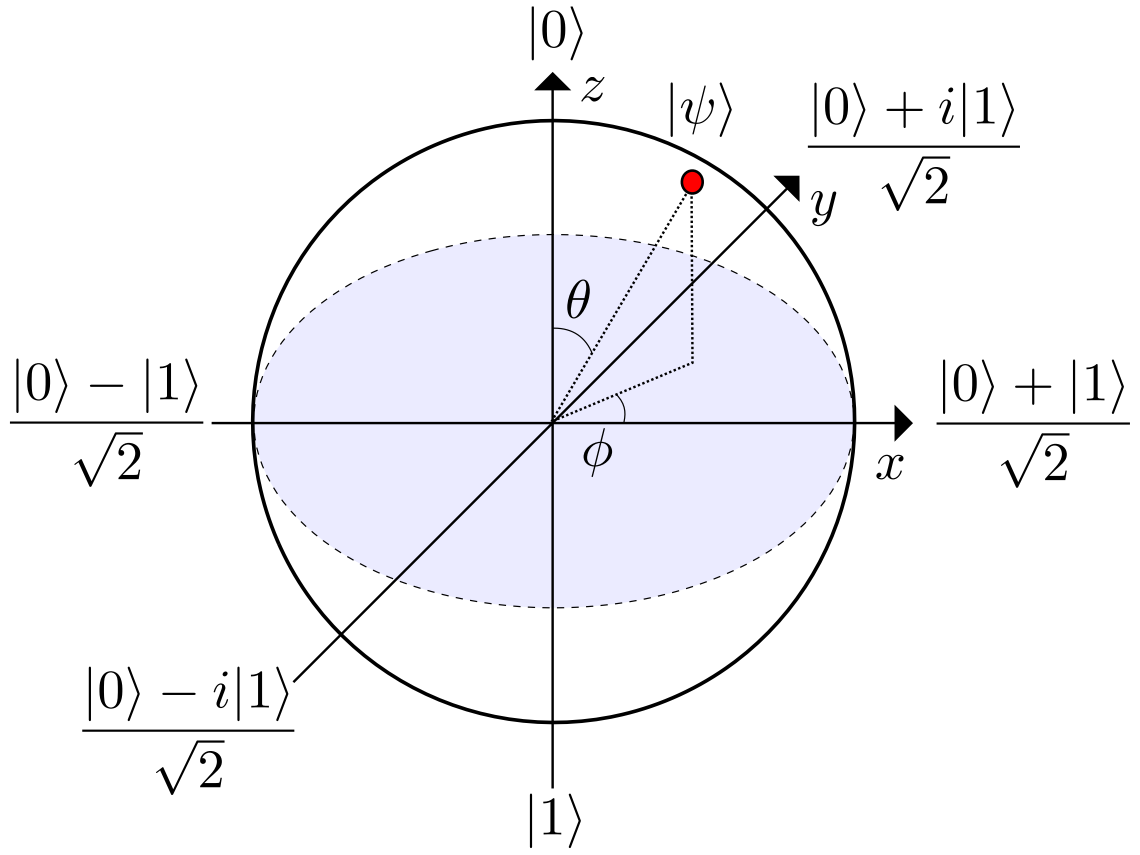
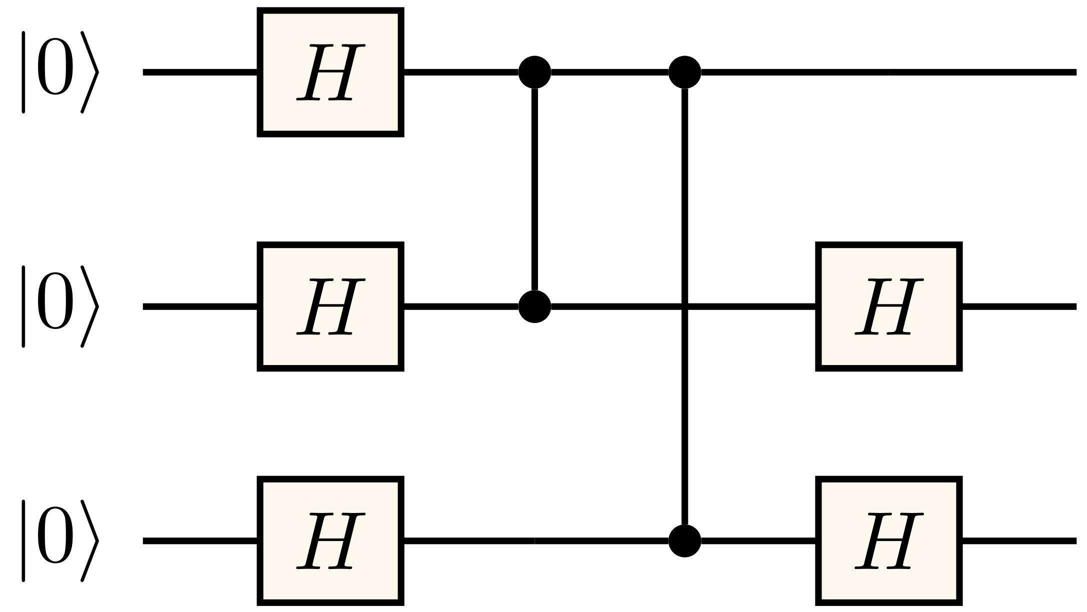
WACQT | Wallenberg Centre for Quantum Technology



Quantum states, qubits, logic gates, and algorithms



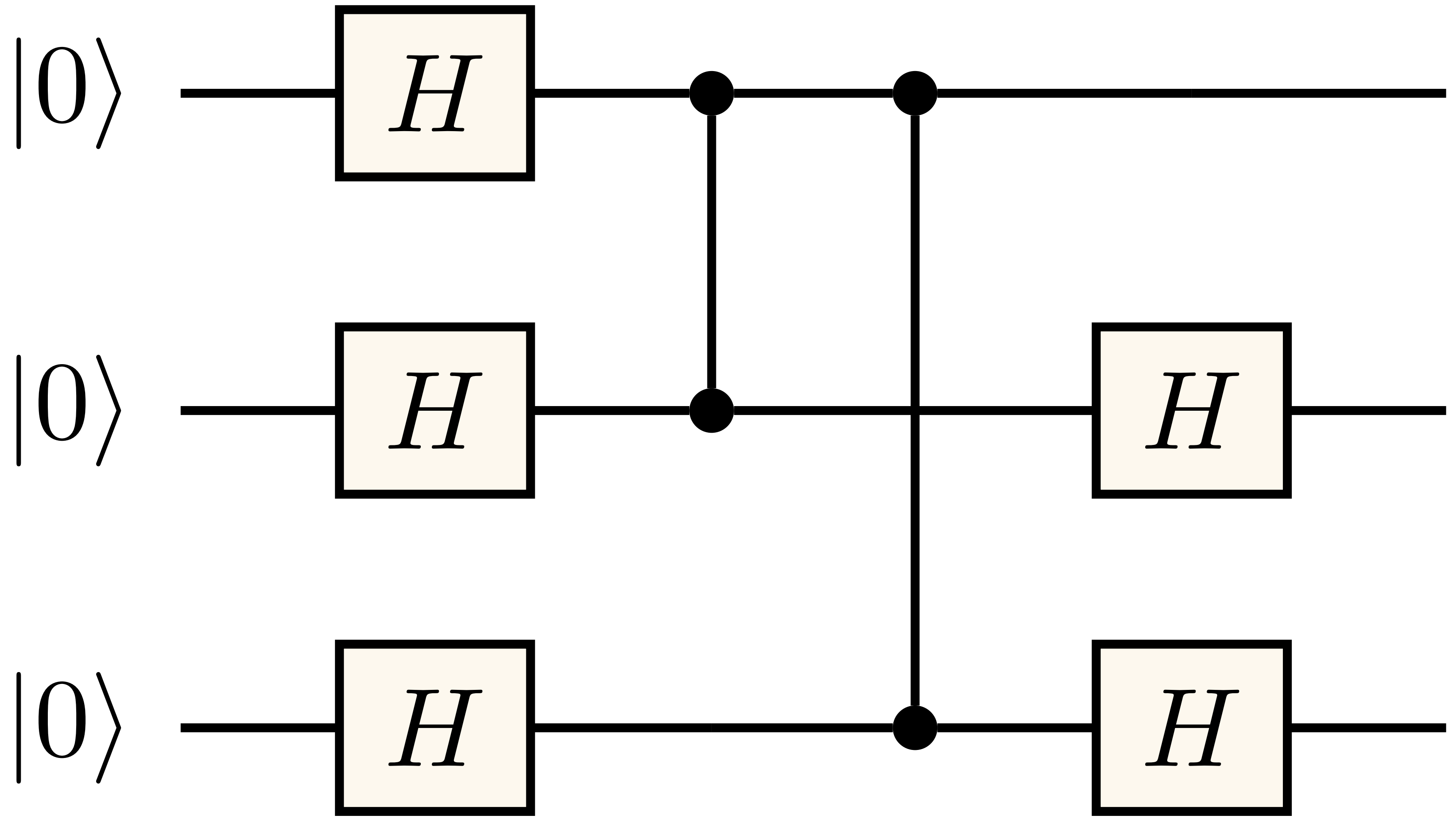
Dr. Anton Frisk Kockum

Researcher, Scientific Coordinator

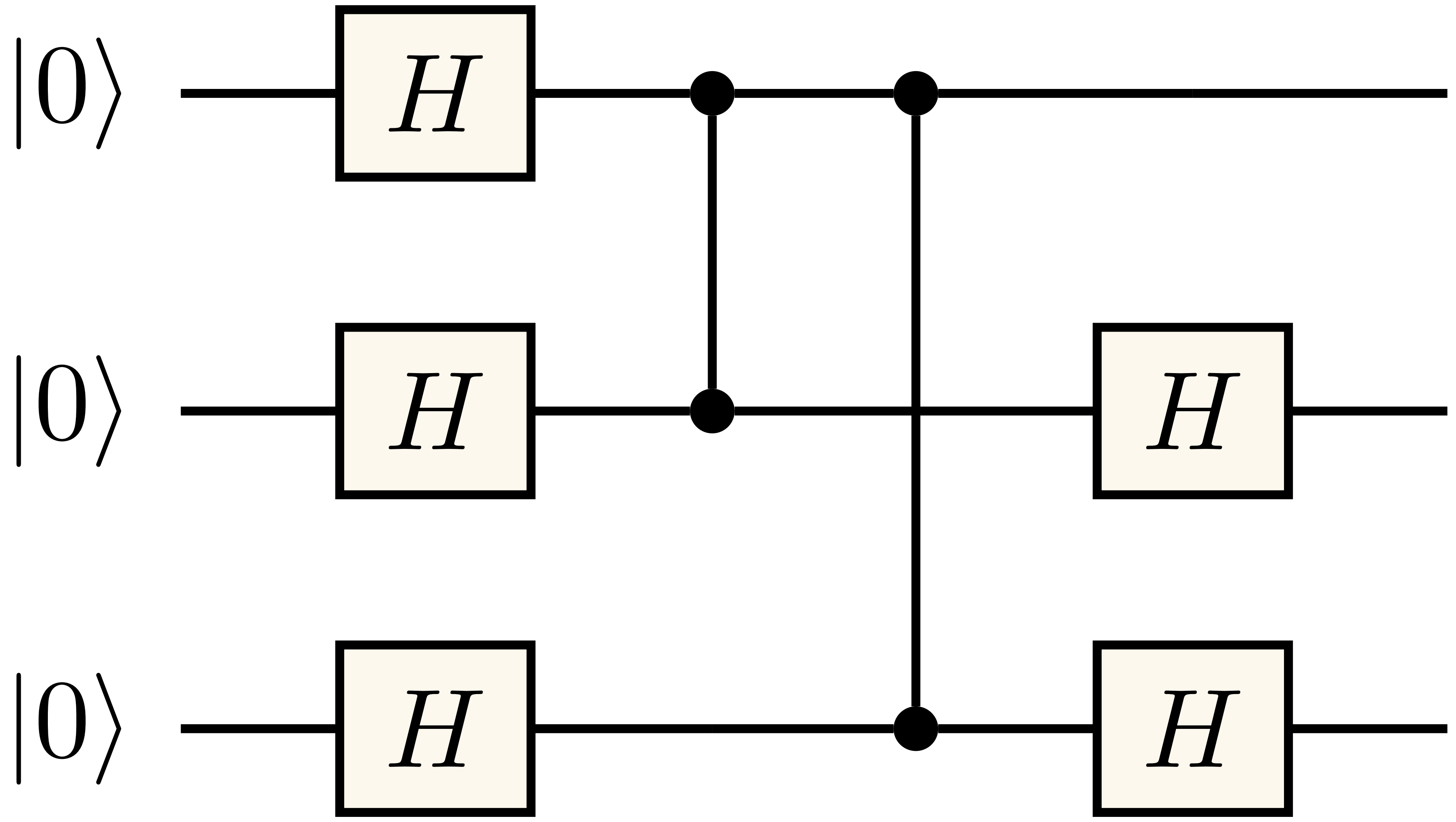
Outline

- Components of a quantum circuit
- Quantum bits
- Single-qubit gates
- Multi-qubit gates
- Universal gate sets
- The Solovay-Kitaev theorem
- Quantum algorithms and compilation
- Summary

