Quantum Computing

SPECIFICALLY, GROVER'S ALGORITHM

(HEAVILY INSPIRED BY 3BLUE1BROWN'S DISCUSSION ON THE TOPIC)

Misconceptions

Quantum Computing is NOT…

- Always faster than classical computers
- Performing operations on every state of a bit set at once

More on this later…

Qubits

- "Quantum bits" that produce a 0 or 1 when read
- State of a qubit = superposition the two possible outcomes
- Built out of actual quantum phenomena:
  - Electron spin
  - Superconducting circuits
  - Trapped ions

Bits vs. Qubits

Bits

- Have a state of 0 or 1
- Yield a 0 or 1 when measured
- Measured value is equal to the physical state of the bit

Qubits

- Have a state that is a **superposition** of 0 and 1
- Also yield a 0 or 1 when measured
- Superposition itself is not measured

Probability Distributions

(2 qubit system)

Due to superposition:

- Each sequence has certain probability of being measured
- Algorithms manipulate probabilities

Superposition collapses when measurement occurs

Sequence	Probability
00	25%
01	25%
10	25%
11	25%

Sequence	Probability
00	9%
01	73%
10	9%
11	9%

Sequence	Probability
00	5%
01	85%
10	5%
11	5%

Sequence	Probability
00	0%
01	100%
10	0%
11	0%

THE STATE VECTOR

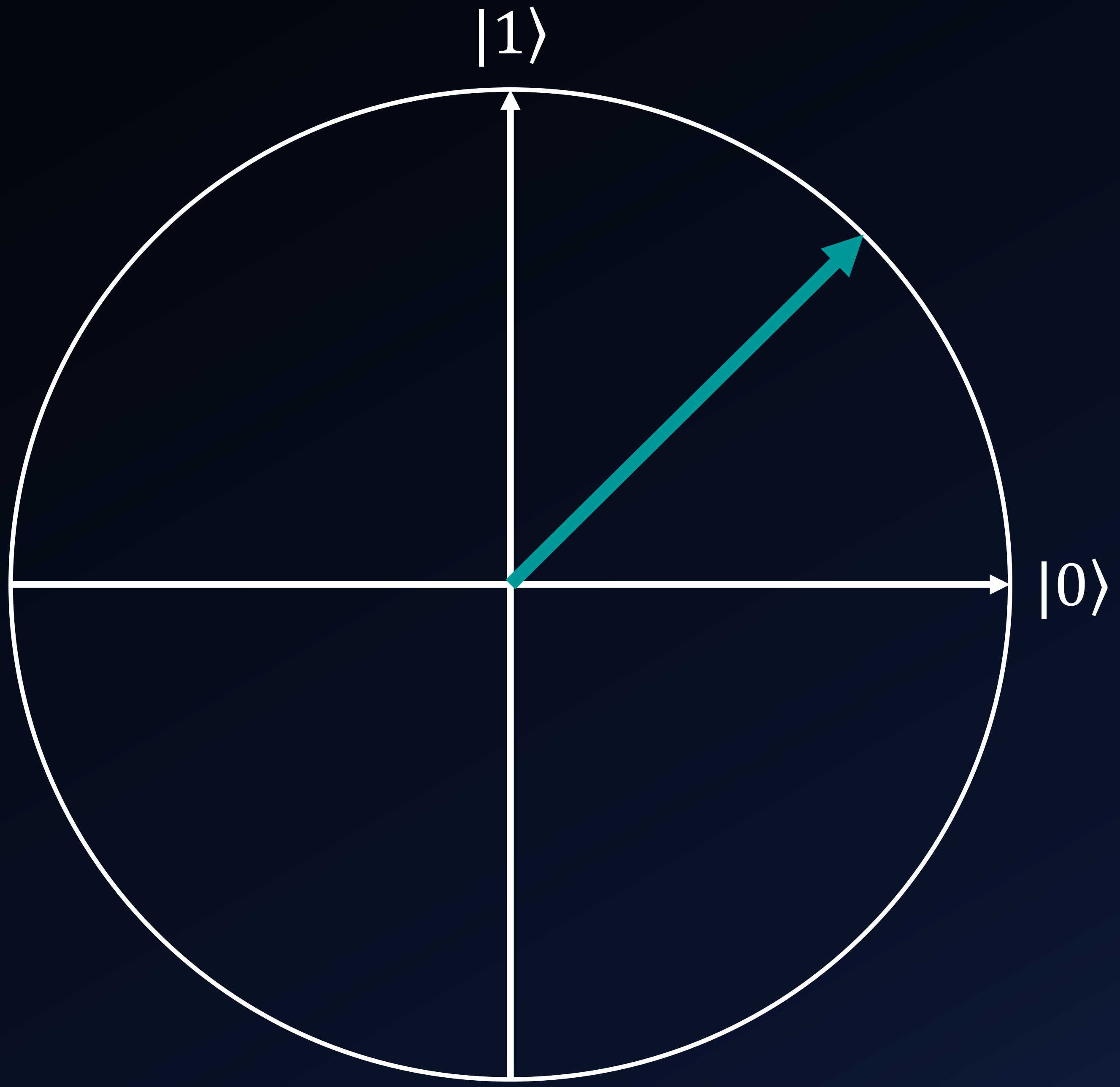
Closely related to the probability distributions in the computer

