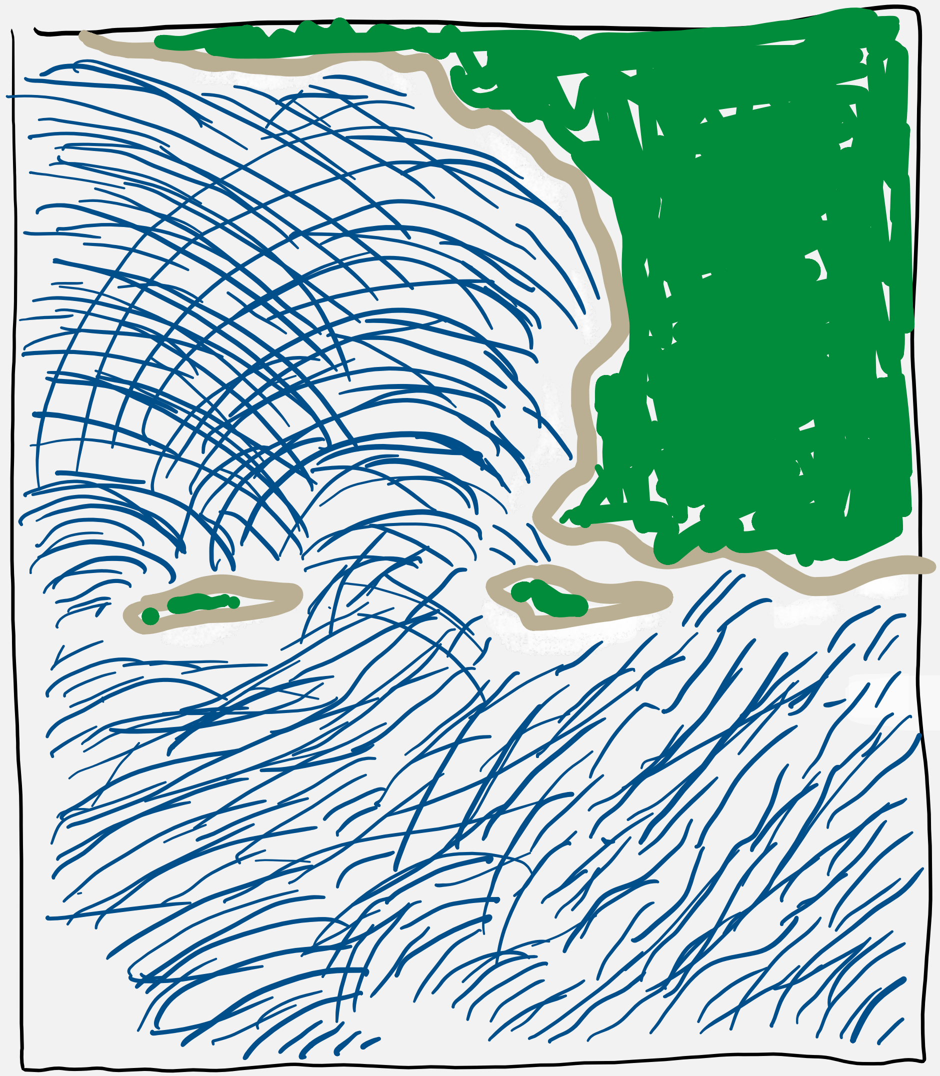
Quantum computers seem really helpful for some problems but not others.

WHY?