Components of a quantum circuit

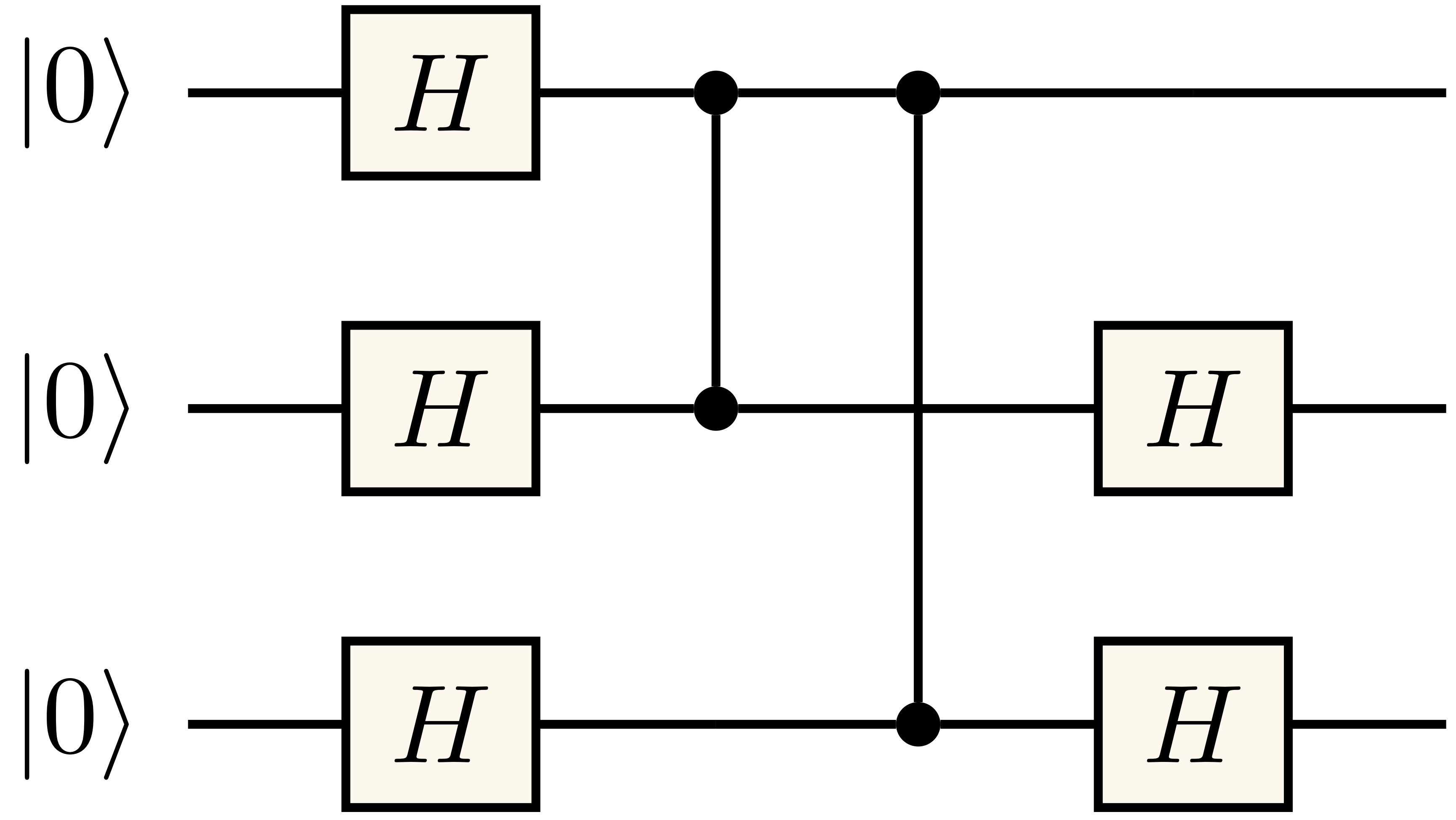


Components of a quantum circuit



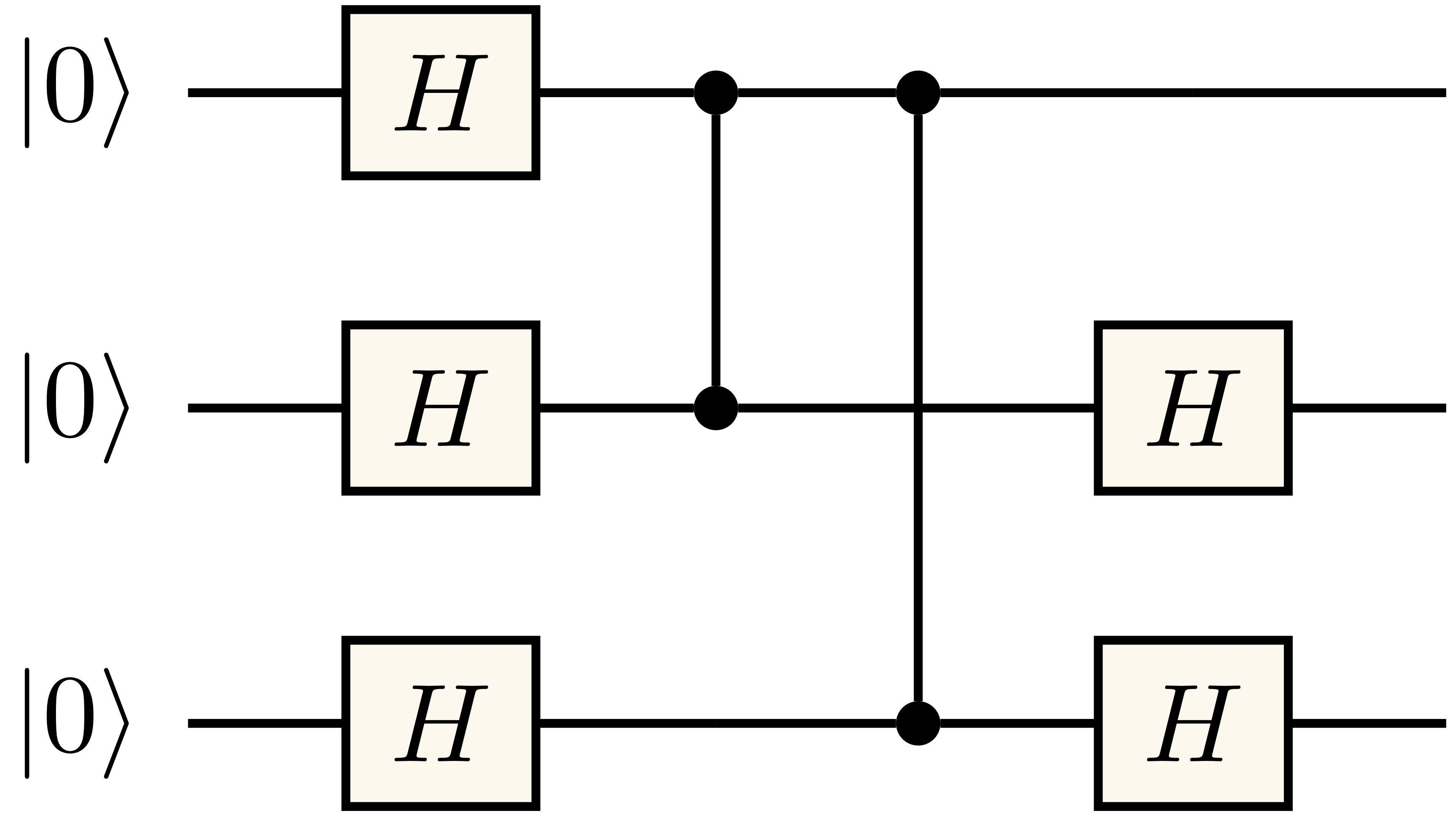
- Represent data: quantum bits

Components of a quantum circuit



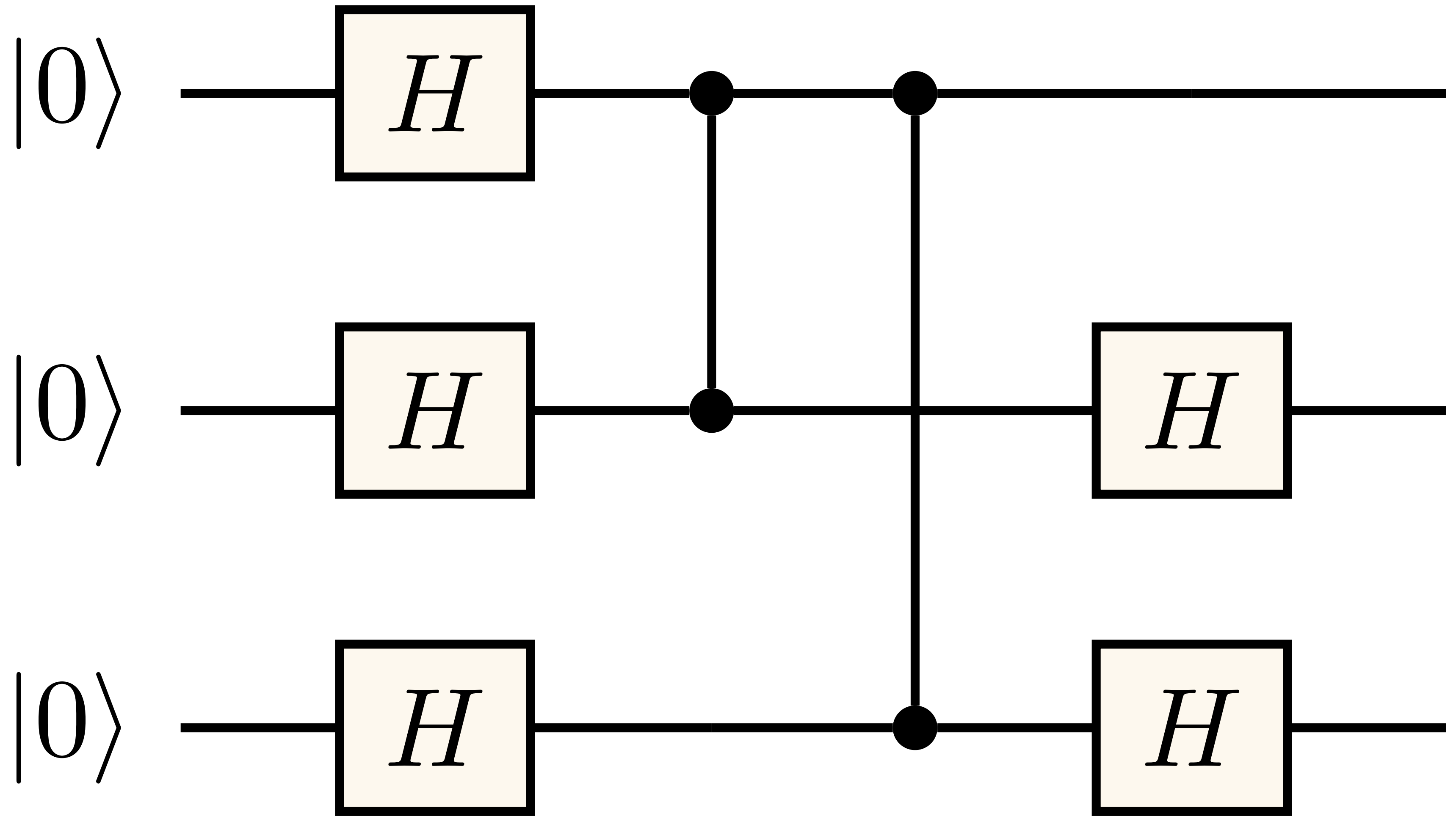
- Represent data: quantum bits
- Initialize the computation: state preparation

