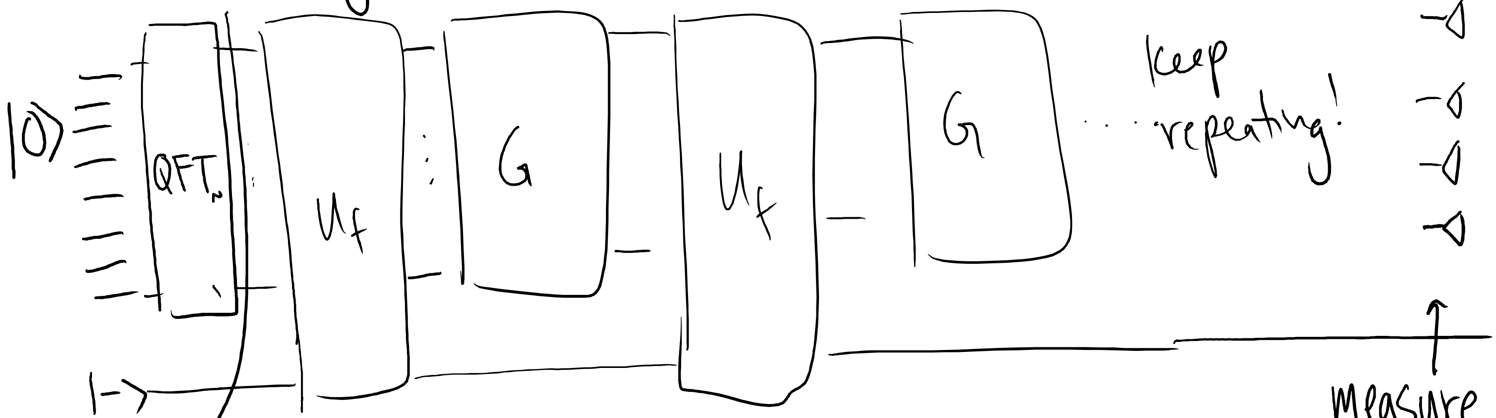
S. KIMMEL

Quantum Alg: (Grover's Algorithm)



measure in standard basis

$|\psi_1\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$

phase kickback

$|x\rangle|-\rangle \rightarrow (-1)^{f(x)}|x\rangle|-\rangle$

$U_f: \mathbb{I} - 2|s\rangle\langle s|$ (applies -1 to $|s\rangle$, $+1$ to all other standard basis states)

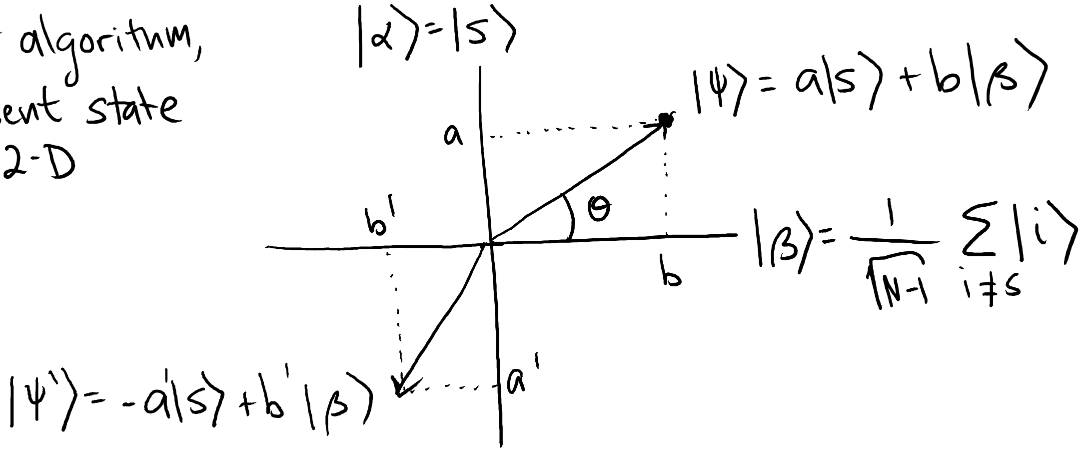
$$U_f = \left(\sum_i |i\rangle\langle i|\right) - 2|s\rangle\langle s| = \left(\sum_{i \neq s} |i\rangle\langle i|\right) - |s\rangle\langle s|$$

$G_f: -\mathbb{I} + 2|\alpha\rangle\langle\alpha|$ where $|\alpha\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle$