- Superposition
- Interference

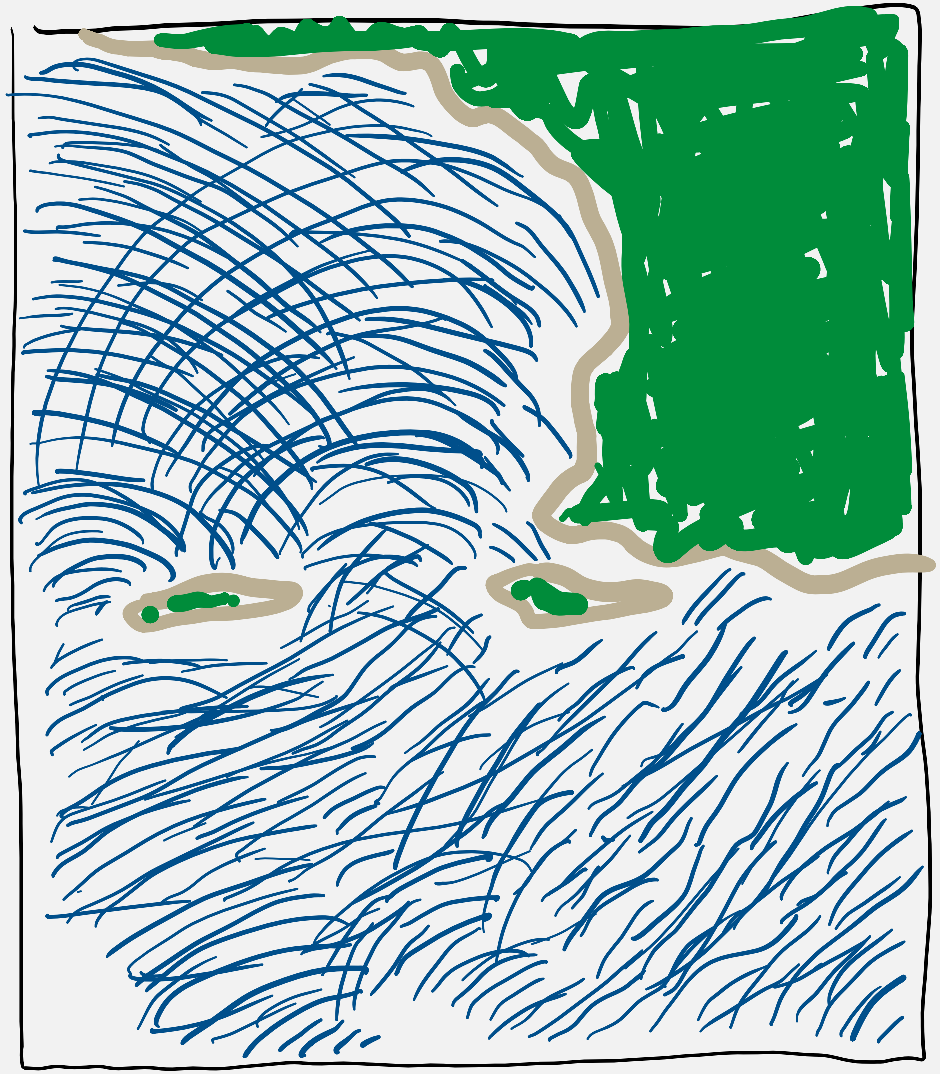
Metaphor for quantum computer

- Quantum computation is like interaction of waves, islands, and shore



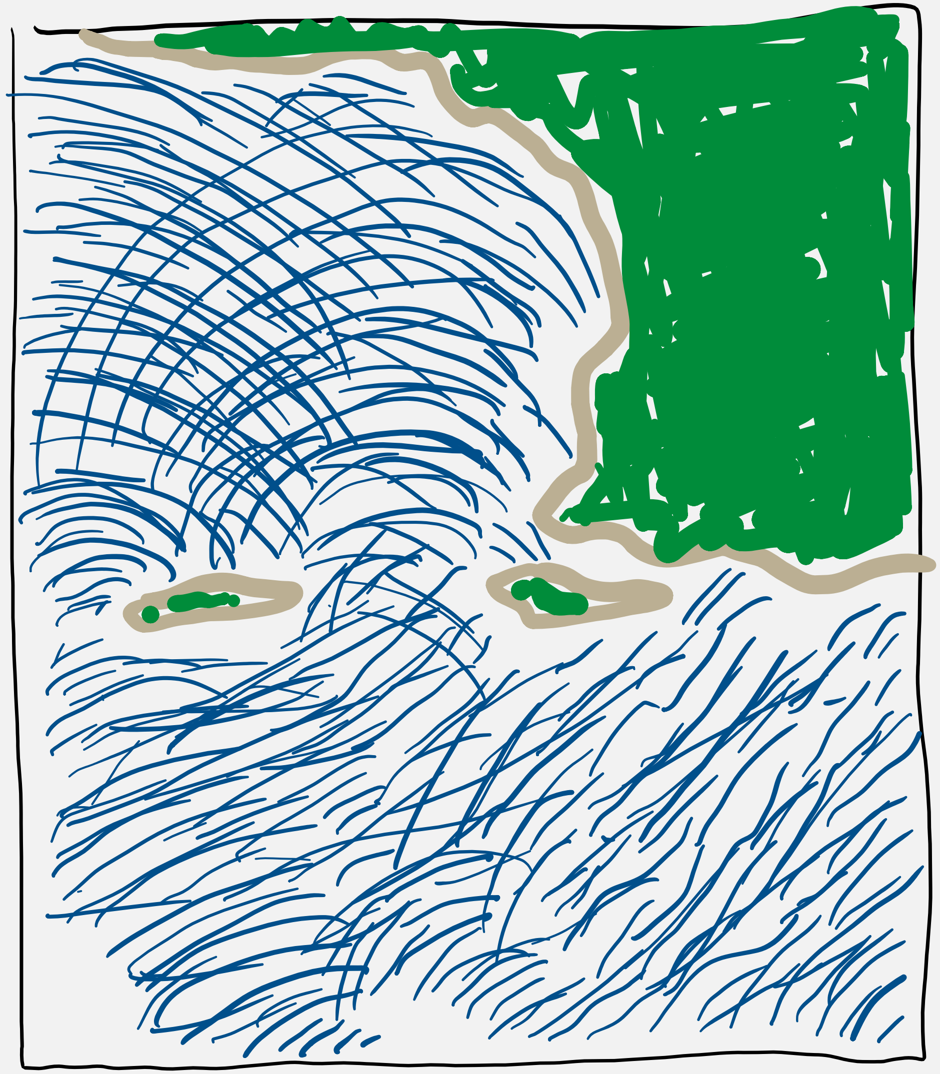
Metaphor for quantum computer

- Quantum computation is like interaction of waves, islands, and shore
- I can control islands and shoreline



Metaphor for quantum computer

- Quantum computation is like interaction of waves, islands, and shore
- I can control islands and shoreline
- Output of computation is location of a large wave



Quantum Advantage

Quantum computers seem really helpful for some problems but not others.

WHY?