Components of a quantum circuit



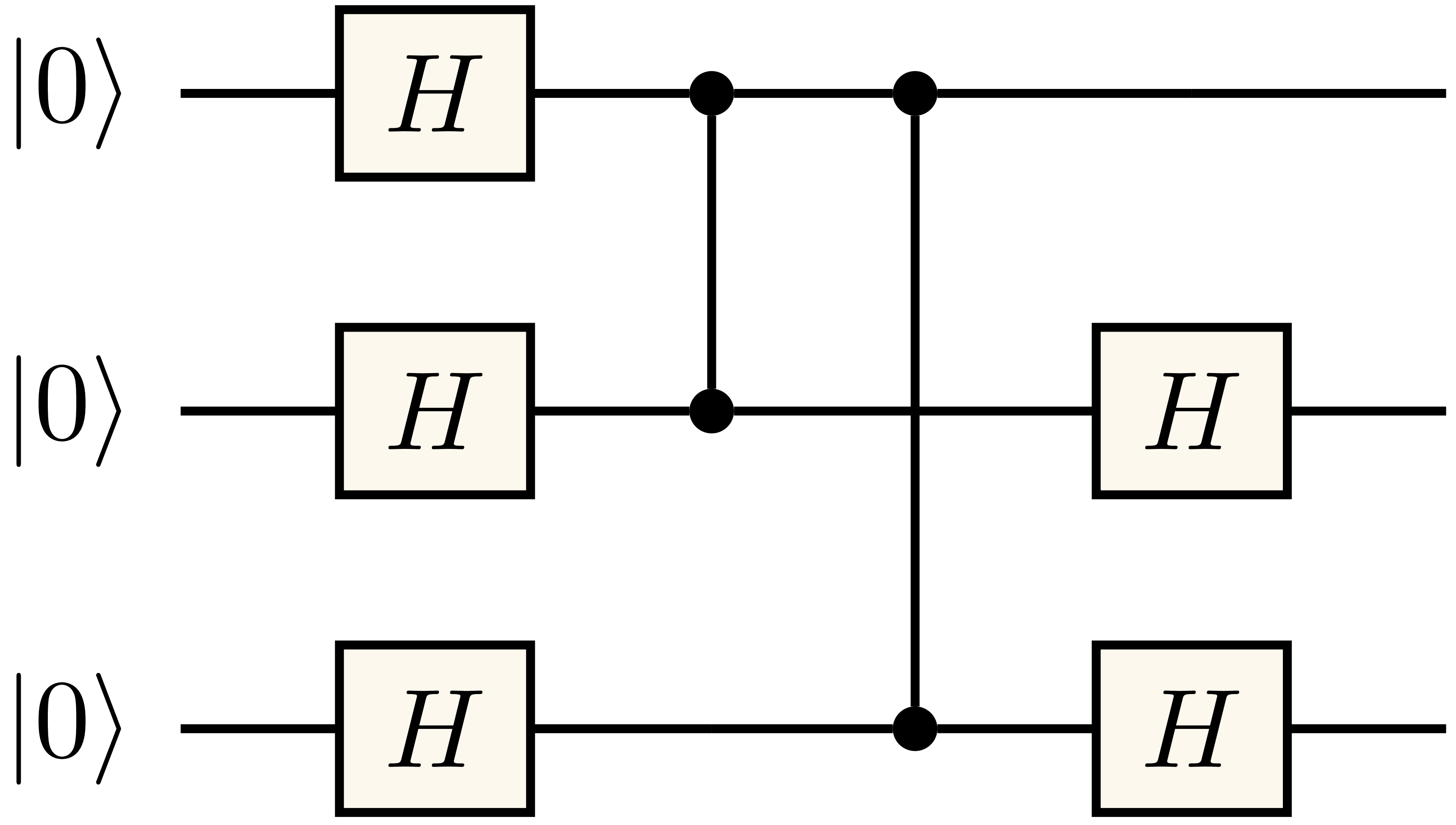
- Represent data: quantum bits
- Initialize the computation: state preparation
- Carry out the computation: quantum gates

Components of a quantum circuit



- Represent data: quantum bits
- Initialize the computation: state preparation
- Carry out the computation: quantum gates
- Read out the result: measurement

Components of a quantum circuit



- Represent data: quantum bits
- Initialize the computation: state preparation
- Carry out the computation: quantum gates
- Read out the result: measurement

Quantum bits

A quantum bit (qubit) can be in a superposition of states

Quantum bits

A quantum bit (qubit) can be in a superposition of states

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha\begin{pmatrix}1\\0\end{pmatrix} + \beta\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\alpha\\\beta\end{pmatrix}$$

$$\alpha, \beta \in \mathbb{C} \qquad |\alpha|^2 + |\beta|^2 = 1$$

Quantum bits

A quantum bit (qubit) can be in a superposition of states

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha\begin{pmatrix}1\\0\end{pmatrix} + \beta\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\alpha\\\beta\end{pmatrix}$$

$$\alpha, \beta \in \mathbb{C} \qquad |\alpha|^2 + |\beta|^2 = 1$$

Can be visualised on the Bloch sphere

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$$

Quantum bits

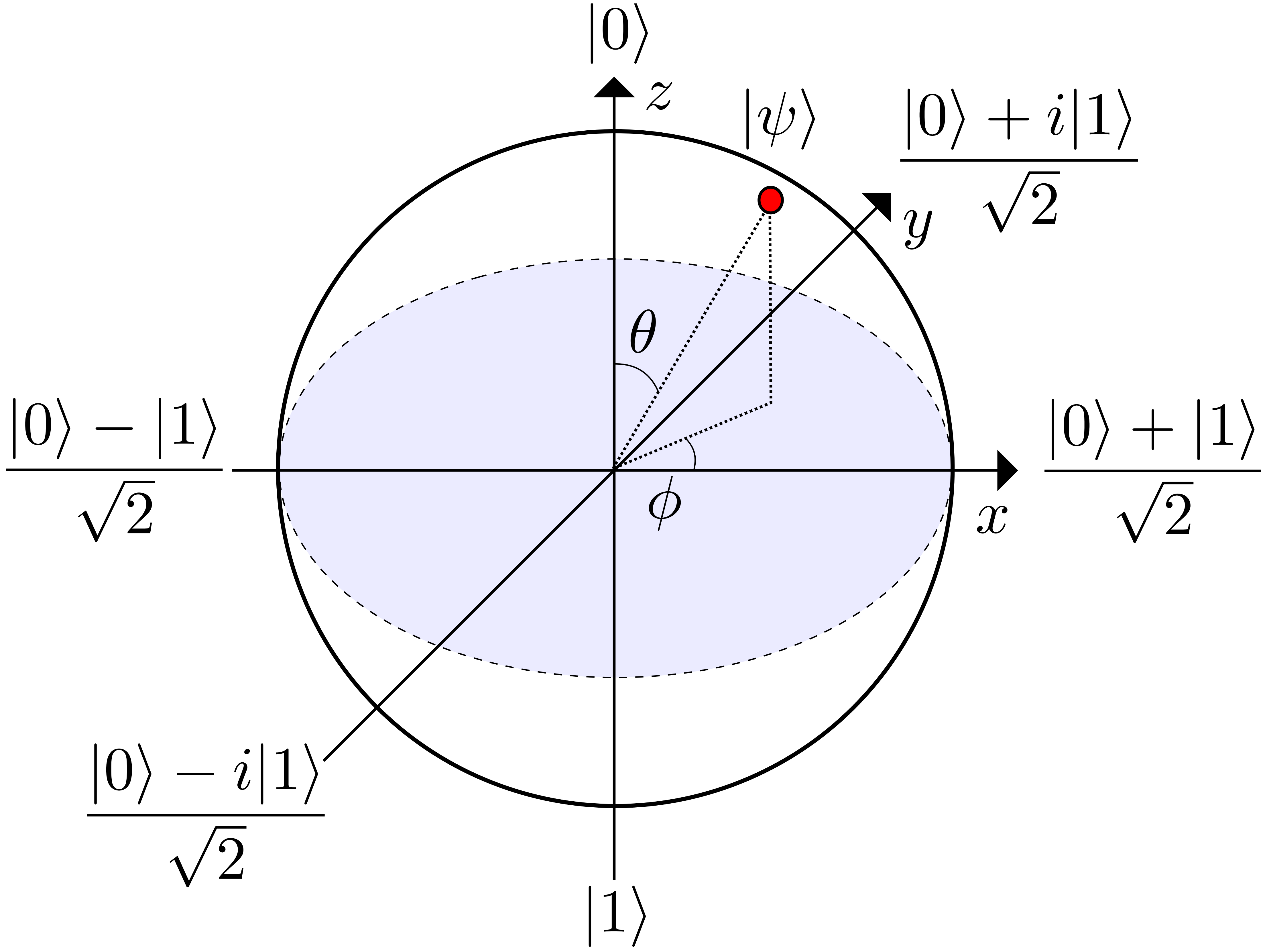
A quantum bit (qubit) can be in a superposition of states

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha\begin{pmatrix}1\\0\end{pmatrix} + \beta\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\alpha\\\beta\end{pmatrix}$$

