Grover’s Algorithm

Matthew Hurtado and Darwin Vargas

June 18, 2024

Background

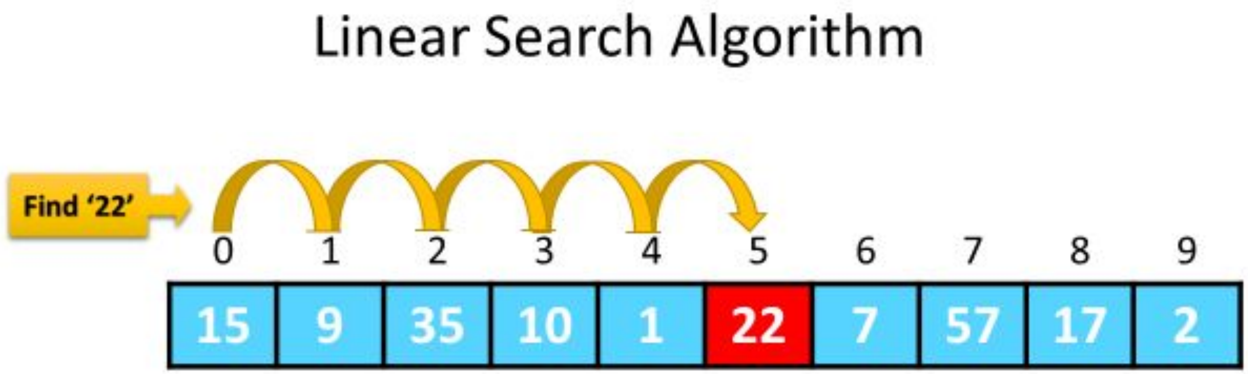
What if you are searching for the number of a given person in a sorted list of names?

What about the reverse? (Finding the name using only the number)

Name	Phone Number
Alice	314-1592
Bob	271-8281
Charlie	105-4571
Dave	885-4187
Eve	125-6637
Frank	299-7924
Grace	729-7352
⋮	⋮
Zoe	200-2319

Background - Classical Solutions

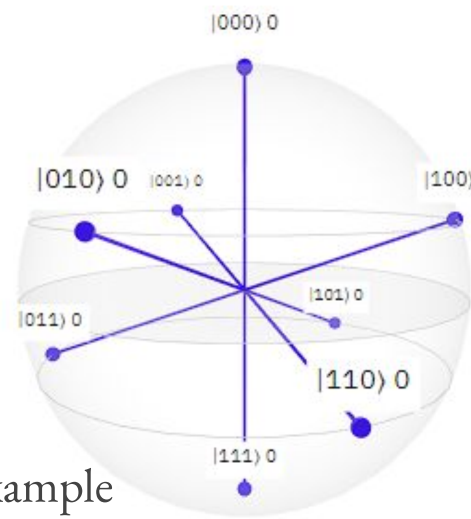
Linear Search/Probabilistic - O(N), where N = 2^n



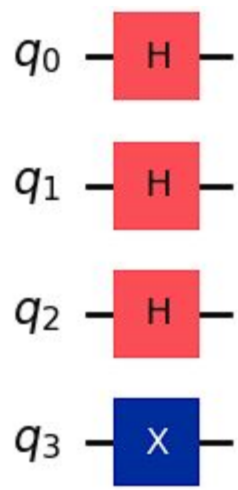
Quantum Approach - Grover's Algorithm

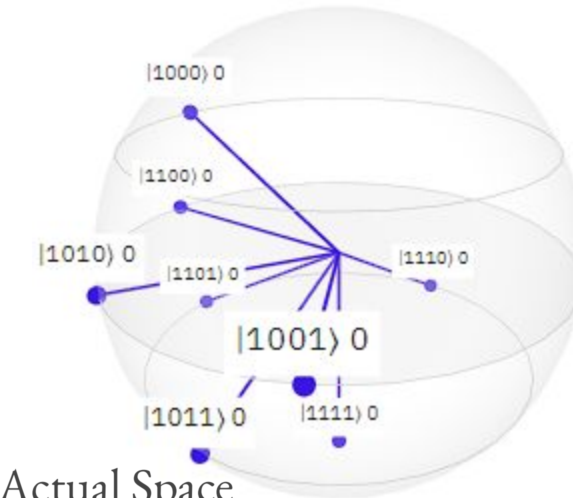
Quantum algorithm used to perform a search within an unordered list of elements.

- Quadratic speedup of O(sqrt(N))
- How?