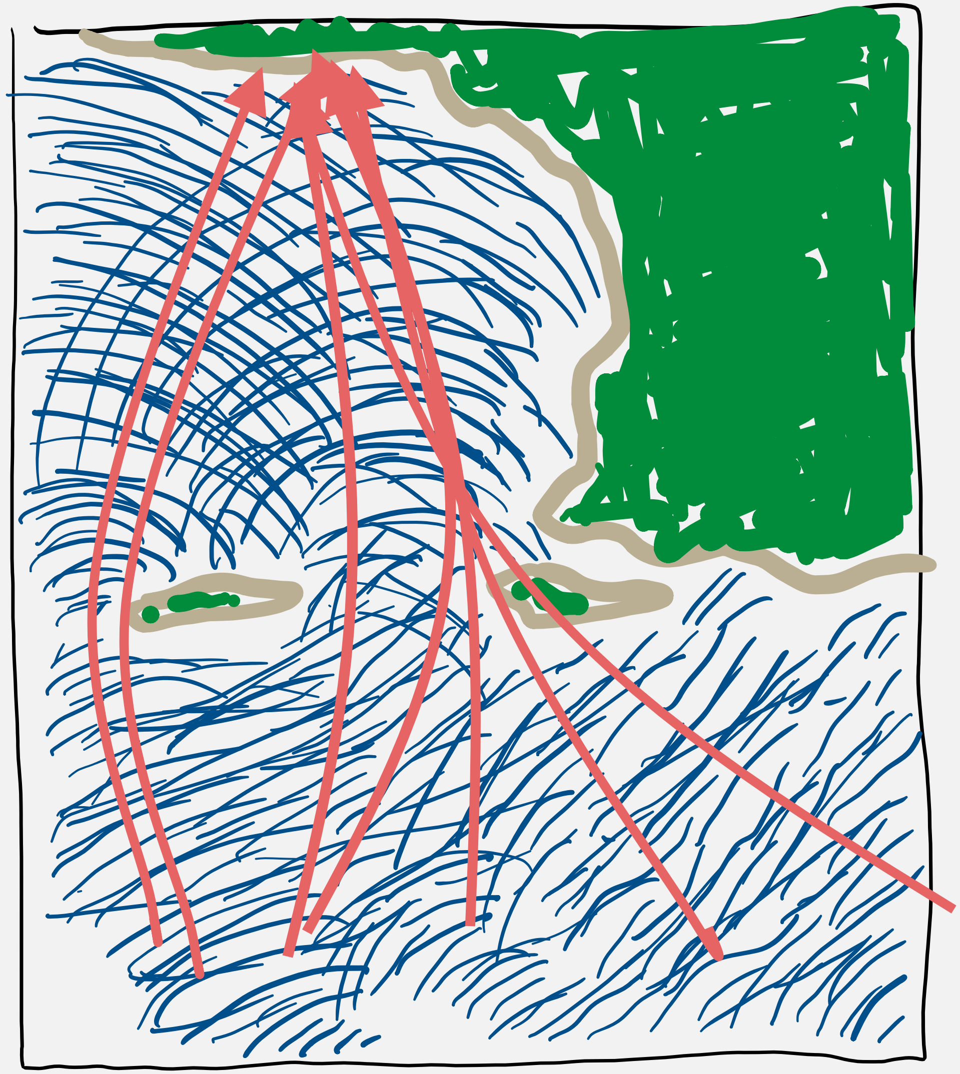
- Superposition
- Interference

Quantum Advantage

- Superposition – "can be in many states at once"

Quantum Advantage

- Superposition: waves explore many paths through the environment, hit all the points on the shore



Quantum Advantage

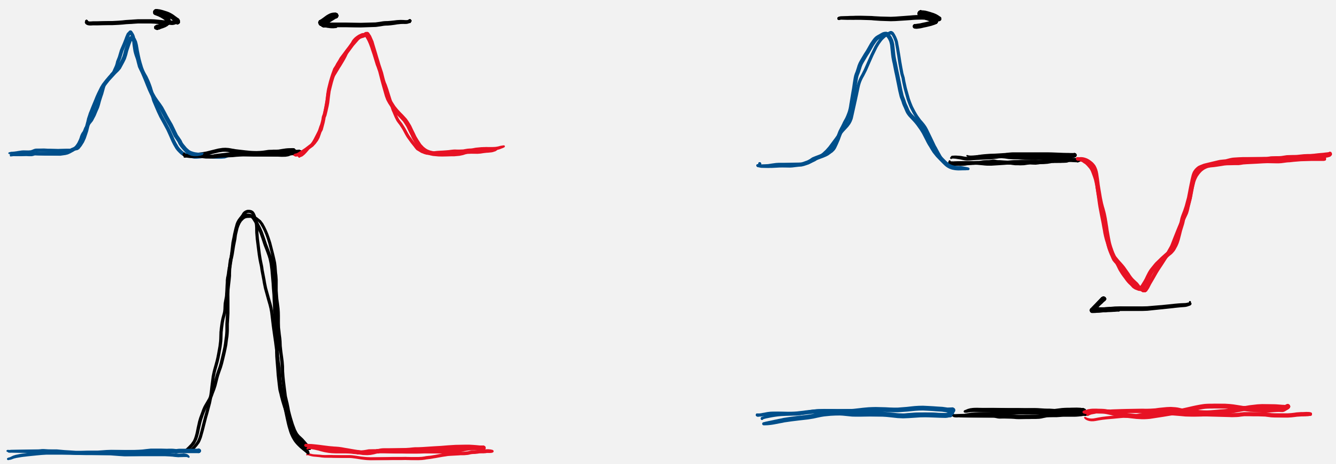
What gives quantum computers their power?
Why is this power helpful for some problems, but not helpful for others?

- Superposition – "can be in many states at once"