$$\alpha, \beta \in \mathbb{C} \qquad |\alpha|^2 + |\beta|^2 = 1$$

Can be visualised on the Bloch sphere

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$$



Quantum bits

A quantum bit (qubit) can be in a superposition of states

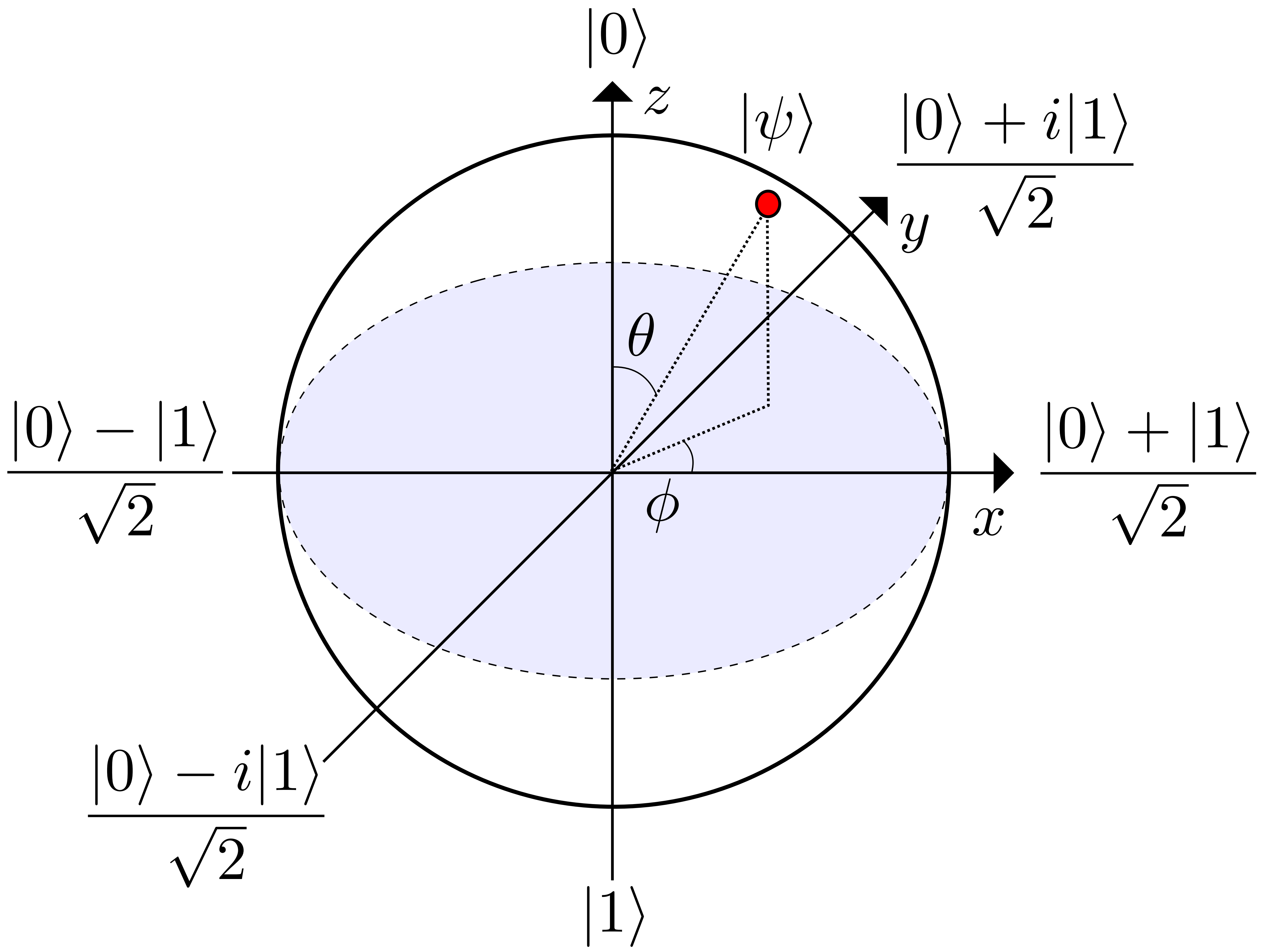
$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha\begin{pmatrix}1\\0\end{pmatrix} + \beta\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}\alpha\\\beta\end{pmatrix}$$

$$\alpha, \beta \in \mathbb{C} \qquad |\alpha|^2 + |\beta|^2 = 1$$

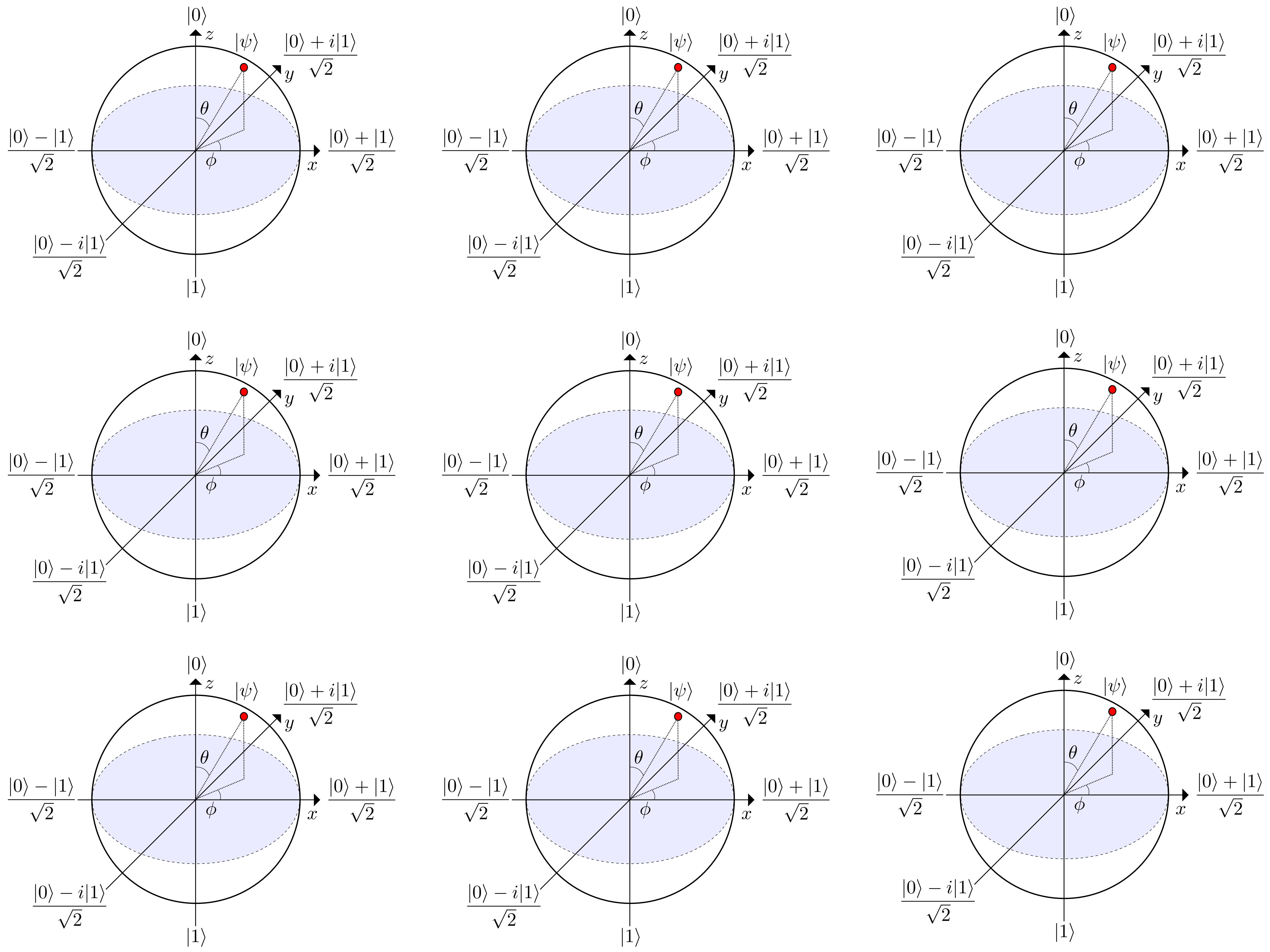
Can be visualised on the Bloch sphere

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle$$

Measurements give either 0 or 1 with probabilities $|\alpha|^2$ and $|\beta|^2$



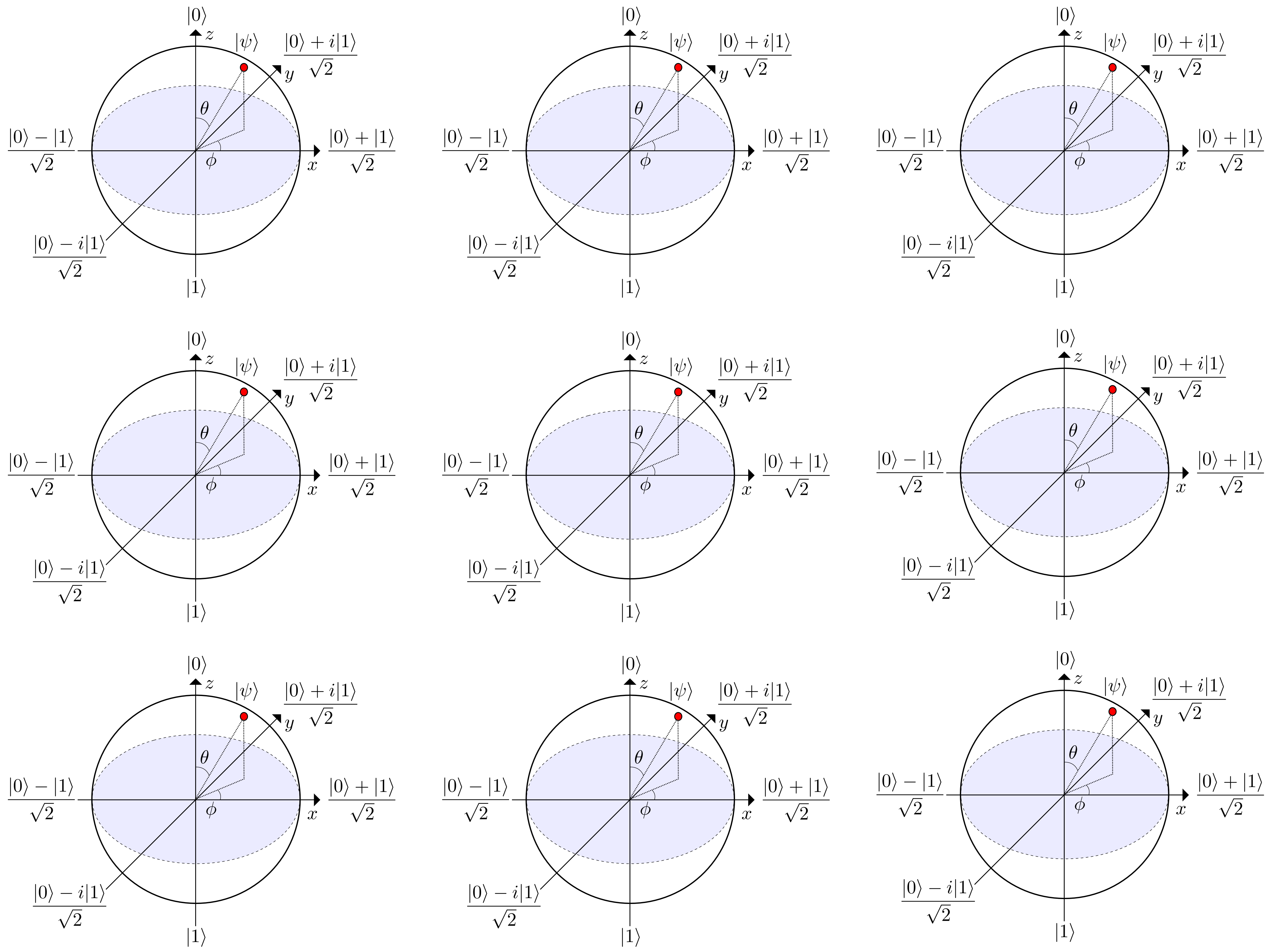
Multi-qubit states



Multi-qubit states

N qubits can be in a superposition of 2^N states

$$|000\ldots00\rangle, |100\ldots00\rangle, |010\ldots00\rangle, \ldots, |111\ldots10\rangle, |111\ldots11\rangle$$