Example





Actual Space

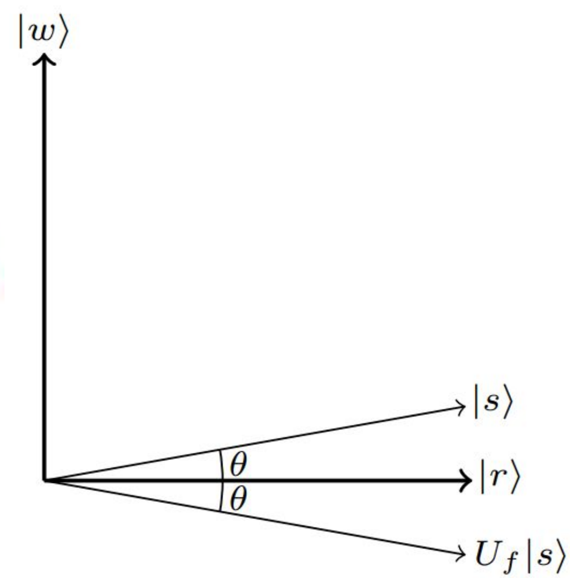
Oracle / Phase Query Gates

f(x) = 1, if x = w

f(x) = 0, otherwise

$$U_f\,|s\rangle = (-1)^{f(w)}\sin\theta\,|w\rangle + (-1)^{f(r)}\cos\theta\,|r\rangle$$

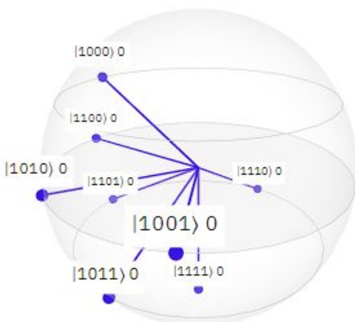
$$\begin{aligned} U_f|s\rangle &= (-1)^1\sin\theta|w\rangle + (-1)^0\cos\theta|r\rangle \\ &= -\sin\theta|w\rangle + \cos\theta|r\rangle. \end{aligned}$$

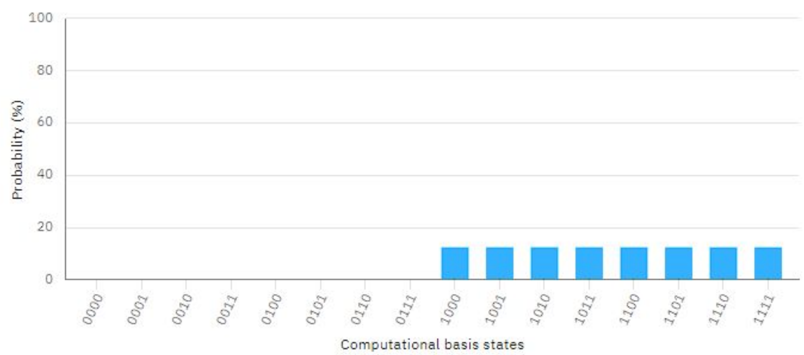


Algorithm Outline

Begin by putting the **input** qubits in a uniform superposition of all n-bit strings.

$$|s\rangle = |+\rangle^{\otimes n} = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} |x\rangle, \qquad N = 2^n$$





Example with 2 Qubits

$$
\begin{aligned}
|s\rangle &= |+\rangle^{\otimes 2} \\
&= \frac{1}{\sqrt{4}}(|00\rangle + |01\rangle + |10\rangle + |11\rangle) \\
&= \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)
\end{aligned}
$$