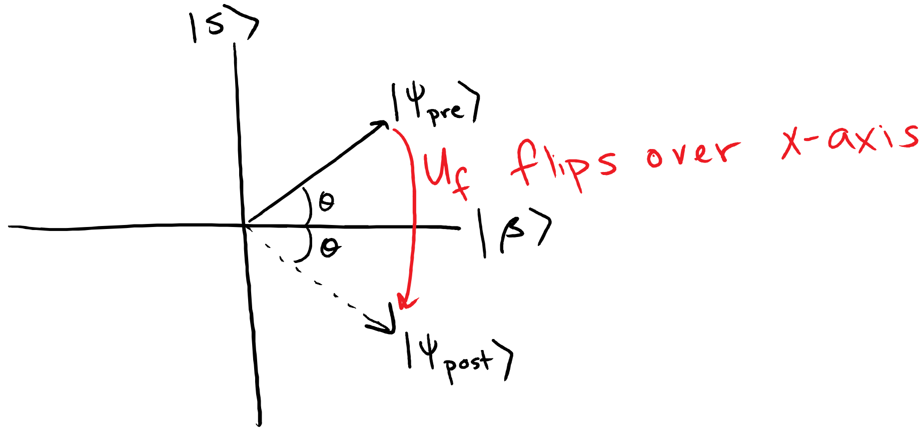
applies $+1$ to $|\alpha\rangle$ and -1 to all orthogonal states

S.KIMMEL

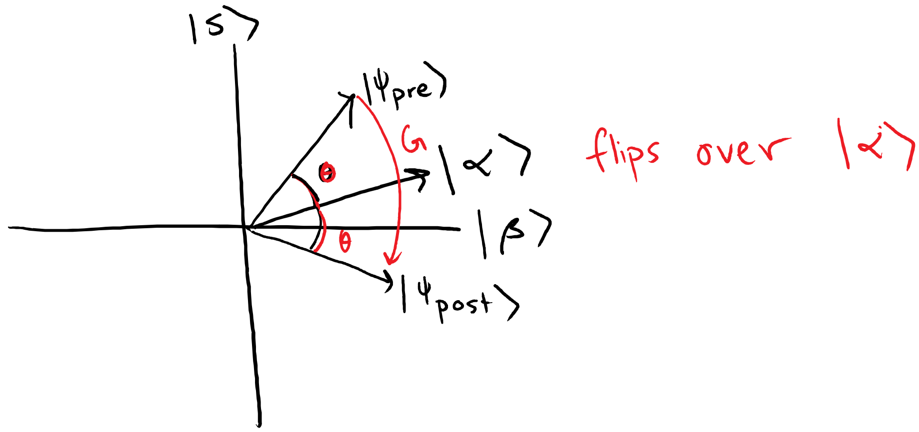
Throughout algorithm, can represent state using a 2-D axis



Effect of U_f: Adds -1 phase to $|s\rangle$

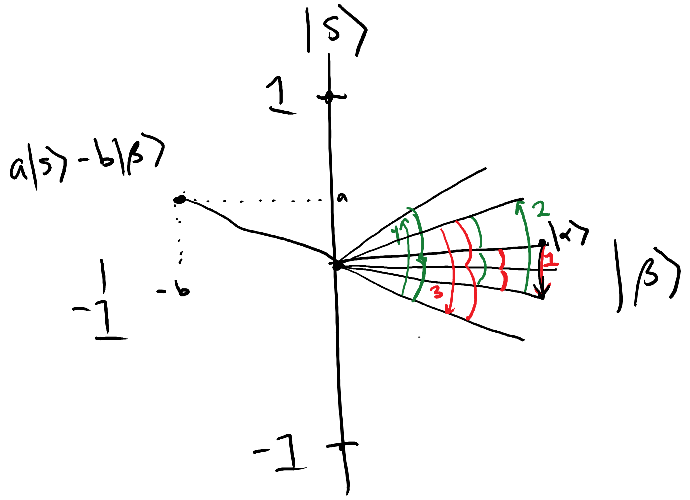


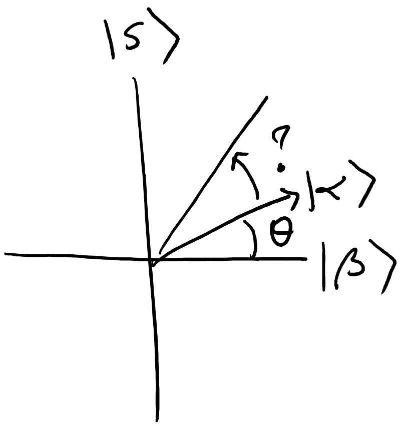
Effect of G: Adds -1 phase to anything orthogonal to $|\alpha\rangle$



S.KIMMEL

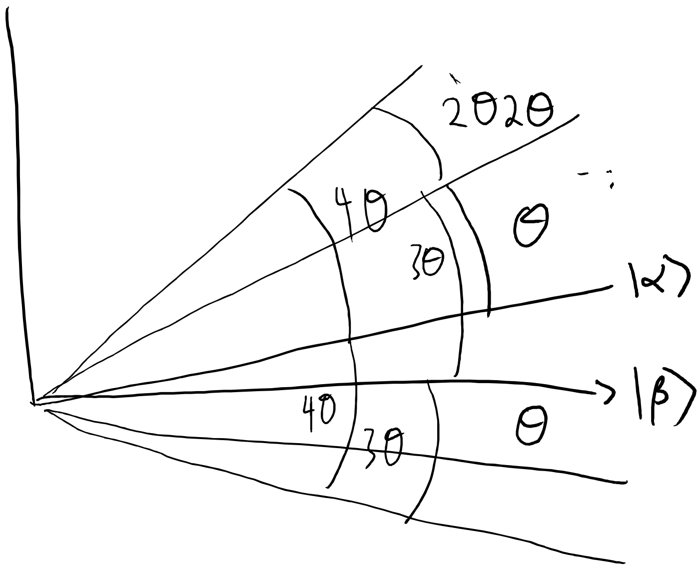
Effect of repeated $U_f, G, U_f, G, \ldots$