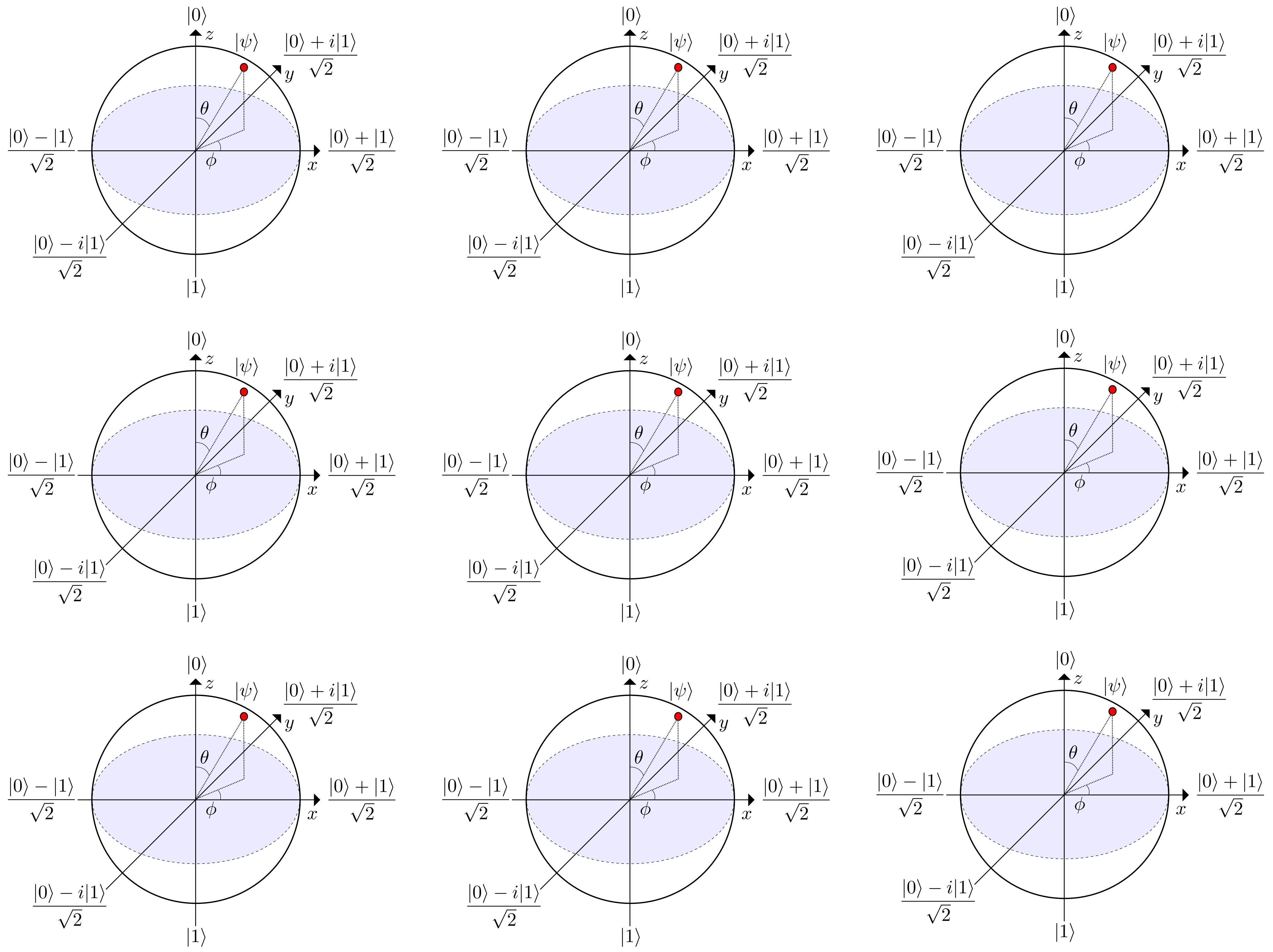


Multi-qubit states

N qubits can be in a superposition of 2^N states

$$|000\ldots00\rangle, |100\ldots00\rangle, |010\ldots00\rangle, \ldots, |111\ldots10\rangle, |111\ldots11\rangle$$

N classical bits can only be in one state at a time



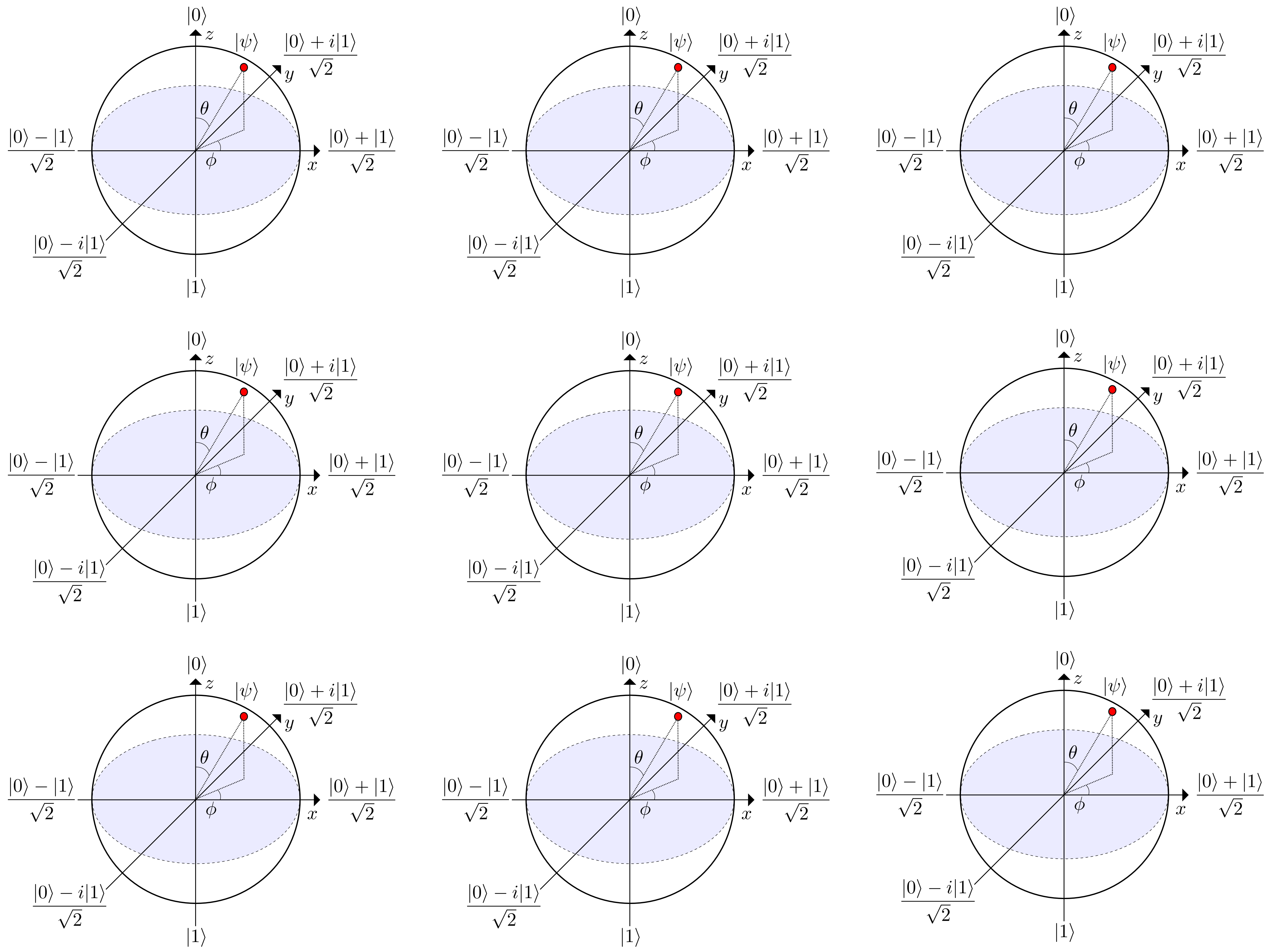
Multi-qubit states

N qubits can be in a superposition of 2^N states

$$|000\ldots00\rangle, |100\ldots00\rangle, |010\ldots00\rangle, \ldots, |111\ldots10\rangle, |111\ldots11\rangle$$

N classical bits can only be in one state at a time

Storing all the information about a quantum state can require $\gg N$ classical bits



Single-qubit gates

Single-qubit gates

Operations changing the state of a qubit must preserve the norm

$$U|\psi\rangle = U(\alpha|0\rangle + \beta|1\rangle) = |\psi'\rangle = \alpha'|0\rangle + \beta'|1\rangle$$

$$|\alpha|^2 + |\beta|^2 = 1 = |\alpha'|^2 + |\beta'|^2$$

Single-qubit gates

Operations changing the state of a qubit must preserve the norm

$$U|\psi\rangle = U(\alpha|0\rangle + \beta|1\rangle) = |\psi'\rangle = \alpha'|0\rangle + \beta'|1\rangle$$

$$|\alpha|^2 + |\beta|^2 = 1 = |\alpha'|^2 + |\beta'|^2$$

They are 2x2 unitary matrices, e.g., the Pauli matrices

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Single-qubit gates

Operations changing the state of a qubit must preserve the norm

$$U|\psi\rangle = U(\alpha|0\rangle + \beta|1\rangle) = |\psi'\rangle = \alpha'|0\rangle + \beta'|1\rangle$$

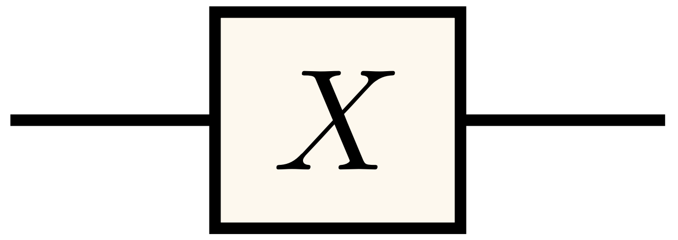
$$|\alpha|^2 + |\beta|^2 = 1 = |\alpha'|^2 + |\beta'|^2$$

They are 2x2 unitary matrices, e.g., the Pauli matrices

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$



is the quantum NOT gate

$$X(\alpha|0\rangle + \beta|1\rangle) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix} = \beta|0\rangle + \alpha|1\rangle$$

Single-qubit gates

Operations changing the state of a qubit must preserve the norm

$$U|\psi\rangle = U(\alpha|0\rangle + \beta|1\rangle) = |\psi'\rangle = \alpha'|0\rangle + \beta'|1\rangle$$

$$|\alpha|^2 + |\beta|^2 = 1 = |\alpha'|^2 + |\beta'|^2$$

$-\boxed{X}-$ is the quantum NOT gate

$$X(\alpha|0\rangle + \beta|1\rangle) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \beta \\ \alpha \end{pmatrix} = \beta|0\rangle + \alpha|1\rangle$$

They are 2x2 unitary matrices, e.g., the Pauli matrices

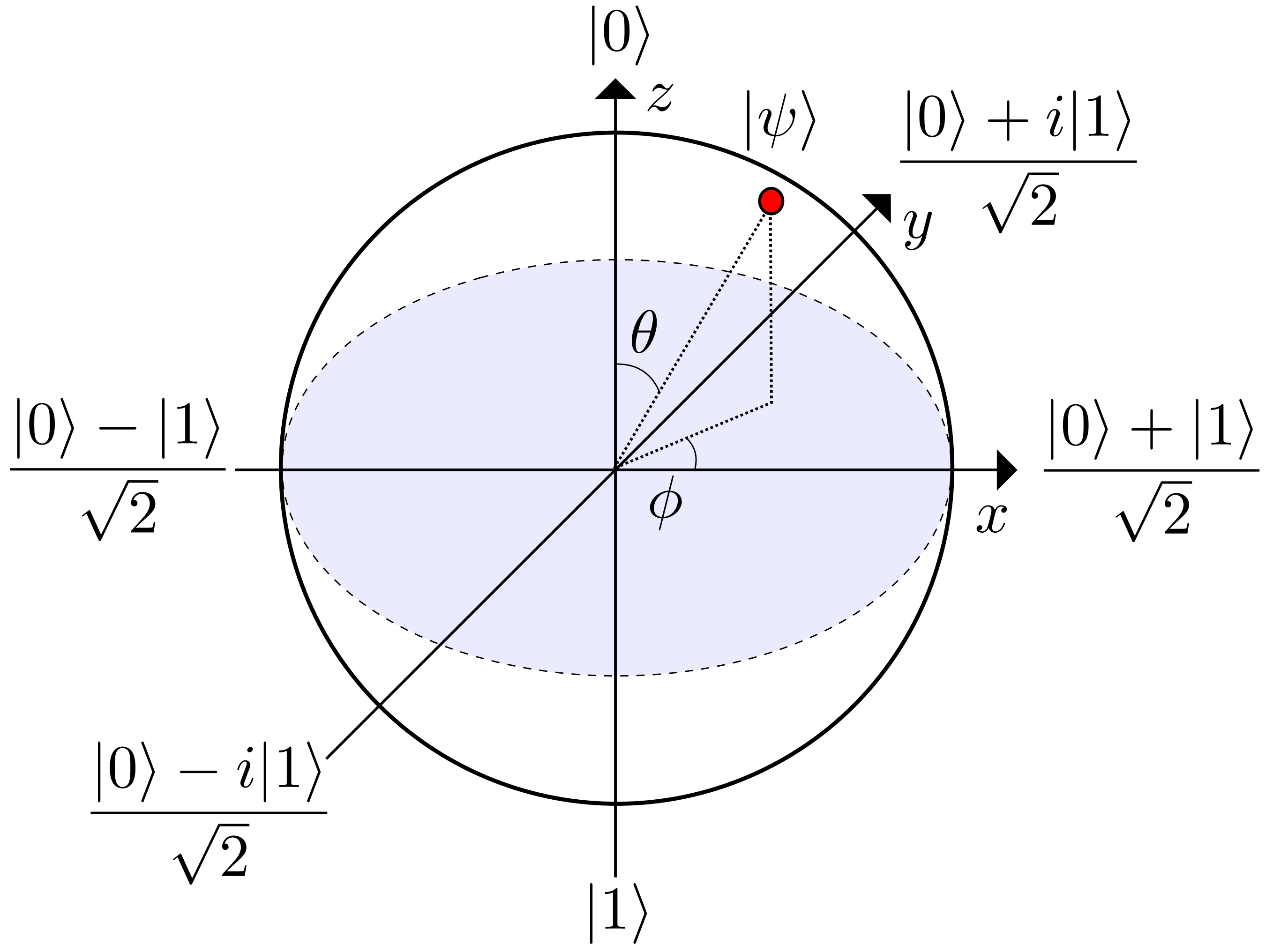
$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},$$

$$Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

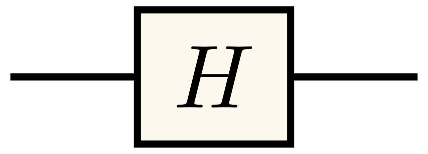
Rotations around different axes of the Bloch sphere

$$\begin{aligned} R_x(\theta) &= \exp(-i\theta X/2) \\ &= \cos(\theta/2)I - i\sin(\theta/2)X \\ &= \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix} \end{aligned}$$