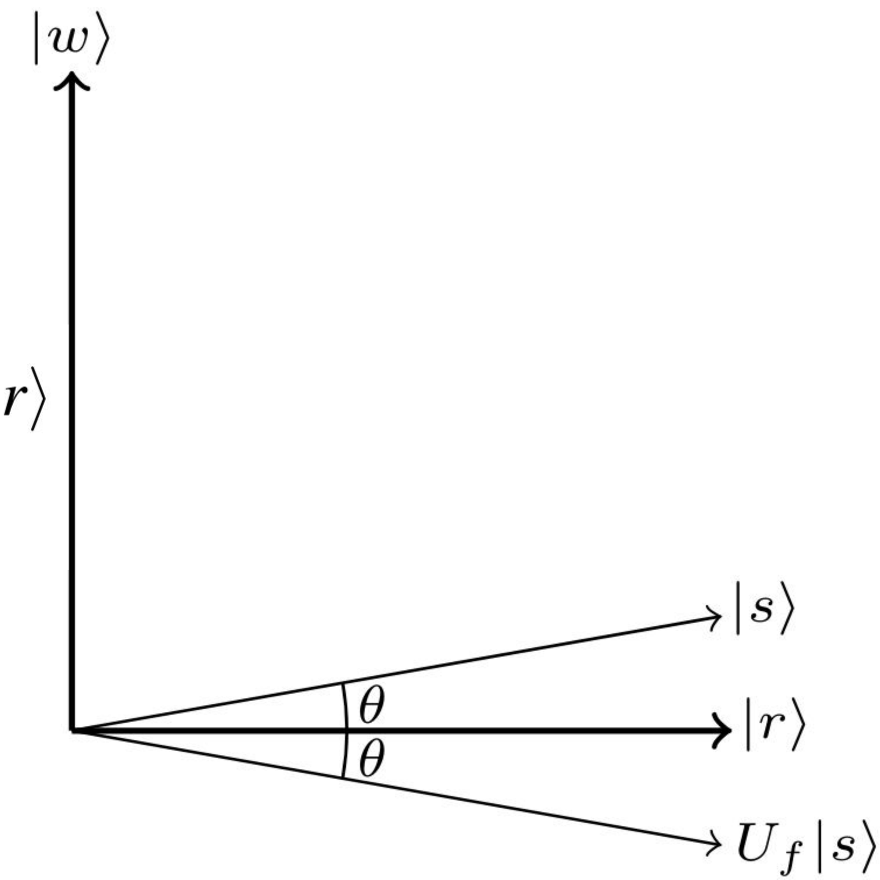
Convenient Form

$|r\rangle$ is a uniform superposition over all n-bit strings that are not $|w\rangle$

$$
\begin{aligned}
|s\rangle &= \frac{1}{\sqrt{N}}\left(|w\rangle + \sum_{i\neq w}|i\rangle\right) \\
&= \frac{1}{\sqrt{N}}|w\rangle + \frac{1}{\sqrt{N}}\sum_{i\neq w}|i\rangle \\
&= \frac{1}{\sqrt{N}}|w\rangle + \sqrt{\frac{N-1}{N}}\underbrace{\frac{1}{\sqrt{N-1}}\sum_{i\neq w}|i\rangle}_{|r\rangle} \\
&= \frac{1}{\sqrt{N}}|w\rangle + \sqrt{\frac{N-1}{N}}|r\rangle \\
&= \sin\theta|w\rangle + \cos\theta|r\rangle,
\end{aligned}
$$

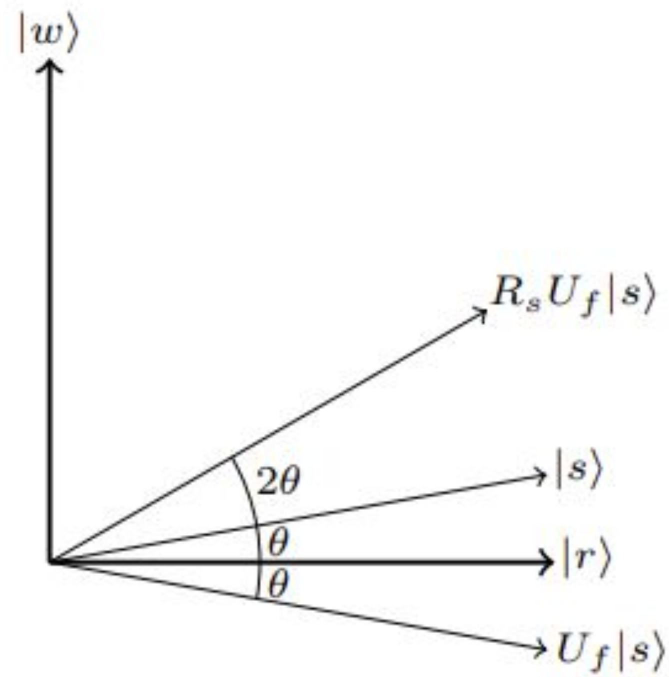
Illustration of Phase Flip

$$
\begin{aligned}
U_f|s\rangle &= (-1)^1 \sin\theta|w\rangle + (-1)^0 \cos\theta|r\rangle \\
&= -\sin\theta|w\rangle + \cos\theta|r\rangle.
\end{aligned}
$$



Reflection about $|s\rangle$

$$
\begin{aligned}
R_s &= 2\,|s\rangle\langle s| - I \\
R_s\,|\psi\rangle &= 2\,|s\rangle\langle s|\,|\psi\rangle - |\psi\rangle \\
&= 2\,|s\rangle\langle s|\psi\rangle - |\psi\rangle \\
&= \begin{cases} |\psi\rangle & \text{if } |\psi\rangle = |s\rangle \\ -|\psi\rangle & \text{if } |\psi\rangle \perp |s\rangle \end{cases}
\end{aligned}
$$