- Interference

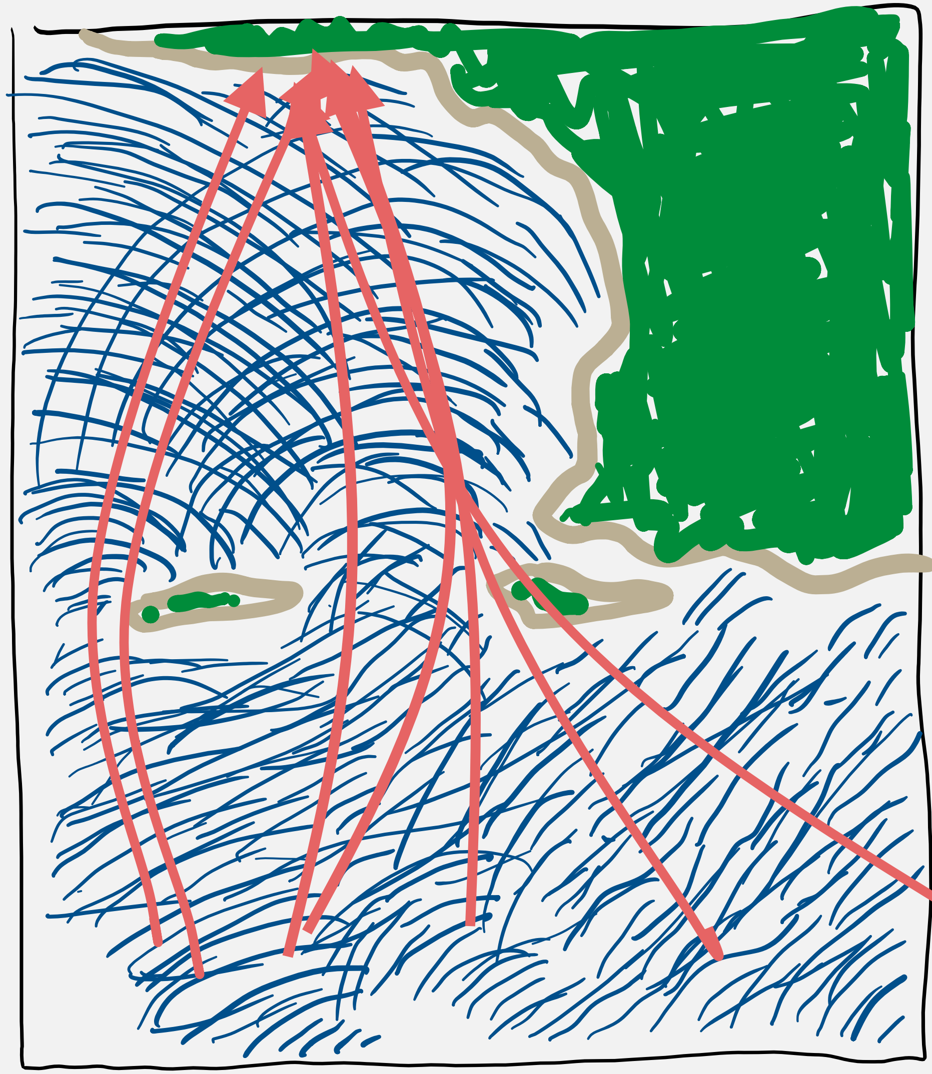
Quantum Advantage

- Interference



Quantum Advantage

- Interference: waves travel from different paths. If in sync when get to shore, get big wave, if out of of sync, get no wave.



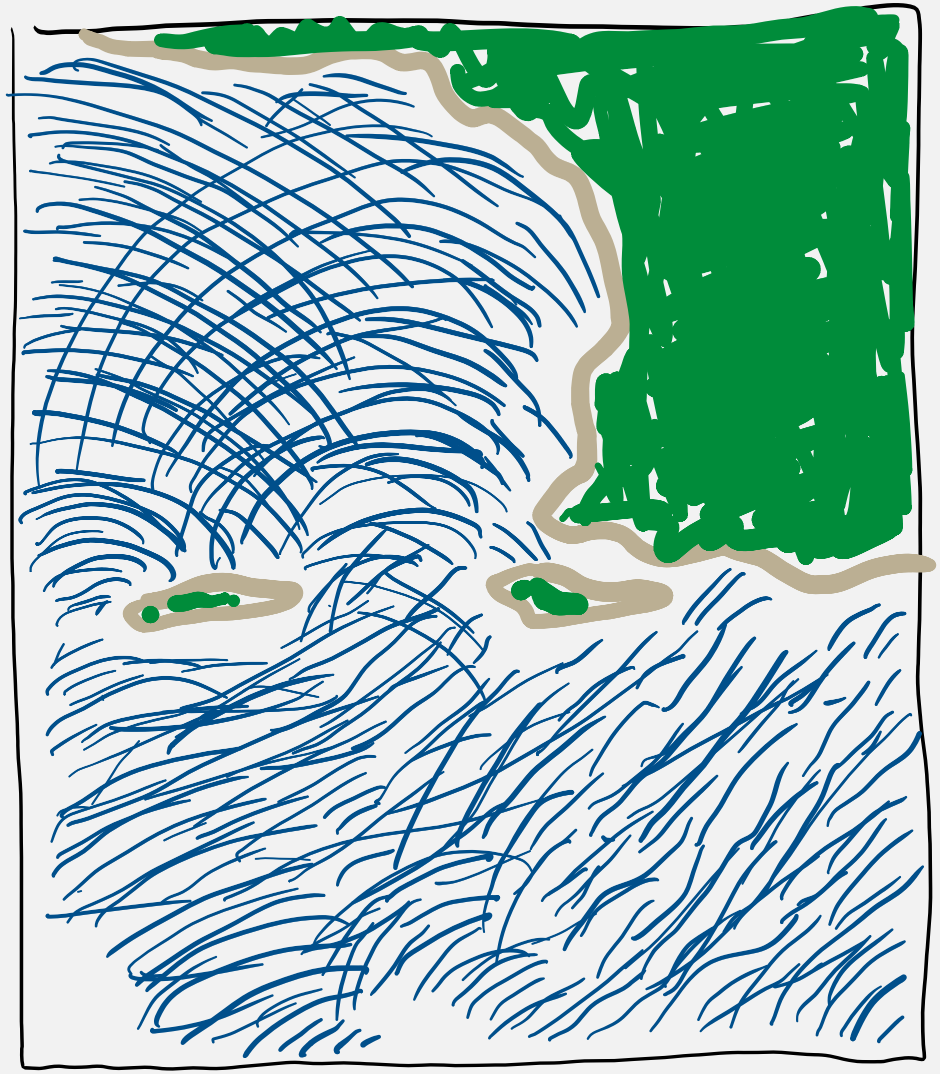
Quantum Advantage

What gives quantum computers their power?
Why is this power helpful for some problems, but not helpful for others?