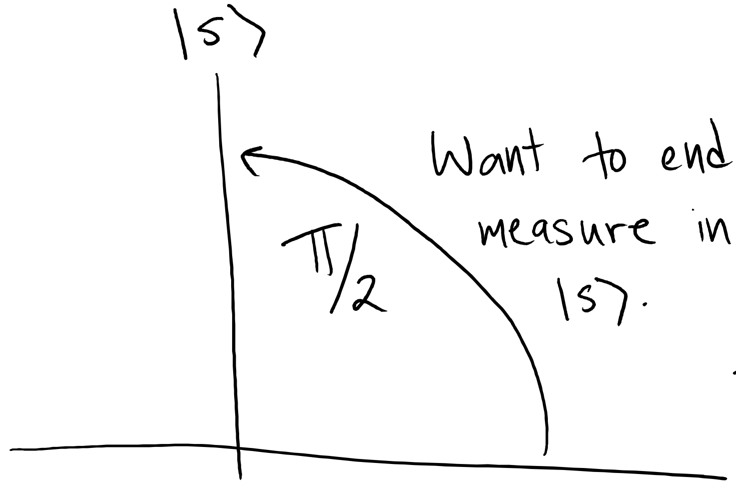


Q: If the initial angle between $|\alpha\rangle$ and $|\beta\rangle$ is θ, how much is the state shifted after U_f and G are each applied once?

S.KIMMEL



Increases by 2θ each round



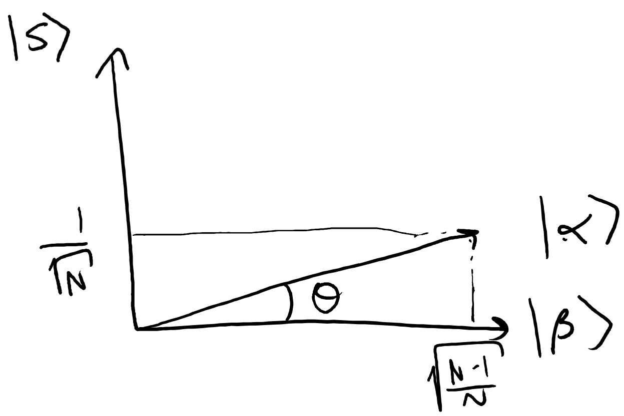
Want to end up at $|s\rangle$, so when measure in standard basis, get outcome $|s\rangle$.

Total: $\pi/2$

Each Step: 2θ

steps: $O\left(\frac{1}{\theta}\right)$

What is θ?



$$|\alpha\rangle = \sum_i \frac{1}{\sqrt{N}}|i\rangle = \frac{1}{\sqrt{N}}|s\rangle + \frac{\sqrt{N-1}}{\sqrt{N}}|\beta\rangle$$

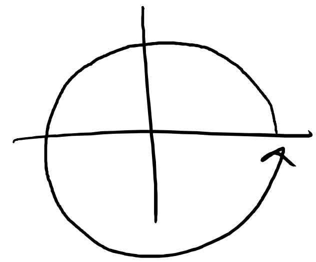
$$\theta = \arctan\left(\frac{1/\sqrt{N}}{\sqrt{N-1}/\sqrt{N}}\right)$$

$$= \arctan\left(\frac{1}{\sqrt{N}}\right) \approx \frac{1}{\sqrt{N}}$$

S. KIMMEL

\# of steps: $O(\sqrt{N})$. Square root speed-up vs. classical!

* If run too long, rotate back to start!



Applications

- Any classical algorithm that succeeds with prob $p \rightarrow$ can create quantum version that succeeds with prob $\sqrt{p}$.

ex: One of the best 3-SAT algorithms: $p = \left(\frac{3}{4}\right)^n$

Need to run algorithm $O\left(\left(\frac{4}{3}\right)^n\right)$ times to be successful.

$\Rightarrow$ Quantum Version: $O\left(\left(\frac{4}{3}\right)^{n/2}\right)$ time

Square Root Speed Up is less impressive than Shor's exponential speed up, but

ex: $n = 200$ (200 cities in traveling salesman) $\Big\}$ 10^{24} classical time, 10^{12} quantum time

1 GHz processor: 31 million years (classical)
16 minutes (quantum)