- Unit vector (draws out a unit hypersphere)
- 2^k components for k bits (one per possible sequence)
- Square of component magnitude = probability of corresponding sequence

$$|\psi\rangle = \begin{bmatrix} +0.71 \\ +0.71 \end{bmatrix} = 0.71\,|0\rangle + 0.71\,|1\rangle$$

$$0.71^2 \approx 50\%$$

50% chance of $|0\rangle$ and 50% chance of $|1\rangle$



Quantum Gates

Operations on bits/sequences of qubits that usually apply some transformation to the state vector

I.e. the Pauli-X gate rotates a single qubit state vector along the X axis

Other special gates/operations can be constructed out of combinations of these gates in certain ways

Quantum gate	Pauli-X (X)	Pauli-Y (Y)	Pauli-Z (Z)	Hadamard (H)	Controlled-Z (CZ)
Mathematical form	$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$Y = \begin{bmatrix} 0 & -\iota \\ \iota & 0 \end{bmatrix}$	$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$	$CZ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$
Graph representation	X or $\oplus$	Y	Z	H	Z or
Quantum gate	Rotational single-qubit gate along X-axis (RX(θ))	Rotational single-qubit gate along Y-axis (RY(θ))	Rotational single-qubit gate along Z-axis (RZ(θ))	SWAP	Controlled-NOT (CNOT, CX)
Mathematical form	$RX(\theta) = \begin{bmatrix} \cos(\frac{\theta}{2}) & -\iota \sin(\frac{\theta}{2}) \\ -\iota \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{bmatrix}$	$RY(\theta) = \begin{bmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{bmatrix}$	$RZ = \begin{bmatrix} e^{-\iota\frac{\theta}{2}} & 0 \\ 0 & e^{\iota\frac{\theta}{2}} \end{bmatrix}$	$SWAP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$	$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Graph representation	RX(θ)	RY(θ)	RZ(θ)	or	X or $\oplus$

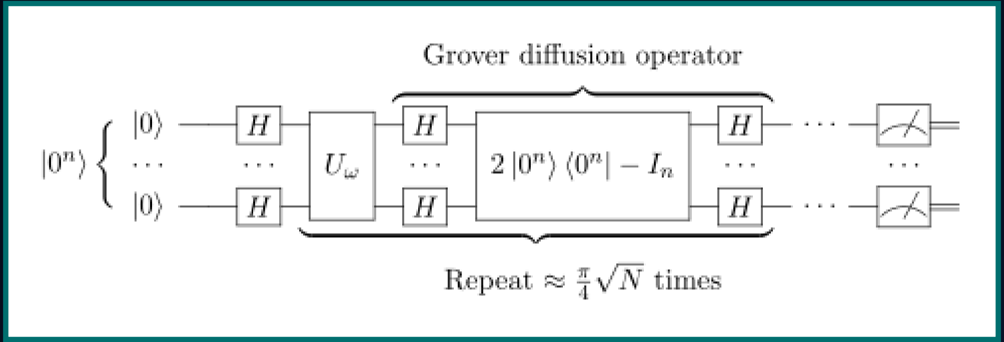
GROVER'S ALGORITHM

An **unstructured search algorithm** that runs in $O(\sqrt{n})$ time.