- Superposition – "can be in many states at once"
- Interference – "cancel the bad, enforce the good"

Metaphor for quantum computer

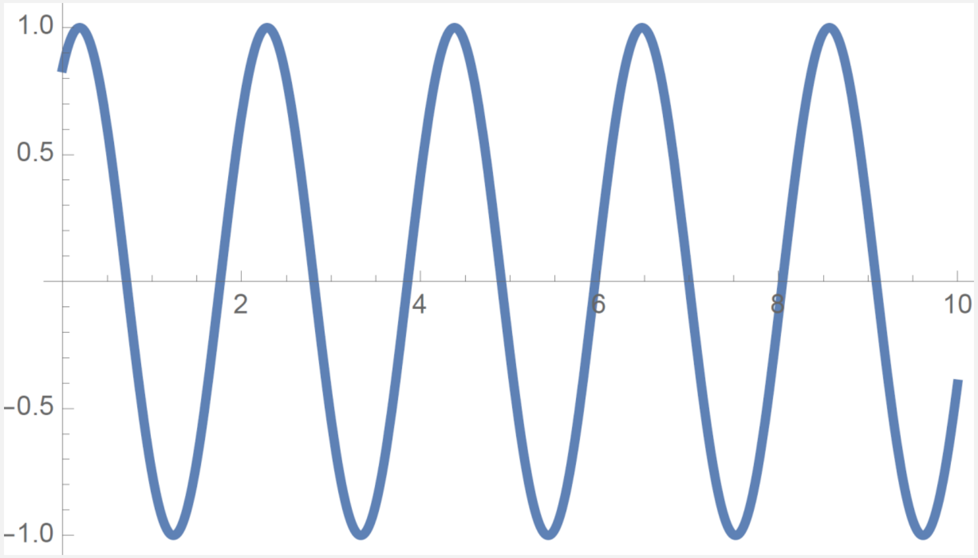
- Quantum computation is like interaction of waves, islands, and shore
- I can control islands and shoreline
- Output of computation is location of a large wave



Quantum Advantage

- Some problems have structure that helps build up interference fast:

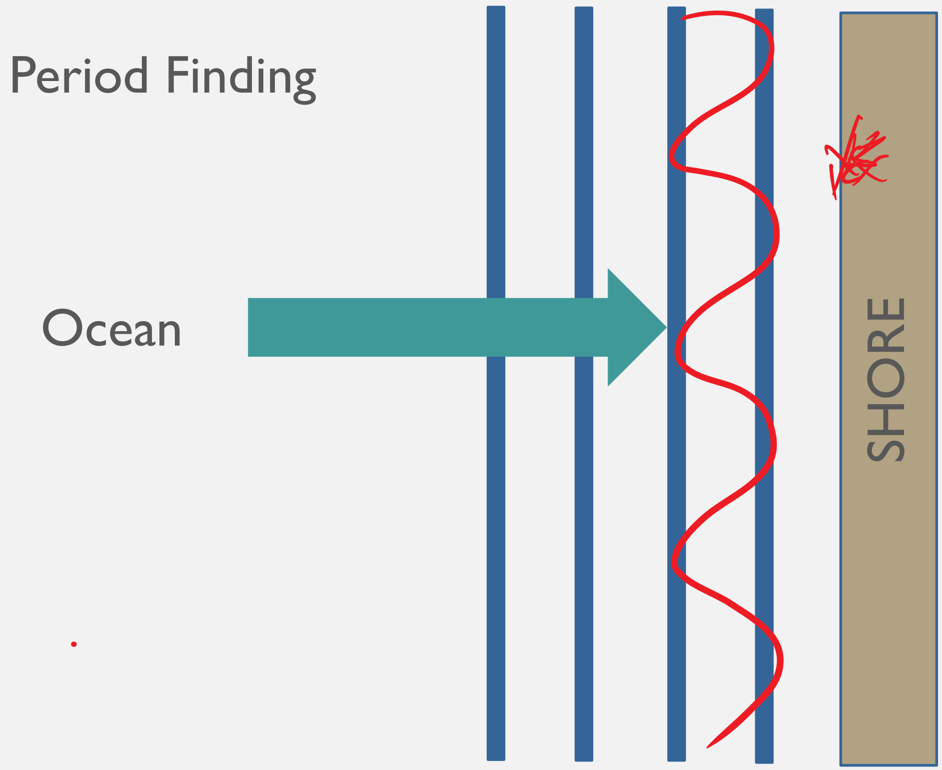
What is the period of a periodic function?



Period

Quantum Advantage

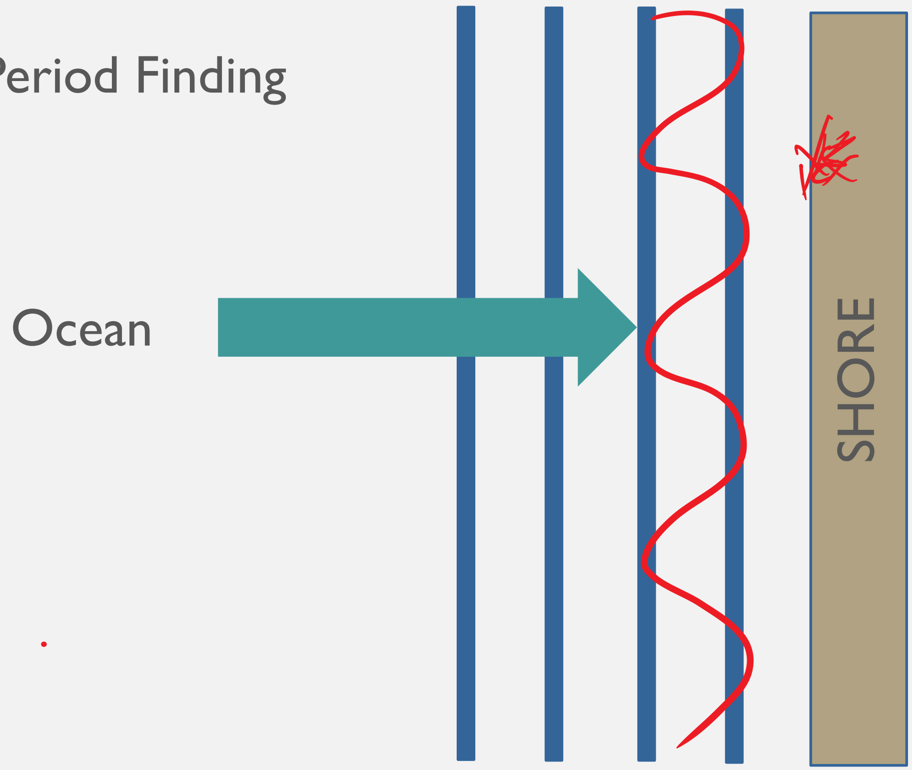
- Some problems have structure that helps build up interference fast:



Quantum Advantage

- Some problems have structure that helps build up interference fast:

Period Finding



Quantum computers can find the period of a function exponentially faster than regular computers

Used to break cryptosystems

Quantum Advantage

- Other problems have very little structure, need more time to build up interference

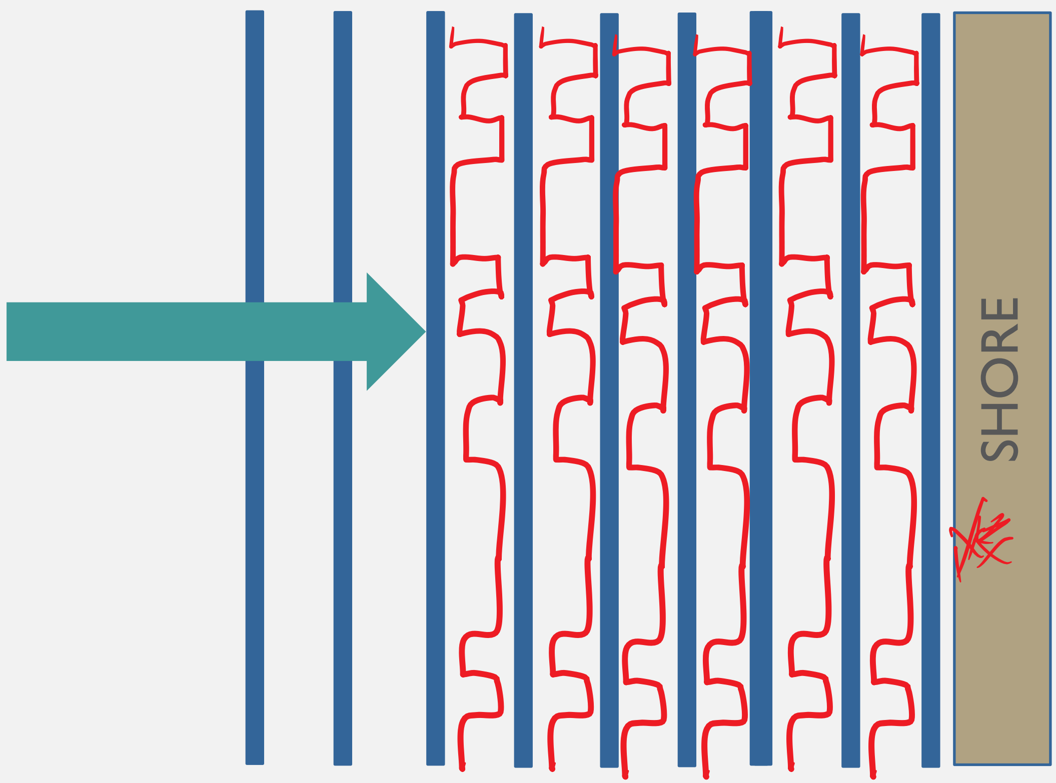
Parity: Even or odd number of 1's

011010101110 (seven $1's$ → odd parity)
011010001110 (six $1's$ → even parity)

Quantum Advantage

- Other problems have very little structure, need more time to build up interference