More single-qubit gates

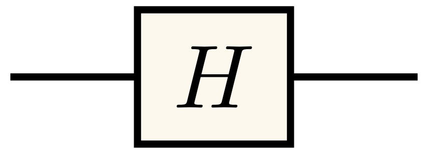
More single-qubit gates



The Hadamard gate

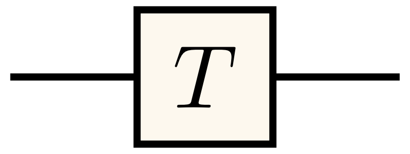
$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{X + Z}{\sqrt{2}}$$

More single-qubit gates



The Hadamard gate

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{X + Z}{\sqrt{2}}$$



The T gate ($\pi/8$ gate)

$$T = \begin{pmatrix} 1 & 0 \\ 0 & \exp(i\pi/4) \end{pmatrix} = \exp(i\pi/8)\begin{pmatrix} \exp(-i\pi/8) & 0 \\ 0 & \exp(i\pi/8) \end{pmatrix}$$

More single-qubit gates

$$\text{—}\boxed{H}\text{—}$$

The Hadamard gate $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{X+Z}{\sqrt{2}}$

$$\text{—}\boxed{T}\text{—}$$

The T gate ($\pi/8$ gate) $T = \begin{pmatrix} 1 & 0 \\ 0 & \exp(i\pi/4) \end{pmatrix} = \exp(i\pi/8)\begin{pmatrix} \exp(-i\pi/8) & 0 \\ 0 & \exp(i\pi/8) \end{pmatrix}$

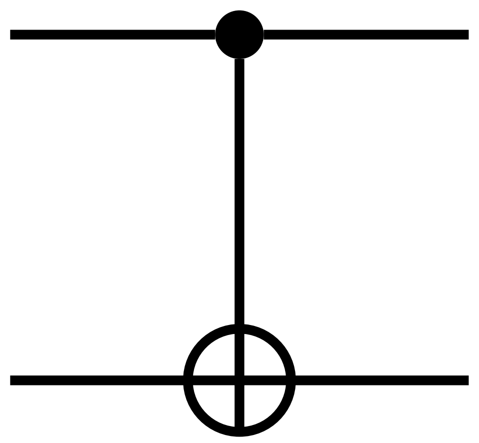
$$\text{—}\boxed{S}\text{—}$$

The phase (or S, or P) gate $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} = T^2$

Two-qubit gates

Two-qubit gates

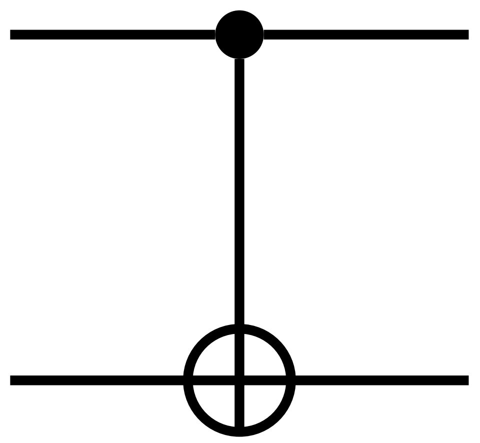
Controlled-NOT $\mathrm{CNOT} = \begin{array}{c c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} & \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} & \begin{array}{c} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array} \end{array}$



Two-qubit gates

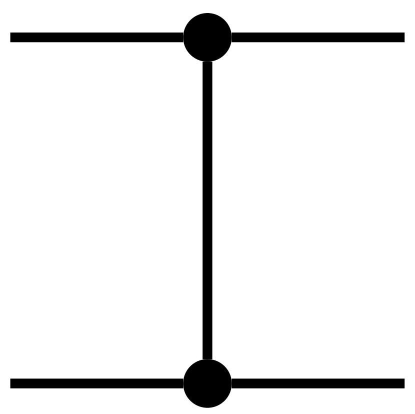
Controlled-NOT

$$\mathrm{CNOT} = \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} \begin{array}{c} \\ |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array}$$



Controlled-Z

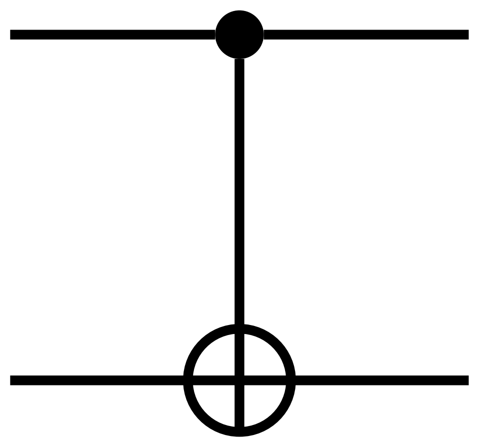
$$\mathrm{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$



Two-qubit gates

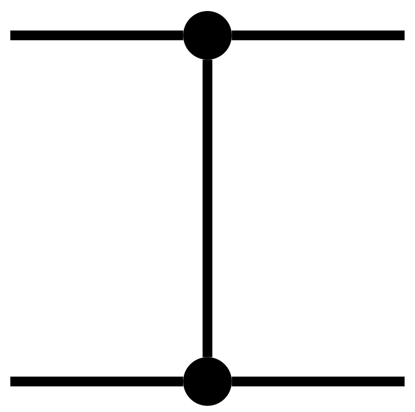
Controlled-NOT

$$\mathrm{CNOT} = \begin{array}{cc} & \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ & \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{array}{c} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array} \end{array}$$



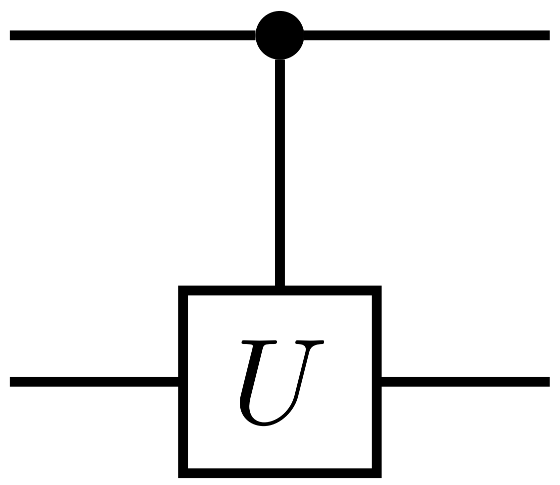
Controlled-Z

$$\mathrm{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$$



Controlled unitary

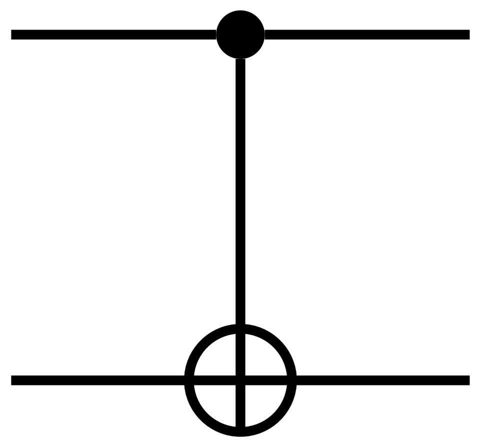
$$\begin{pmatrix} I_2 & 0_2 \\ 0_2 & U \end{pmatrix}$$



Two-qubit gates

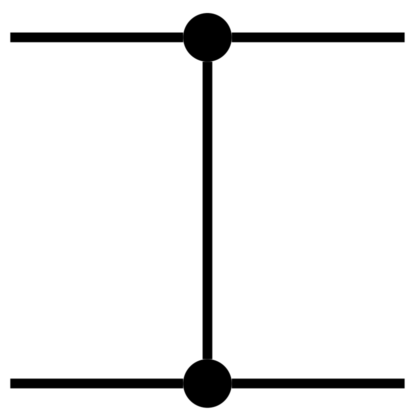
Controlled-NOT

$$
\mathrm{CNOT} = \begin{array}{c}
\begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}
\end{array}
\begin{array}{c} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array}
$$



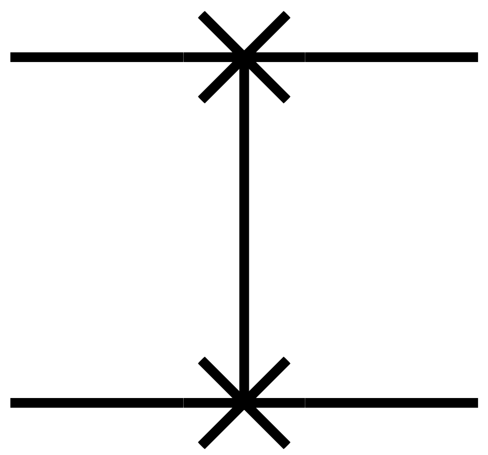
Controlled-Z

$$
\mathrm{CZ} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}
$$



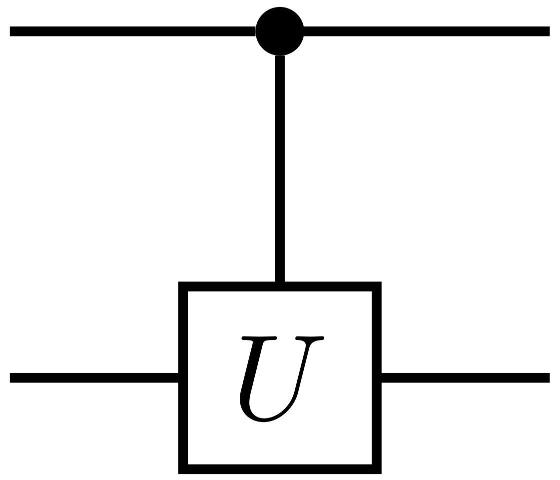
SWAP

$$
\mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}
$$



Controlled unitary

$$
\begin{pmatrix} I_2 & 0_2 \\ 0_2 & U \end{pmatrix}
$$

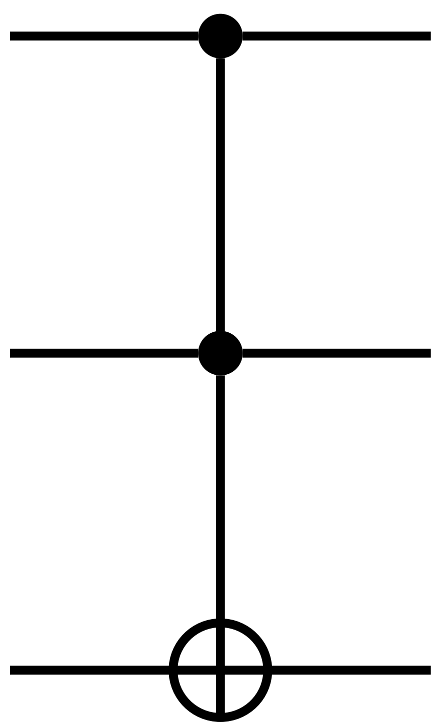


Three-qubit gates

Three-qubit gates

Controlled-controlled-NOT

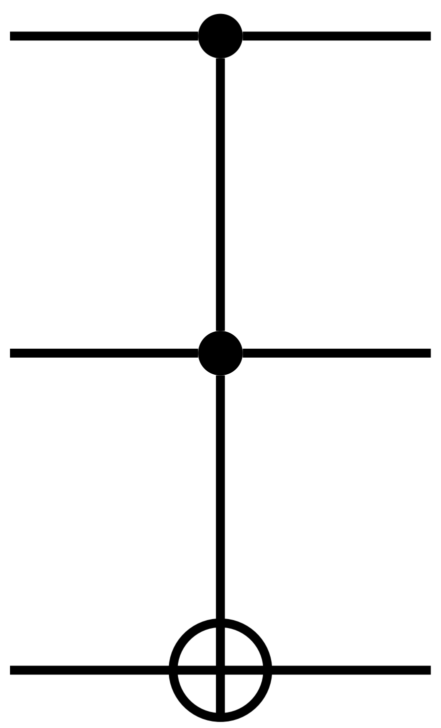
$$\text{Toffoli} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Three-qubit gates