- Pinpoints a specific value in a data set where the value can be quickly checked for correctness (NP problems?)
  - I.e. finding answer to a Sudoku puzzle (assuming there's only one)
- Takes the same amount of time as a classical algorithm with n^2 inputs

Uses two operations:

- Z_f - flips the sign of the component for the desired outcome
- Z_{OR} - flips the state vector around the state prior to applying Z_f

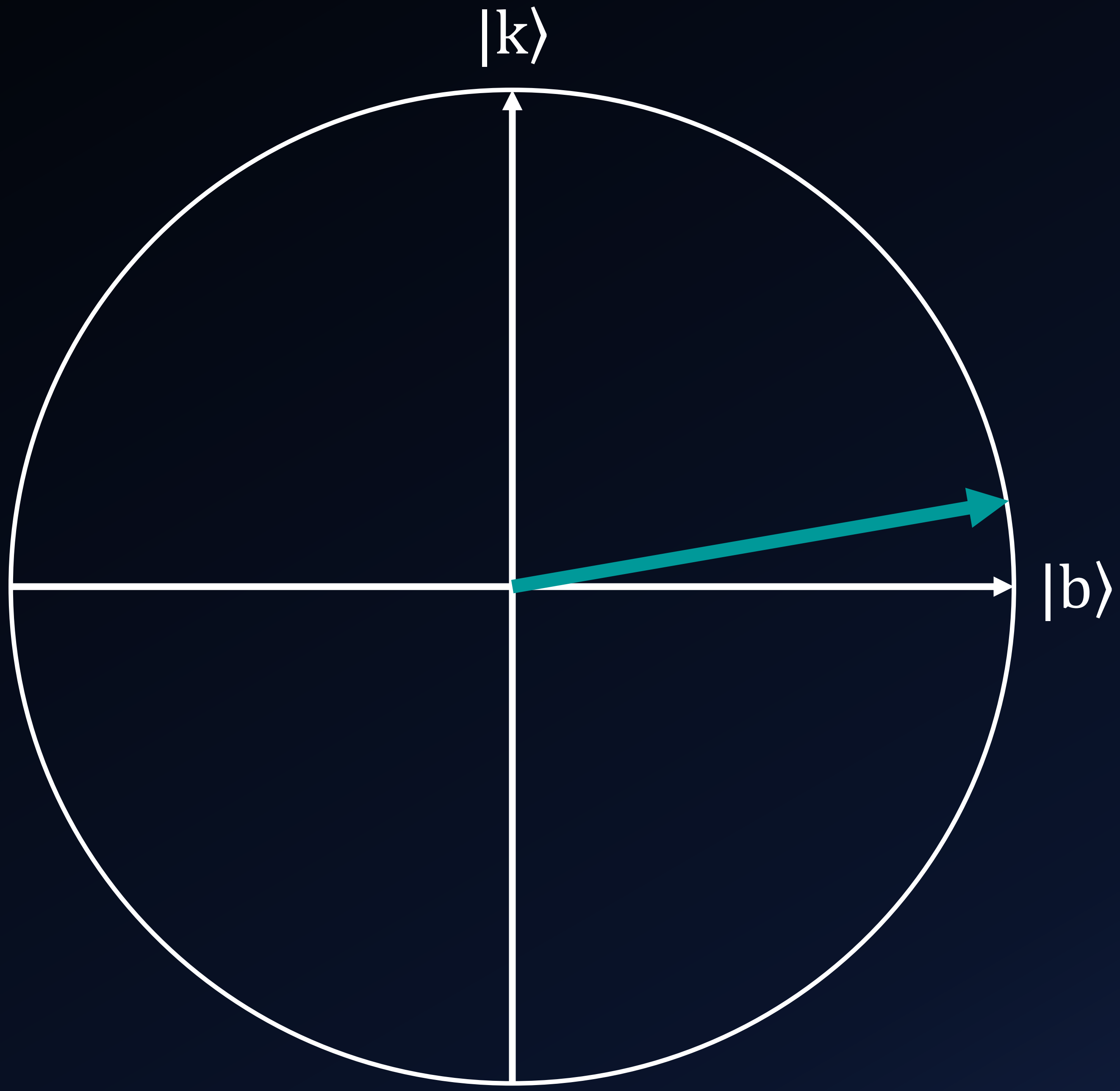


EXAMPLE

Apply Z_f and Z_{OR} - to an N-qubit system

Only a 2-dimensional slice of the system is shown – all operations only rotate the vector within this slice