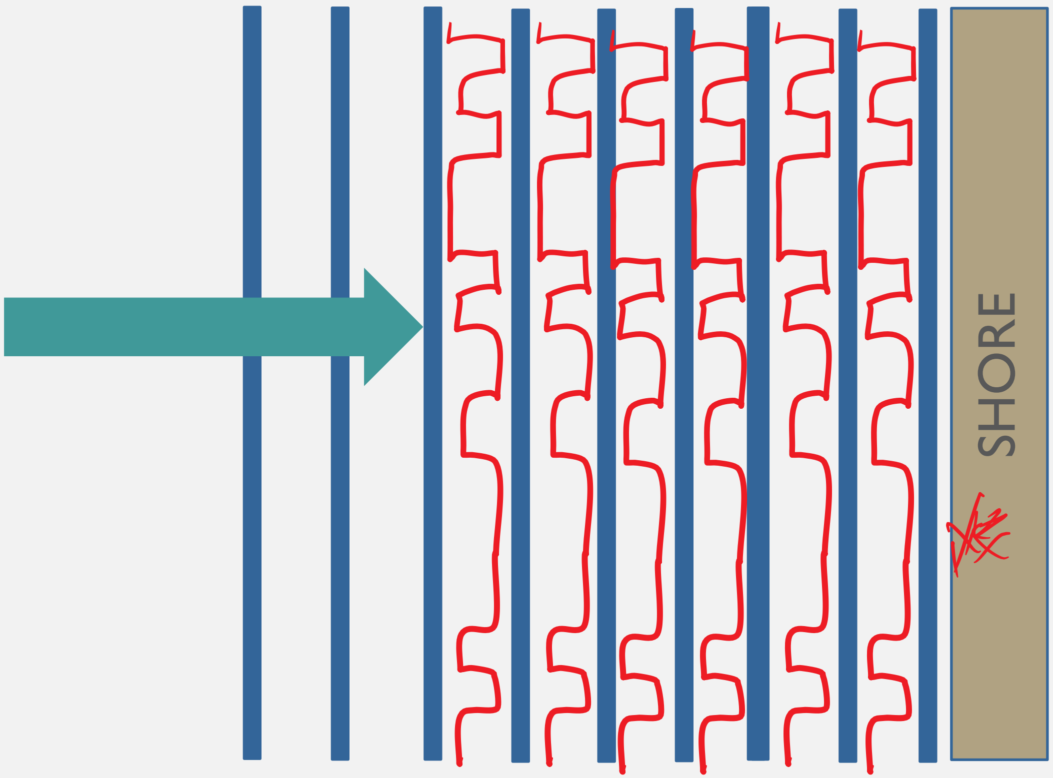
Parity



Quantum Advantage

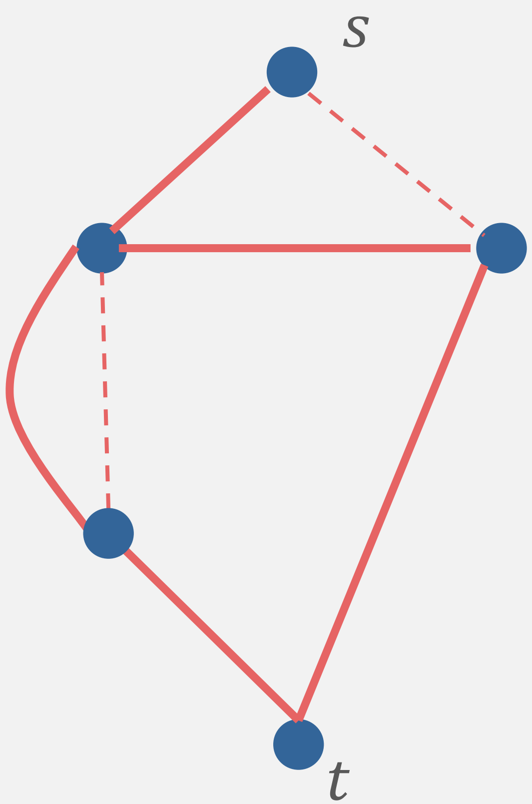
- Other problems have very little structure, need more time to build up interference

Parity:
No quantum advantage



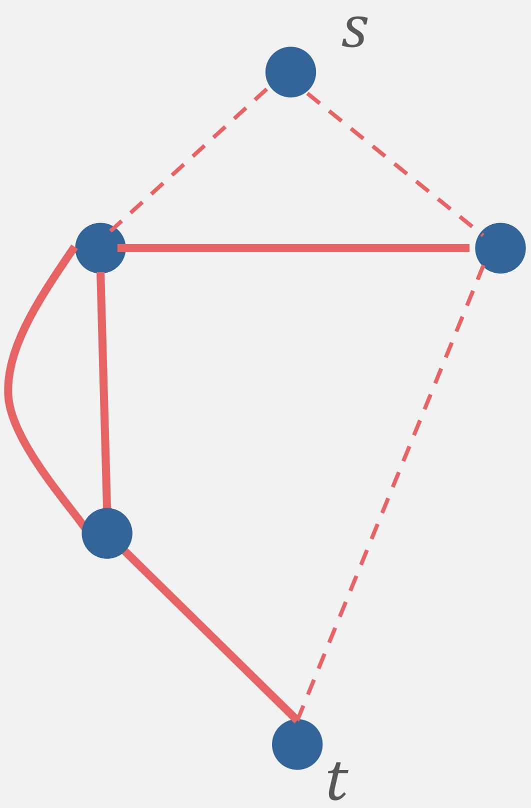
Quantum Advantage for st-connectivity

$\boldsymbol{st - connectivity}$:
is there a path from s to t?