Elucidating the Reflection

$$
\begin{aligned}
|s\rangle &= |+\rangle^{\otimes n} = H^{\otimes n}\,|0^{\otimes n}\rangle \\
&= H\,|0\rangle \dots H\,|0\rangle \\
\implies \langle s| &= \langle 0|\,H^{\dagger} \dots \langle 0|\,H^{\dagger} \\
&= \langle 0|\,H \dots \langle 0|\,H \qquad (H^{\dagger} = H) \\
&= \langle 0^{\otimes n}|\,H^{\otimes n}
\end{aligned}
$$

$$
\begin{aligned}
I_n &= I \otimes \dots \otimes I \\
&= HH \otimes \dots \otimes HH \\
&= (H \otimes \dots \otimes H)(H \otimes \dots \otimes H) \\
&= H^{\otimes n} H^{\otimes n}
\end{aligned}
$$

Rewriting R_s

$$
\begin{aligned}
R_s &= 2\left|s\right\rangle\left\langle s\right| - I \\
&= 2(H^{\otimes n}\left|0^{\otimes n}\right\rangle)(\left\langle 0^{\otimes n}\right| H^{\otimes n}) - H^{\otimes n}H^{\otimes n} \\
&= H^{\otimes n}(2\left|0^{\otimes n}\right\rangle\left\langle 0^{\otimes n}\right| - I)H^{\otimes n}
\end{aligned}
$$

$$R_s = H^{\otimes n}R_0H^{\otimes n}$$

We define the gate $R_0 = 2\left|0^n\right\rangle\left\langle 0^n\right| - I$

Analysis of R_0

What does this gate do?

$$R_0 = 2\left|0^n\right\rangle\left\langle 0^n\right| - I$$

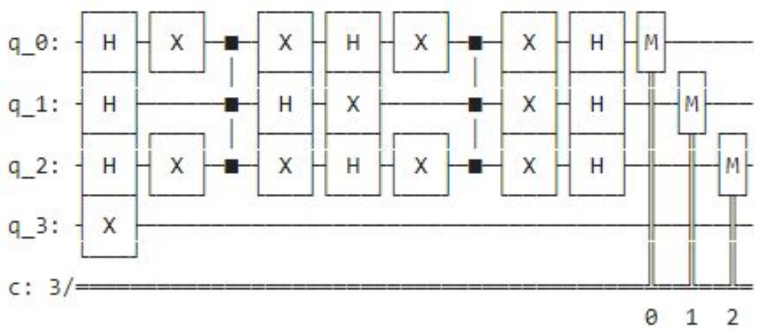
$$R_0\left|\psi\right\rangle = 2\left|0^n\right\rangle\left\langle 0^n|\psi\right\rangle - \left|\psi\right\rangle$$

$$= \begin{cases} \left|\psi\right\rangle & \text{if } \left|\psi\right\rangle = \left|0^n\right\rangle \\ -\left|\psi\right\rangle & \text{if } \left|\psi\right\rangle \perp \left|0^n\right\rangle \end{cases}$$

Let’s Check out the code!

```
for i in range(10):
  main()
  print('\n')
```

```
Secret bit sequence: [0, 1, 0]
Circuit:
```



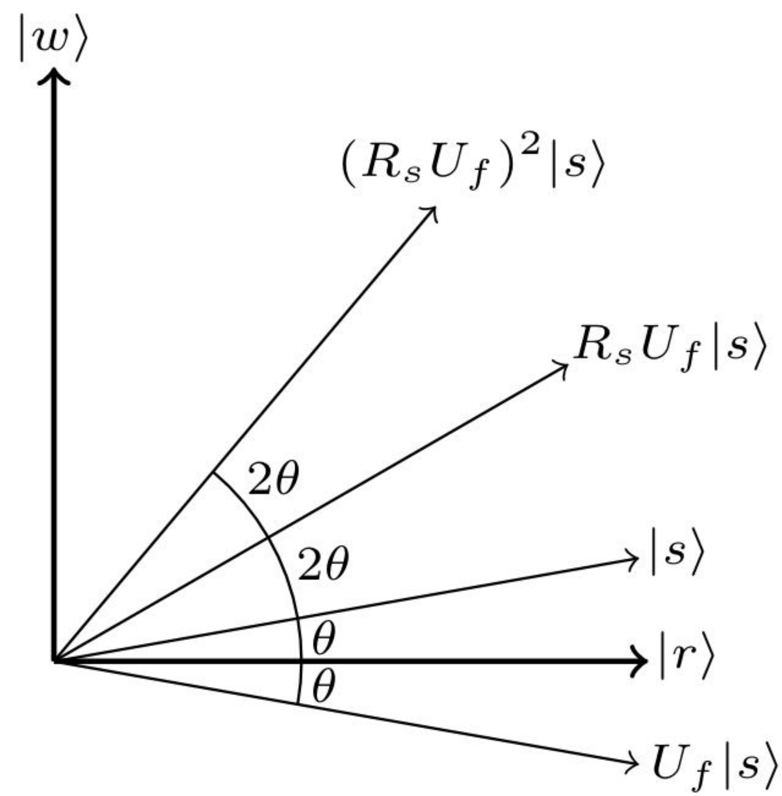

```
Sampled results:
{'110': 87, '100': 90, '101': 79, '000': 89, '010': 1469, '001': 83, '011': 78, '111': 73}
Most common bitstring: 010
Found a match: True
```