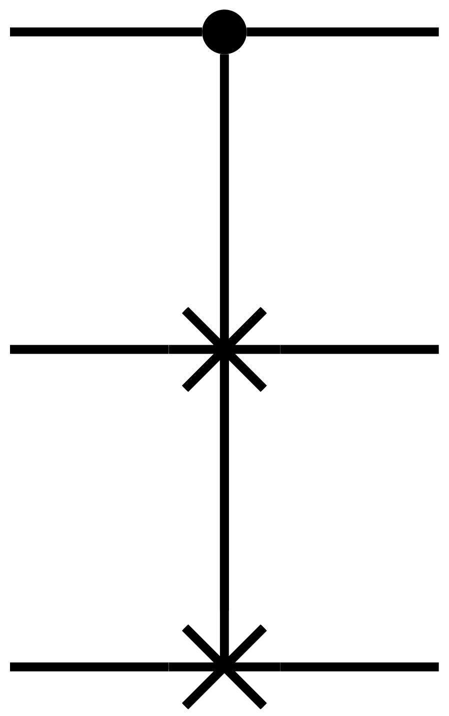
Controlled-controlled-NOT

$$\text{Toffoli} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Controlled-SWAP

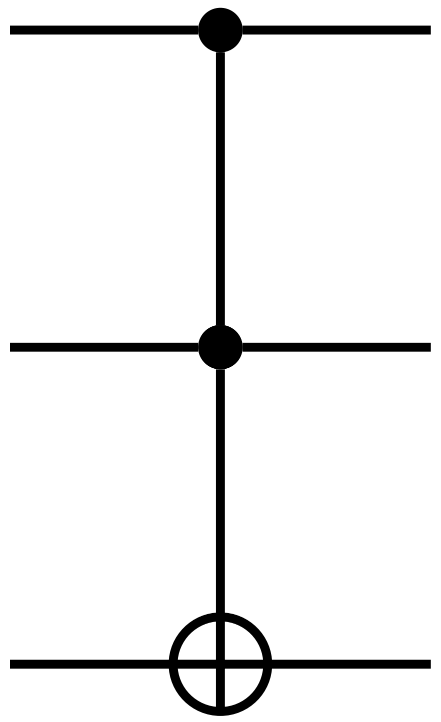
$$\text{Fredkin} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$



Three-qubit gates

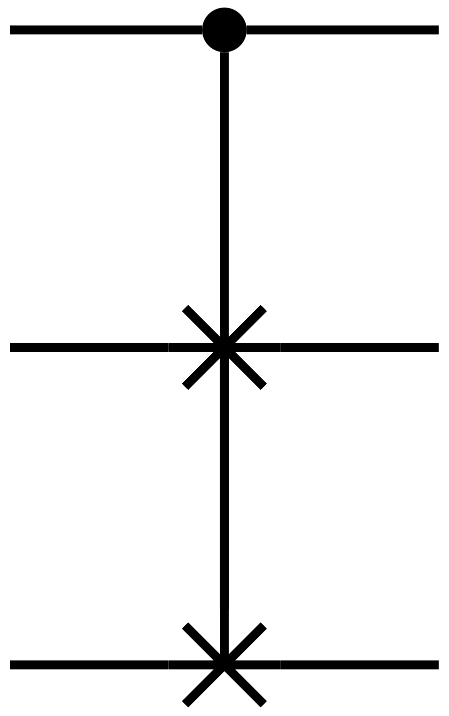
Controlled-controlled-NOT

$$\text{Toffoli} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$



Controlled-SWAP

$$\text{Fredkin} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$



Multi-qubit gates can be decomposed into sequences of single- and two-qubit gates

Universal gate sets

Universal gate sets

<u>Classical computing</u>

A gate set is universal if it enables expressing any Boolean function on any number of bits

Universal gate sets

Classical computing

A gate set is universal if it enables expressing any Boolean function on any number of bits

The NAND gate is universal

{AND, OR} is not a universal gate set

Universal gate sets

Classical computing

A gate set is universal if it enables expressing any Boolean function on any number of bits

Quantum computing

A gate set is universal if the gates therein can approximate any unitary transformation on any number of qubits to any precision

The NAND gate is universal

{AND, OR} is not a universal gate set

Universal gate sets

Classical computing

A gate set is universal if it enables expressing any Boolean function on any number of bits

The NAND gate is universal

{AND, OR} is not a universal gate set

Quantum computing

A gate set is universal if the gates therein can approximate any unitary transformation on any number of qubits to any precision

What are requirements for a universal set of quantum gates?

Universal quantum gate sets

Failure modes

Universal quantum gate sets

Failure modes

- Inability to create superposition states
  {X, CNOT}

Universal quantum gate sets

Failure modes

- Inability to create superposition states
  {X, CNOT}
- Inability to create entanglement
  {H, S}

Universal quantum gate sets

Failure modes

- Inability to create superposition states
  {X, CNOT}
- Inability to create entanglement
  {H, S}
- Inability to create complex amplitudes
  {H, CNOT}

Universal quantum gate sets

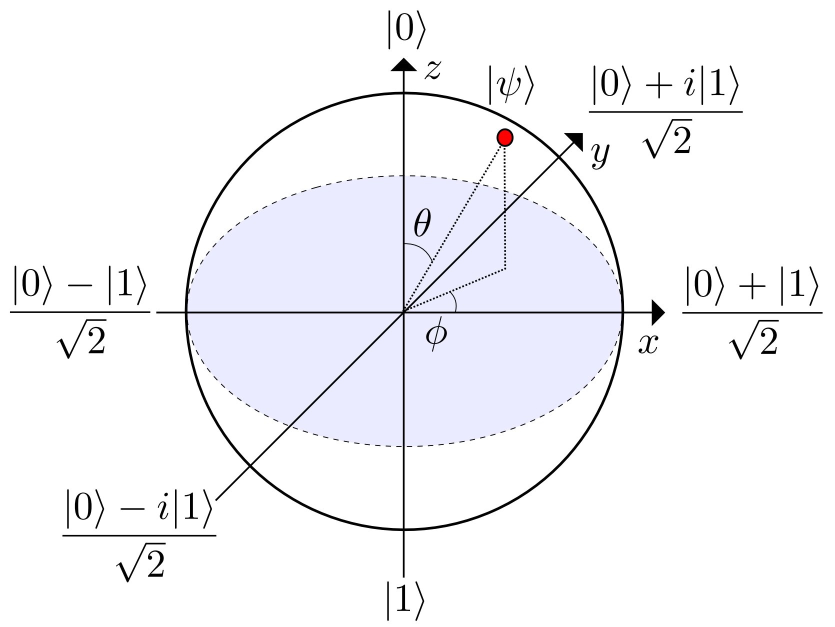
Failure modes

- Inability to create superposition states
  {X, CNOT}
- Inability to create entanglement
  {H, S}
- Inability to create complex amplitudes
  {H, CNOT}
- The Gottesman-Knill theorem
  {H, CNOT, S} still not enough!

Universal quantum gate sets

<u>Failure modes</u>

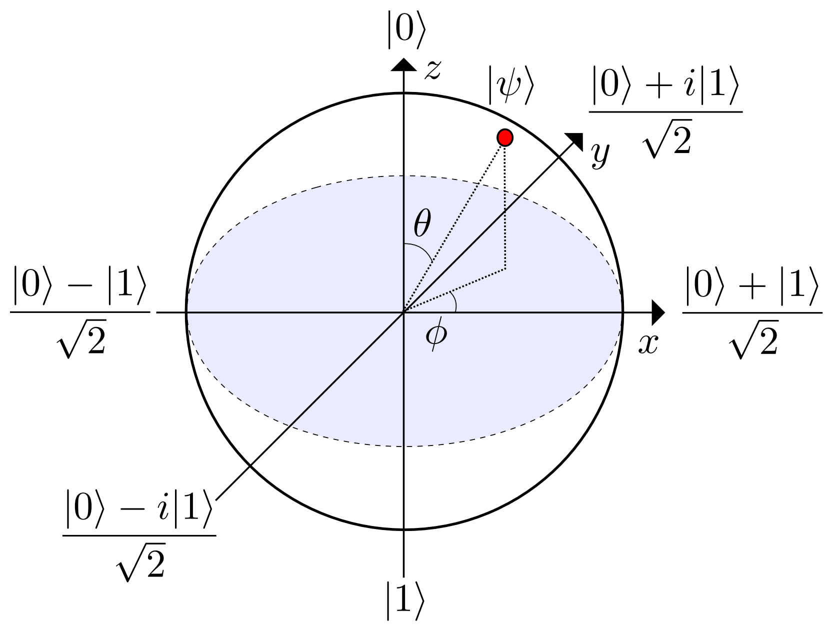
- Inability to create superposition states
  {X, CNOT}
- Inability to create entanglement
  {H, S}
- Inability to create complex amplitudes
  {H, CNOT}
- The Gottesman-Knill theorem
  {H, CNOT, S} still not enough!



Universal quantum gate sets

Failure modes

- Inability to create superposition states
  {X, CNOT}
- Inability to create entanglement
  {H, S}
- Inability to create complex amplitudes
  {H, CNOT}
- The Gottesman-Knill theorem
  {H, CNOT, S} still not enough!



Universal gate sets

- Almost anything else than
  H in {H, CNOT, S}
- Almost any two-qubit gate
  on its own
- In practice: many single-qubit gates
  + one or two two-qubit gates

The Solovay-Kitaev theorem

Are there problems that a classical computer can’t solve but a quantum computer can?

The Solovay-Kitaev theorem

Are there problems that a classical computer can't solve but a quantum computer can?

The Solovay-Kitaev theorem

Are there problems that a classical computer can't solve but a quantum computer can?

The classical computer, given enough resources, can store all the complex amplitudes specifying the state of the quantum computer and simulate all the gates in a quantum circuit

The Solovay-Kitaev theorem

Are there problems that a classical computer can't solve but a quantum computer can?

The classical computer, given enough resources, can store all the complex amplitudes specifying the state of the quantum computer and simulate all the gates in a quantum circuit

For the quantum computer to be faster, one thing to worry about is whether the universal gate set can represent the desired algorithm with enough precision without requiring too long circuits

The Solovay-Kitaev theorem

The Solovay-Kitaev theorem

Let G be a finite subset of $SU(2)$ and $U \in SU(2)$. If the group generated by G is dense in $SU(2)$, then for any $\varepsilon > 0$ it is possible to approximate U to precision ε using $\mathcal{O}\left(\log^4\left[\frac{1}{\varepsilon}\right]\right)$ gates from G.

The Solovay-Kitaev theorem

Let G be a finite subset of $SU(2)$ and $U \in SU(2)$. If the group generated by G is dense in $SU(2)$, then for any $\varepsilon > 0$ it is possible to approximate U to precision ε using $\mathcal{O}\left(\log^4\left[\frac{1}{\varepsilon}\right]\right)$ gates from G.