(Assume $|k\rangle$ is the desired outcome)

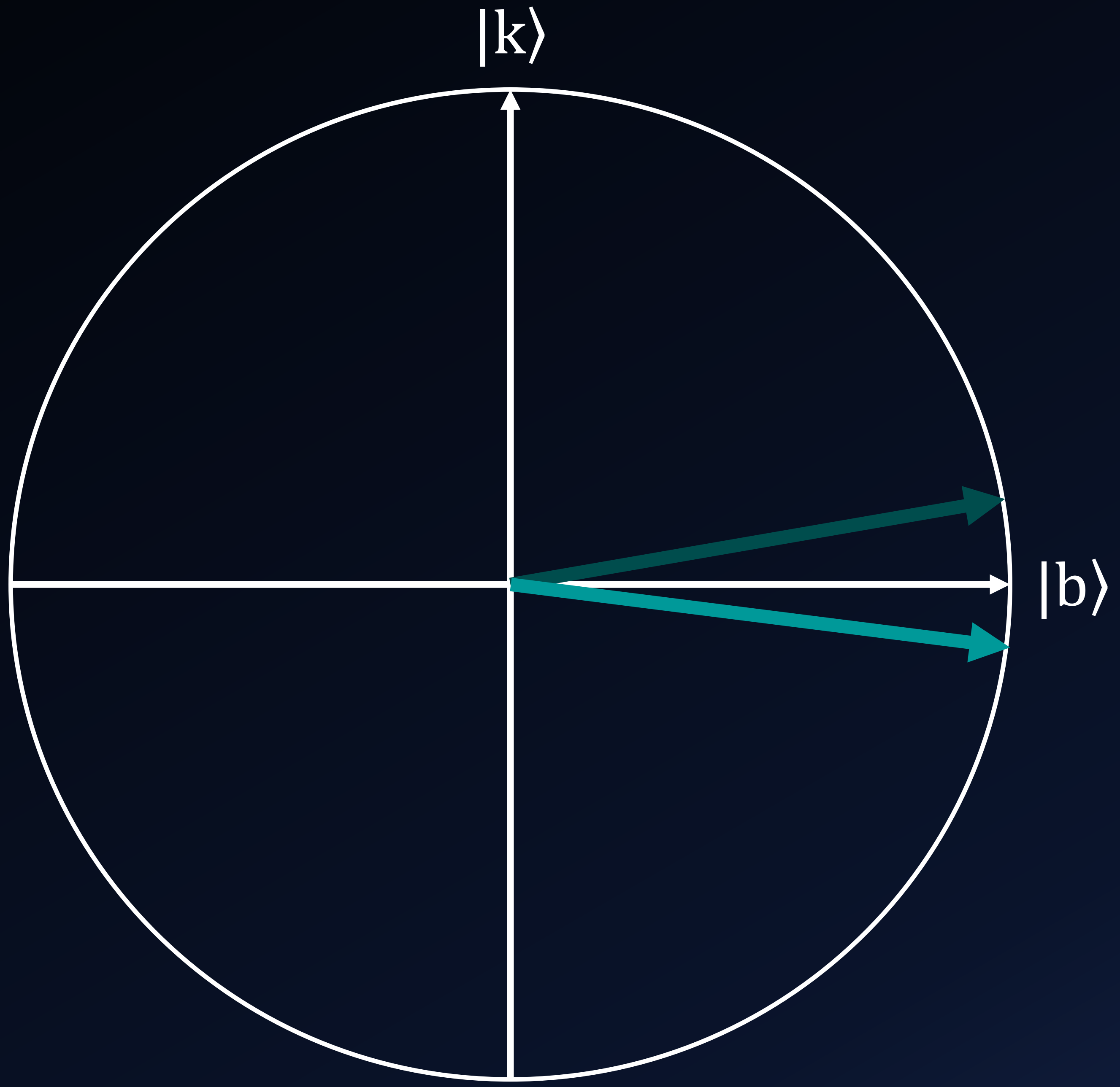


EXAMPLE

Apply Z_f and Z_{OR} - to an N-qubit system

Only a 2-dimensional slice of the system is shown – all operations only rotate the vector within this slice

(Assume $|k\rangle$ is the desired outcome)