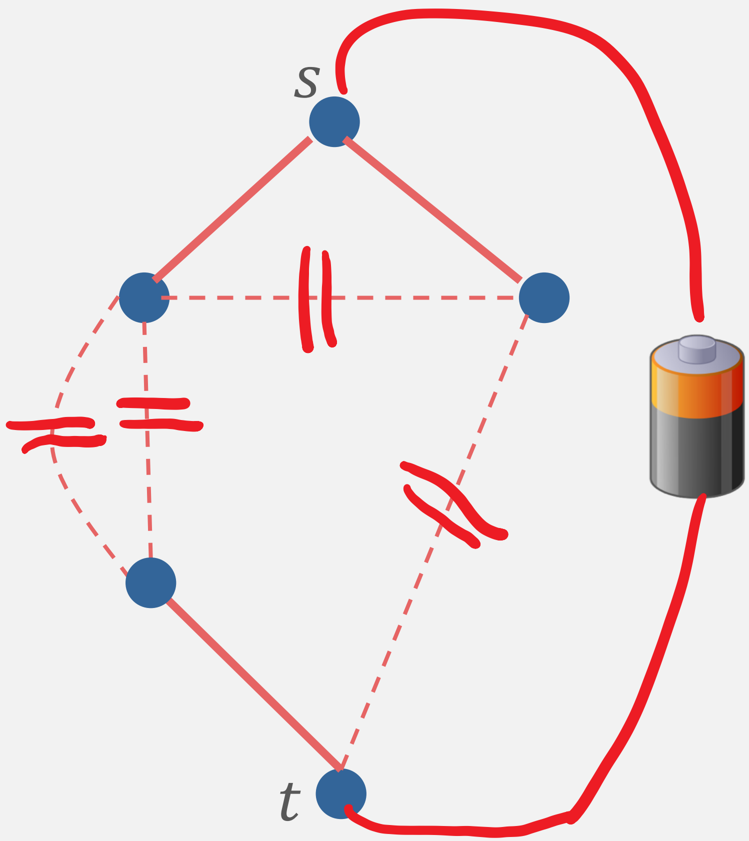


Quantum Advantage for st-connectivity

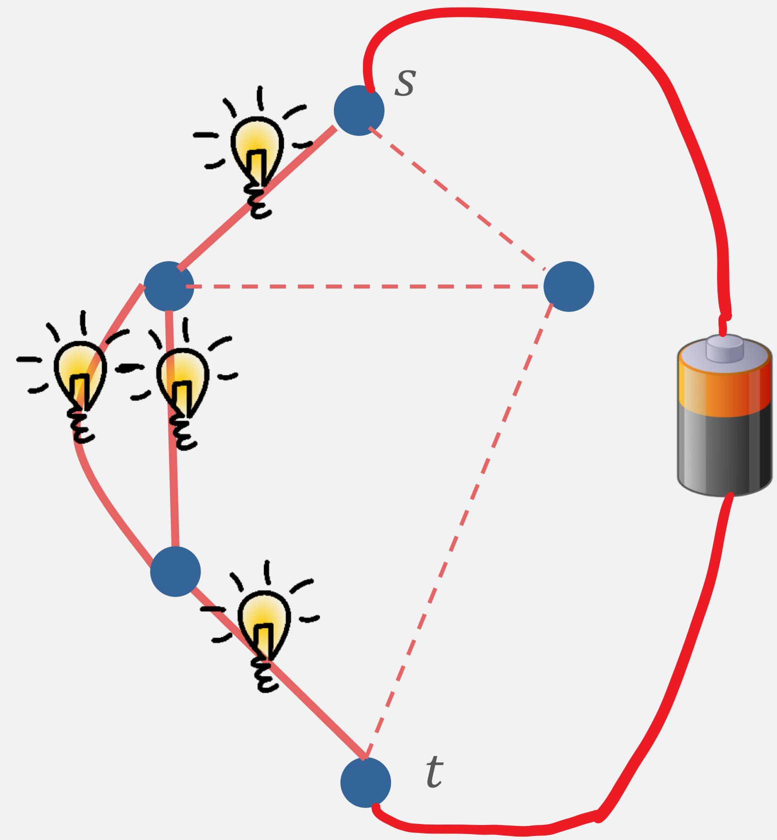
$st - connectivity$:
is there a path from s to t?



Quantum Advantage for st-connectivity



Less charge build-up (smaller effective capacitance) → easier for quantum computer to solve



More current flow (smaller effective resistance) → easier for quantum computer to solve

Quantum Algorithms: Promise and Perspective

1. **Temper expectations:** quantum computers are sometimes, but not always, helpful

2. **Build excitement:** quantum computing devices provide an unprecedented tool for designing and studying quantum algorithms

Designing a Quantum Algorithm

Quantum Algorithm Designer's Toolbox

Quantum Walk

Phase Estimation

Adiabatic Evolution

QAOA

Quantum Annealing

Span Program

Designing a Quantum Algorithm

Quantum Algorithm Designer's Toolbox

Quantum Walk

Phase Estimation

Adiabatic Evolution

QAOA

Quantum Annealing

Span Program

Unless the problem has very nice/simple structure, analyzing correctness and performance is very difficult

Designing a Quantum Algorithm

Quantum Algorithm Designer's Toolbox

Quantum Walk

Phase Estimation

Adiabatic Evolution

QAOA

Quantum Annealing

Span Program

With advent of quantum computing devices, can test!

Challenges

Quantum Algorithm Designer's Toolbox

Quantum Walk

Phase Estimation

Adiabatic Evolution

QAOA

Quantum Annealing

Span Program

Challenges

Span program algorithm:

$$
\begin{aligned}
\forall i \in [N], b \in \{0,1\}: \ H_{i,b} &= \mathrm{span}\{|e\rangle : e \in \overrightarrow{E}_{i,b}\} \\
U &= \mathrm{span}\{|v\rangle : v \in V(G)\} \\
\tau &= |s\rangle - |t\rangle \\
\forall e = (u,v,\ell) \in \overrightarrow{E}(G): \ A|u,v,\ell\rangle &= \sqrt{c(u,v,\ell)}(|u\rangle - |v\rangle)
\end{aligned}
$$

```
p.defgate("RACL", oracl)
p.defgate("ZEROPHS", zeroPhase)

#initialize
p.inst([H(i) for i in range(n)])

#run Grover iterate
for _ in range(int(m)):
    p.inst(("RACL"+s))
    p.inst([H(i) for i in range(n)])
    p.inst(("ZEROPHS"+s))
    p.inst([H(i) for i in range(n)])
```

Challenges

$$\begin{aligned} \forall i \in [N], b \in \{0,1\}: \ H_{i,b} &= \mathrm{span}\{|e\rangle : e \in \overrightarrow{E}_{i,b}\} \\ U &= \mathrm{span}\{|v\rangle : v \in V(G)\} \\ \tau &= |s\rangle - |t\rangle \\ \forall e = (u,v,\ell) \in \overrightarrow{E}(G): \ A|u,v,\ell\rangle &= \sqrt{c(u,v,\ell)}(|u\rangle - |v\rangle) \end{aligned}$$

```
p.defgate("RACL", oracl)
p.defgate("ZEROPHS", zeroPhase)

#initialize
p.inst([H(i) for i in range(n)])

#run Grover iterate
for _ in range(int(m)):
    p.inst(("RACL"+s))
    p.inst([H(i) for i in range(n)])
    p.inst(("ZEROPHS"+s))
    p.inst([H(i) for i in range(n)])
```

Quantum Machine Code (sequence of simple quantum operations)

Challenges

$$\forall i \in [N], b \in \{0,1\}:\ H_{i,b} = \mathrm{span}\{|e\rangle : e \in \overrightarrow{E}_{i,b}\}$$
$$U = \mathrm{span}\{|v\rangle : v \in V(G)\}$$
$$\tau = |s\rangle - |t\rangle$$
$$\forall e = (u,v,\ell) \in \overrightarrow{E}(G):\ A|u,v,\ell\rangle = \sqrt{c(u,v,\ell)}(|u\rangle - |v\rangle)$$

```
p.defgate("RACL", oracl)
p.defgate("ZEROPHS", zeroPhase)

#initialize
p.inst([H(i) for i in range(n)])

#run Grover iterate
for _ in range(int(m)):
    p.inst(("RACL"+s))
    p.inst([H(i) for i in range(n)])
    p.inst(("ZEROPHS"+s))
    p.inst([H(i) for i in range(n)])
```

Quantum Machine Code (sequence of simple quantum operations) (tailored for particular implementation)

Outlook

1. Temper your expectations…
2. Potential of quantum algorithms is just beginning to be explored

Questions?

<u>Theoretical collaborators:</u> Stacey Jeffery, Michael Jarret, Alvaro Piedrafita