https://colab.research.google.com/drive/1Tr7RAmyDLTwqNIuB701jgWRl3B2CJCNi#scrollTo=K-6feT241Llj

Runtime Analysis

Perform t rotations until we reach w.

How do we get t?



Runtime Analysis

$$
\begin{aligned}
\theta + t(2\theta) &= \frac{\pi}{2} \\
t(2\theta) &= \frac{\pi}{2} - \theta \\
t &= \frac{\pi}{4\theta} - \frac{1}{2} \\
t &\approx \frac{\pi}{4}\sqrt{N} - \frac{1}{2} \\
\implies t &\in O(\sqrt{N})
\end{aligned}
$$

$$\sin\theta = \frac{1}{\sqrt{N}}$$

$$\theta = \arcsin\left(\frac{1}{\sqrt{N}}\right)$$

We assume that N is sufficiently large

$$\theta \approx \frac{1}{\sqrt{N}}$$

Open Questions

Grover's Algorithm, as presented, seems to require "knowing" the solution beforehand.

How can we encode data bases in a quantum setting?

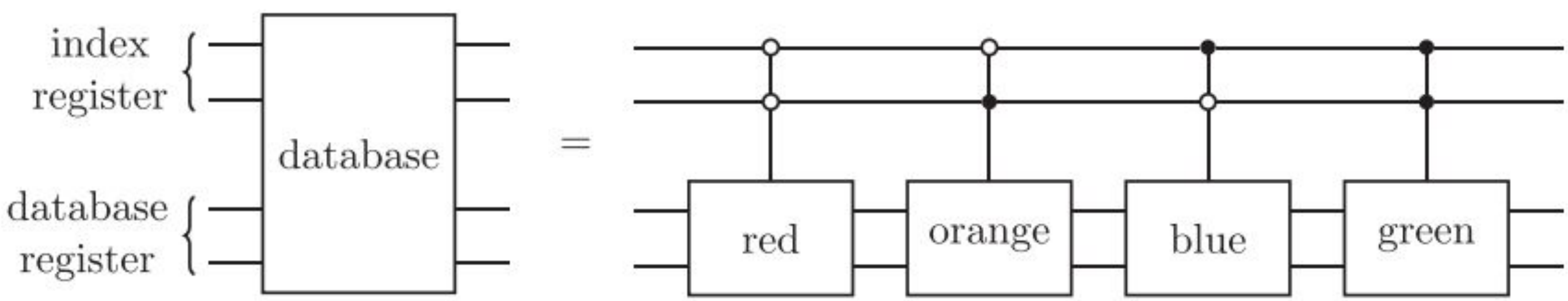
Does the additional overhead of encoding the database diminish the quadratic speed-up of the search?

Name	Phone Number
Alice	314-1592
Bob	271-8281
Charlie	105-4571
Dave	885-4187
Eve	125-6637
Frank	299-7924
Grace	729-7352
⋮	⋮
Zoe	200-2319

Encoding the Database

To encode our database, we need an additional n qubits: n for our index and n for our database values.

Via a few more circuits, the index of the desired value, despite being unknown to the user, is marked with a phase flip. Grover's Algorithm can proceed from there.



References

Introduction to Classical and Quantum Computing - Thomas G. Wong

Quantum Computing: Applied Approach - Jack D. Hidary

“Searching a Quantum Database with Grover's Search Algorithm” - Ben Kain

IBM Qiskit API

IBM Quantum Composer

ChatGPT