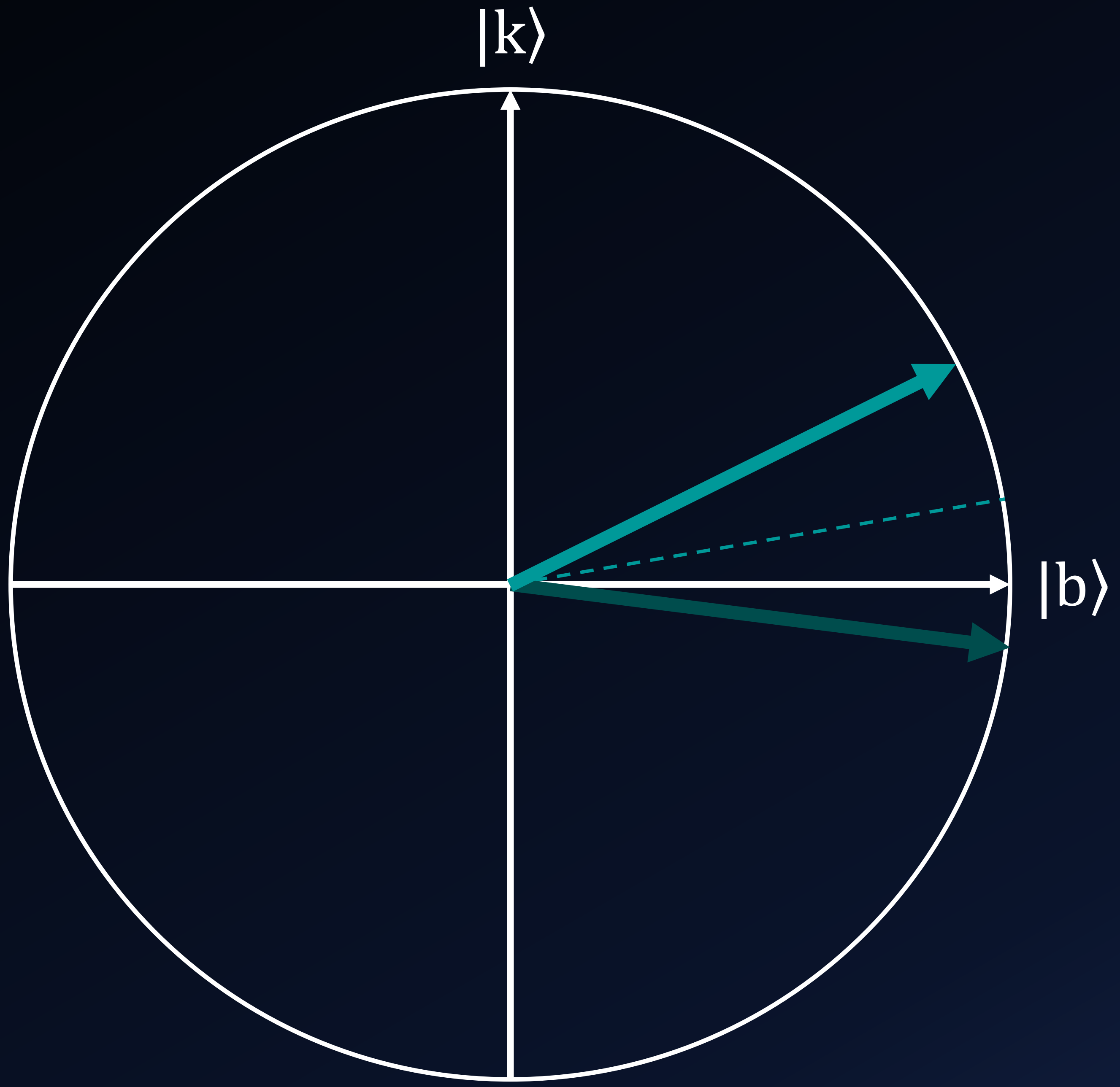


EXAMPLE

Apply Z_f and Z_{OR} - to an N-qubit system

Only a 2-dimensional slice of the system is shown – all operations only rotate the vector within this slice

(Assume $|k\rangle$ is the desired outcome)