For an N-qubit unitary, at most $\mathcal{O}\left(4^N \log^4\left[\frac{1}{\varepsilon}\right]\right)$ gates suffice

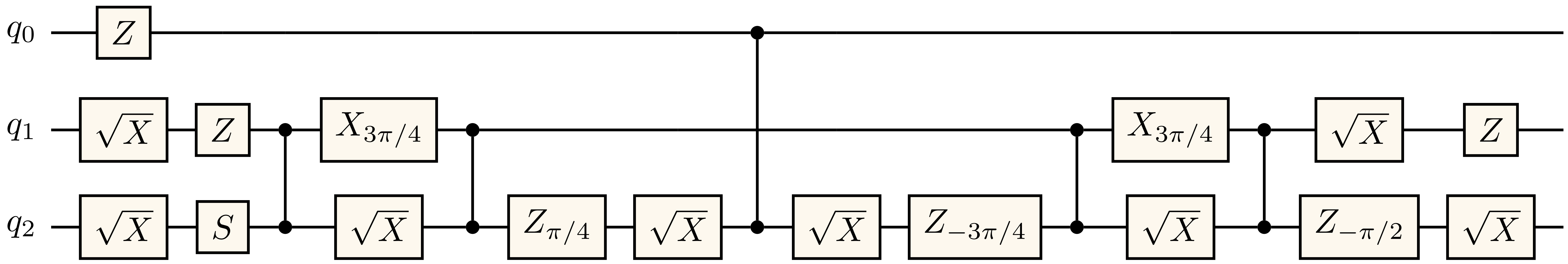
The Solovay-Kitaev theorem

Let G be a finite subset of $SU(2)$ and $U \in SU(2)$. If the group generated by G is dense in $SU(2)$, then for any $\varepsilon > 0$ it is possible to approximate U to precision ε using $\mathcal{O}\left(\log^4\left[\frac{1}{\varepsilon}\right]\right)$ gates from G.

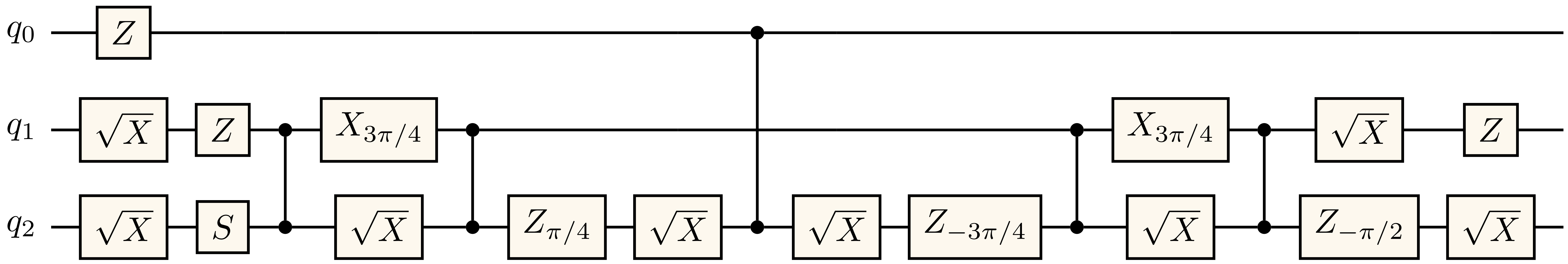
For an N-qubit unitary, at most $\mathcal{O}\left(4^N \log^4\left[\frac{1}{\varepsilon}\right]\right)$ gates suffice

Precision is thus not a problem in practice for available universal gate sets

Quantum algorithms and compilation

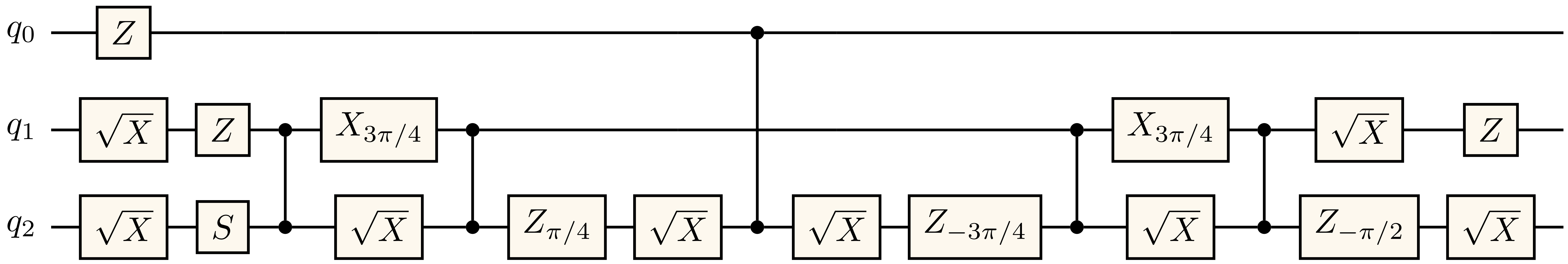


Quantum algorithms and compilation



Quantum algorithms are sequences of gates acting on quantum states

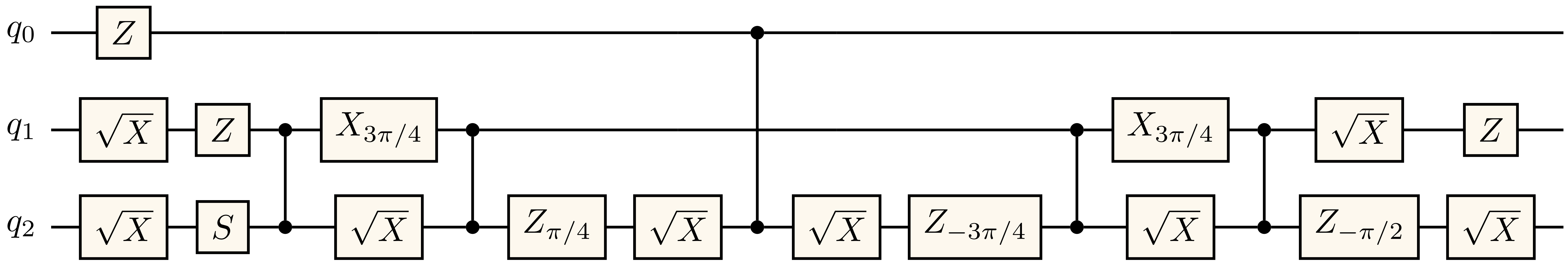
Quantum algorithms and compilation



Quantum algorithms are sequences of gates acting on quantum states

<u>Compilation steps</u>

Quantum algorithms and compilation

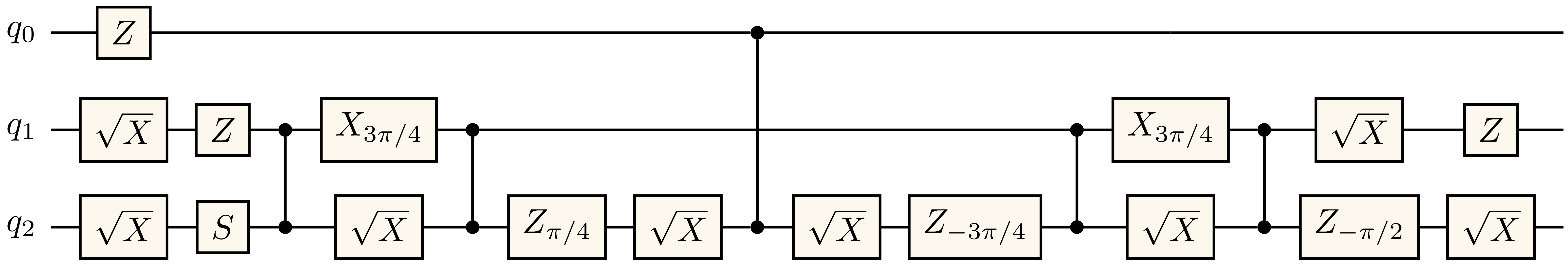


Quantum algorithms are sequences of gates acting on quantum states

<u>Compilation steps</u>

- Convert the gates of the algorithm into gates in your native universal gate set

Quantum algorithms and compilation

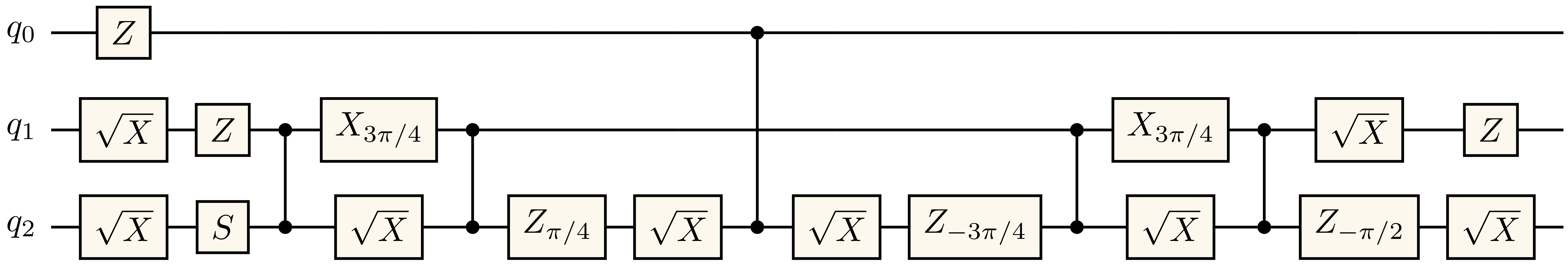


Quantum algorithms are sequences of gates acting on quantum states

<u>Compilation steps</u>

- Convert the gates of the algorithm into gates in your native universal gate set
- Map qubits in the algorithm to qubits on your hardware

Quantum algorithms and compilation

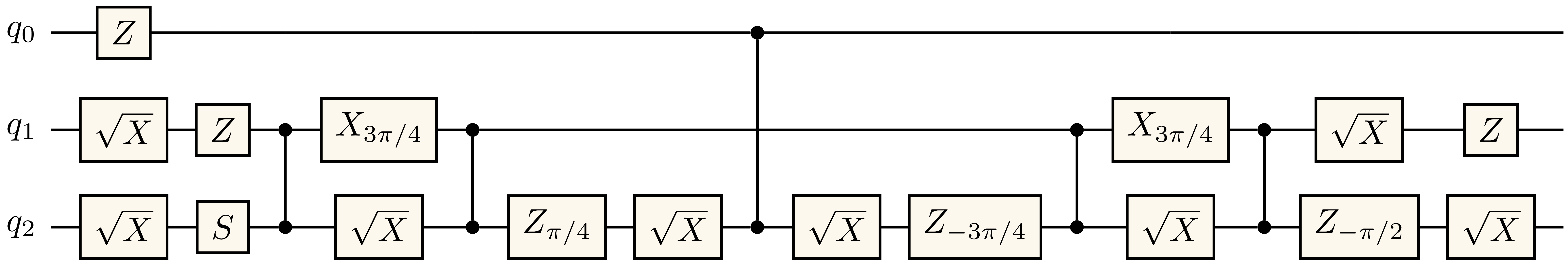


Quantum algorithms are sequences of gates acting on quantum states

<u>Compilation steps</u>

- Convert the gates of the algorithm into gates in your native universal gate set
- Map qubits in the algorithm to qubits on your hardware
- Insert SWAP gates to connect qubits far apart that need to interact

Quantum algorithms and compilation



Quantum algorithms are sequences of gates acting on quantum states

<u>Compilation steps</u>

- Convert the gates of the algorithm into gates in your native universal gate set
- Map qubits in the algorithm to qubits on your hardware
- Insert SWAP gates to connect qubits far apart that need to interact
- Compress the resulting circuit

Summary

- Qubits can be in superposition states; exponentially many classical bits are required to describe many qubits
- Quantum algorithms are implemented by applying a sequence of single- and two-qubit gates (unitary matrices) to the qubits (states represented as vectors)
- Quantum algorithms need to be compiled to fit on the quantum hardware; the Solovay-Kitaev theorem tells us that universal gate sets can achieve this without prohibitive overhead to ensure precision