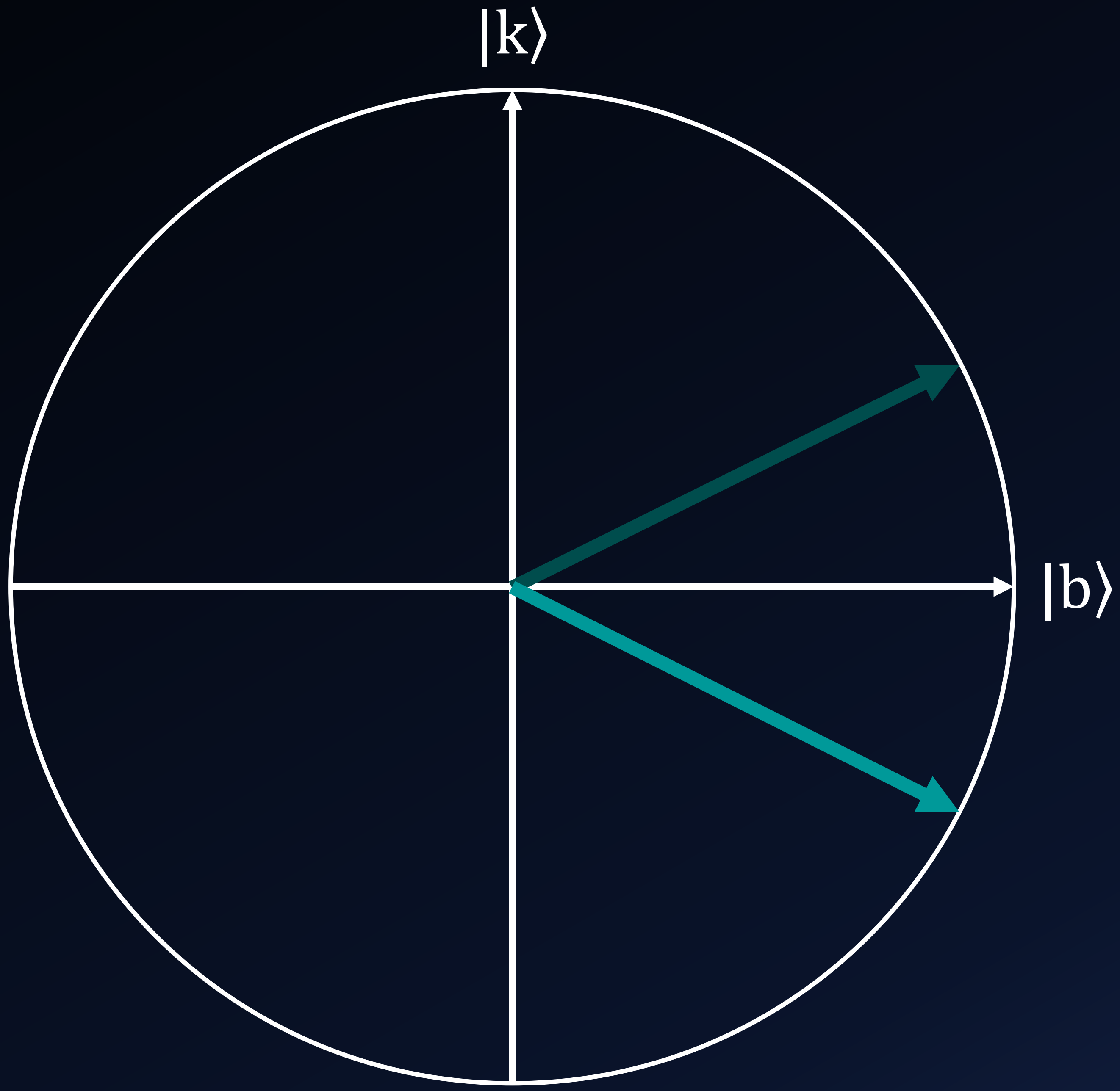


EXAMPLE

Apply Z_f and Z_{OR} - to an N-qubit system

Only a 2-dimensional slice of the system is shown – all operations only rotate the vector within this slice

(Assume $|k\rangle$ is the desired outcome)

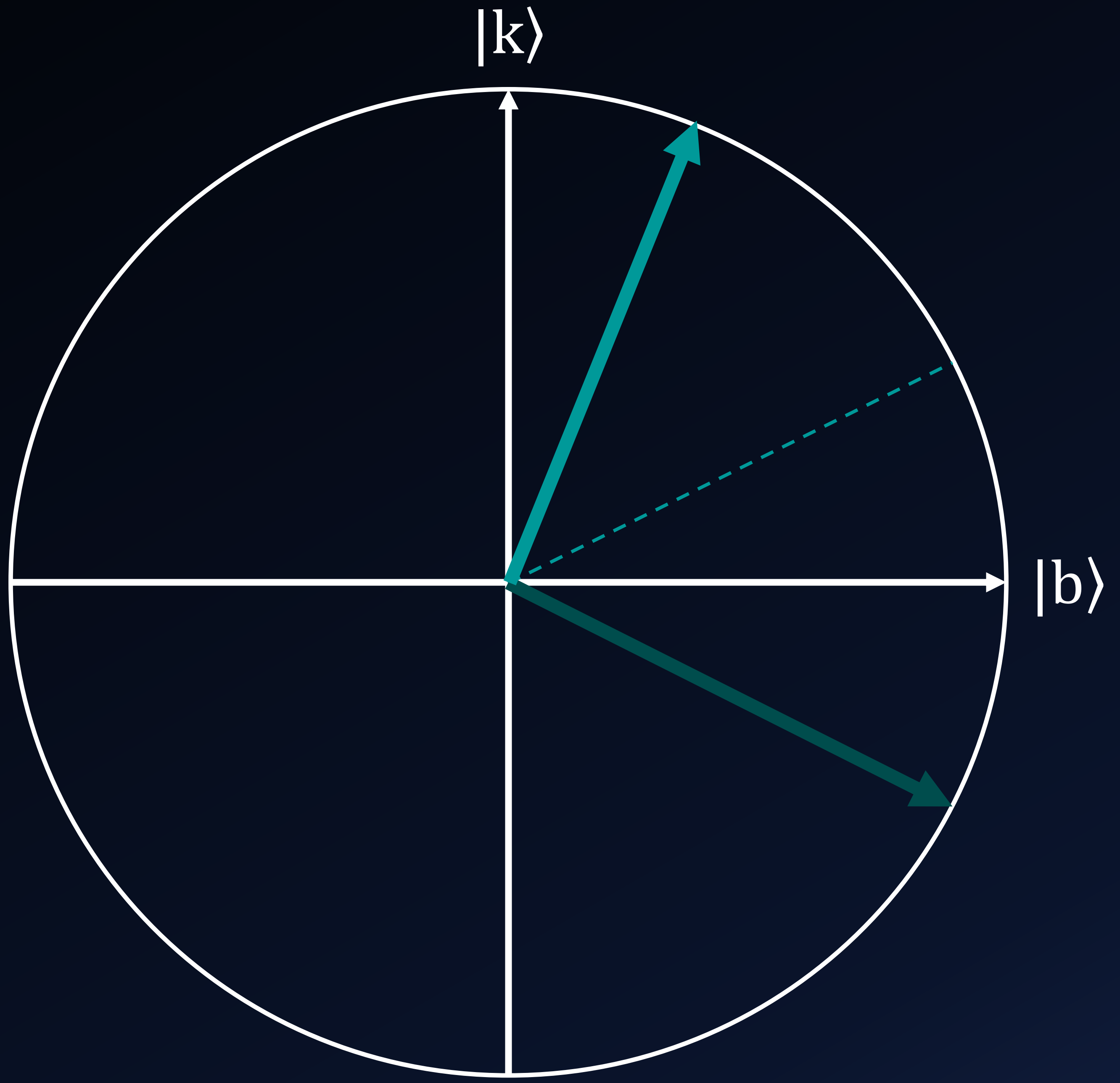


EXAMPLE

Apply $\mathrm{Z_f}$ and $\mathrm{Z_{OR}}$ - to an N-qubit system

Only a 2-dimensional slice of the system is shown – all operations only rotate the vector within this slice

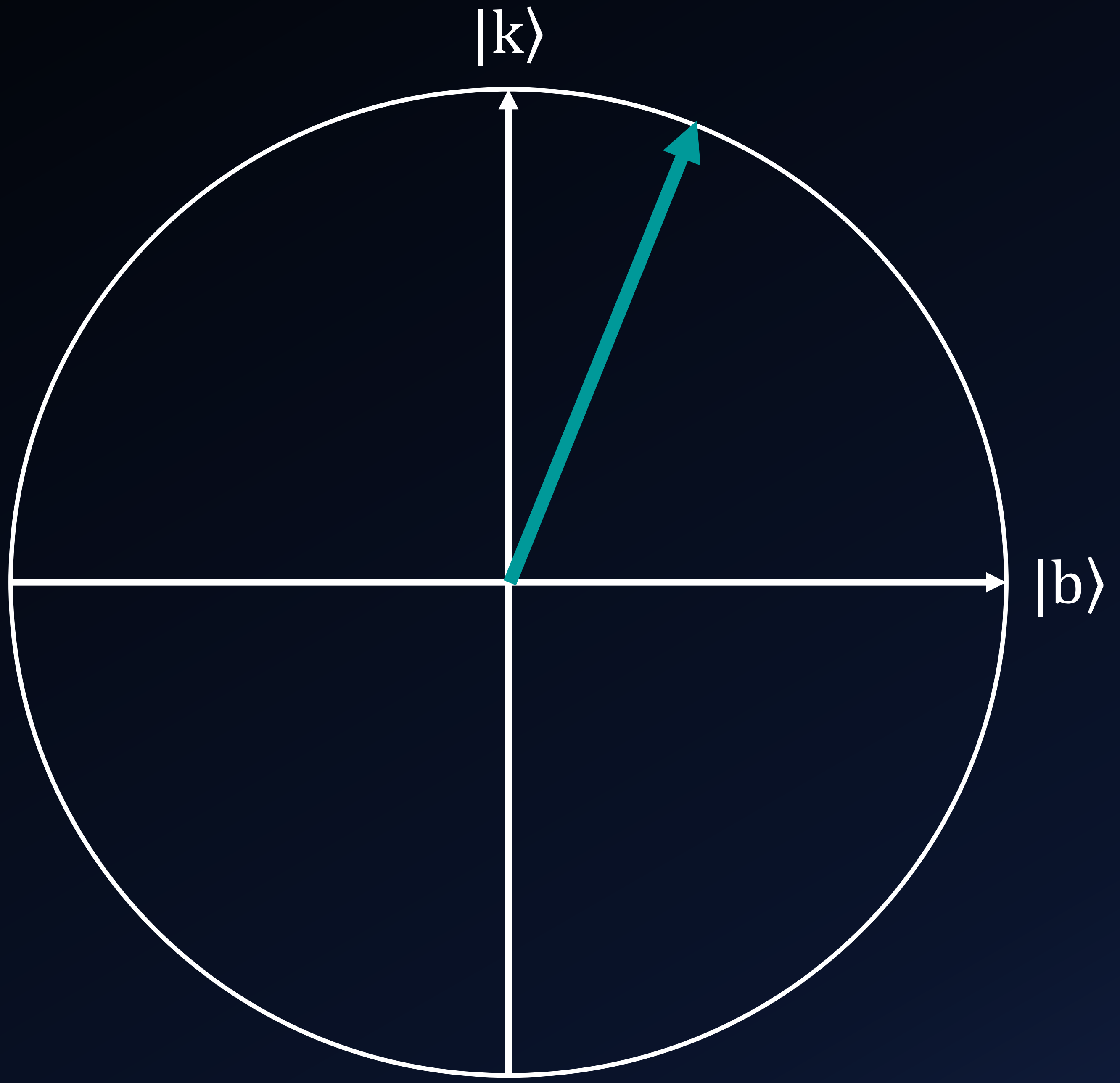
(Assume $|\mathrm{k}\rangle$ is the desired outcome)



EXAMPLE

The state vector is now much more closely aligned with the desired outcome

The vector thus has a much higher chance of collapsing into that outcome when measured



APPLICATIONS

More of a proof of concept than an algorithm with large benefits, but it can perform certain tasks quickly

- Breaking cryptography
- Searching unstructured datasets
- Pattern recognition
- Solving NP problems

Note: Grover’s algorithm can only perform a quadratic speedup compared to many classical algorithms.

Many other algorithms (i.e. Shor’s Algorithm) can perform an exponential speedup in comparison.

Final Thoughts

STILL OVERSIMPLIFIED

- Most actual programs rely on state vectors in the complex plane and other techniques too
  - Shor’s algorithm is a big example of this

HARD TO ACHIEVE

- Actual quantum computers must account for errors in qubit output due to noise, decoherence, etc.
- Majority of work in quantum computing has been solving those issues

Thank you!

QUESTIONS?