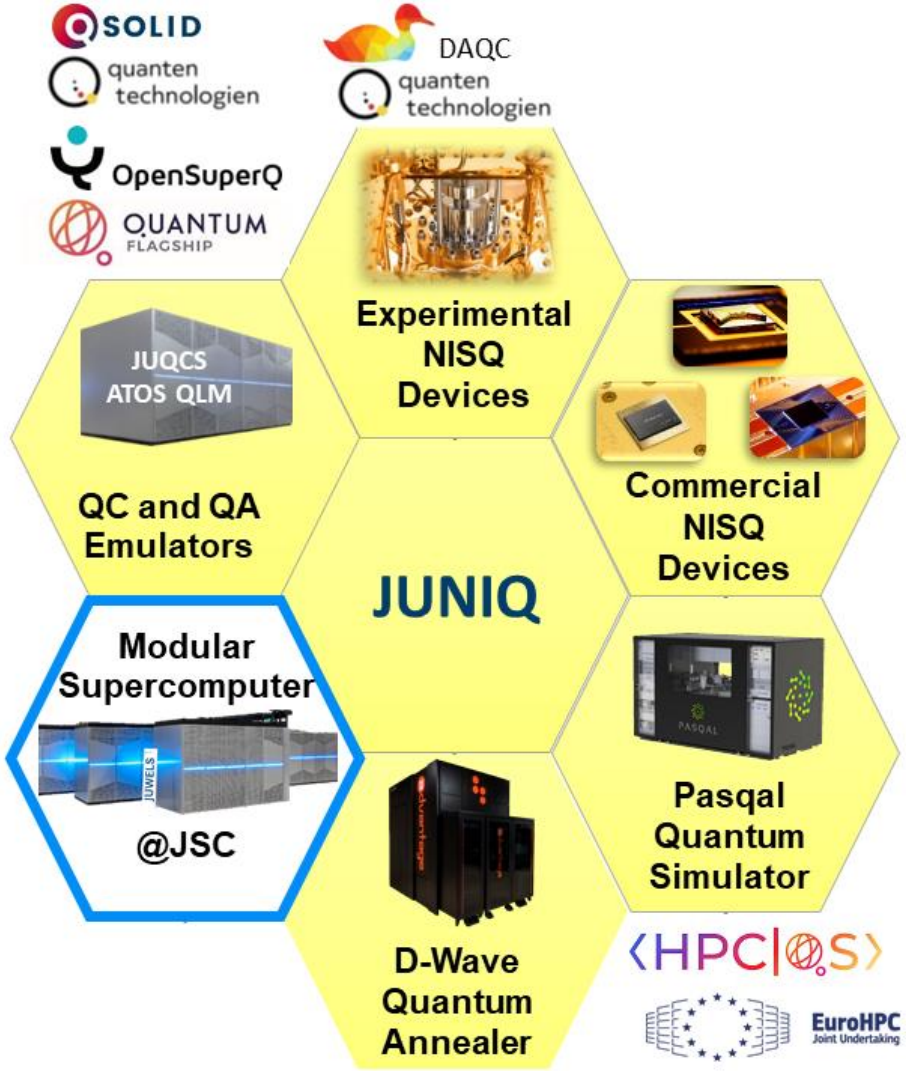
INTRODUCTION TO GATE-BASED QUANTUM COMPUTING

JUNIQ Summer School on Quantum Information Processing 2023

28 AUGUST 2023 I DR. DENNIS WILLSCH

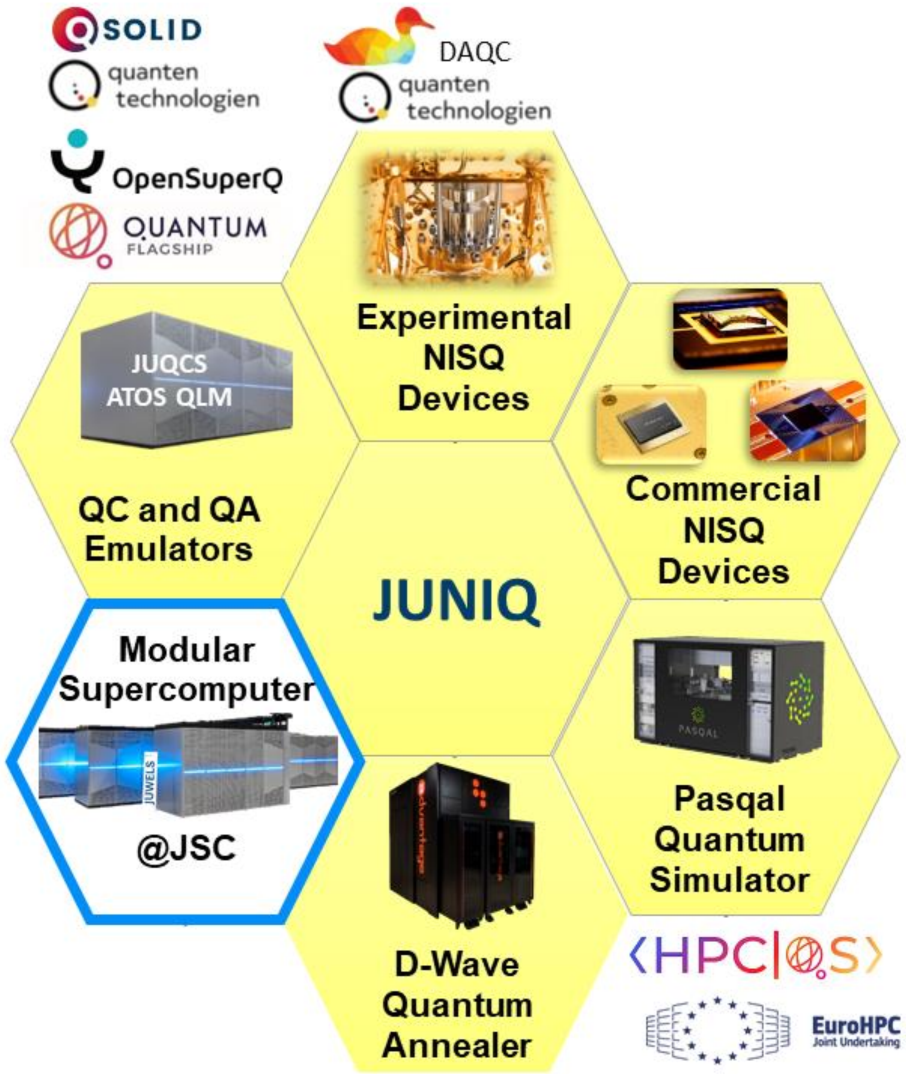
CONTENTS

Lecture Notes:
Programming Quantum Computers
https://arxiv.org/pdf/2201.02051.pdf



CONTENTS

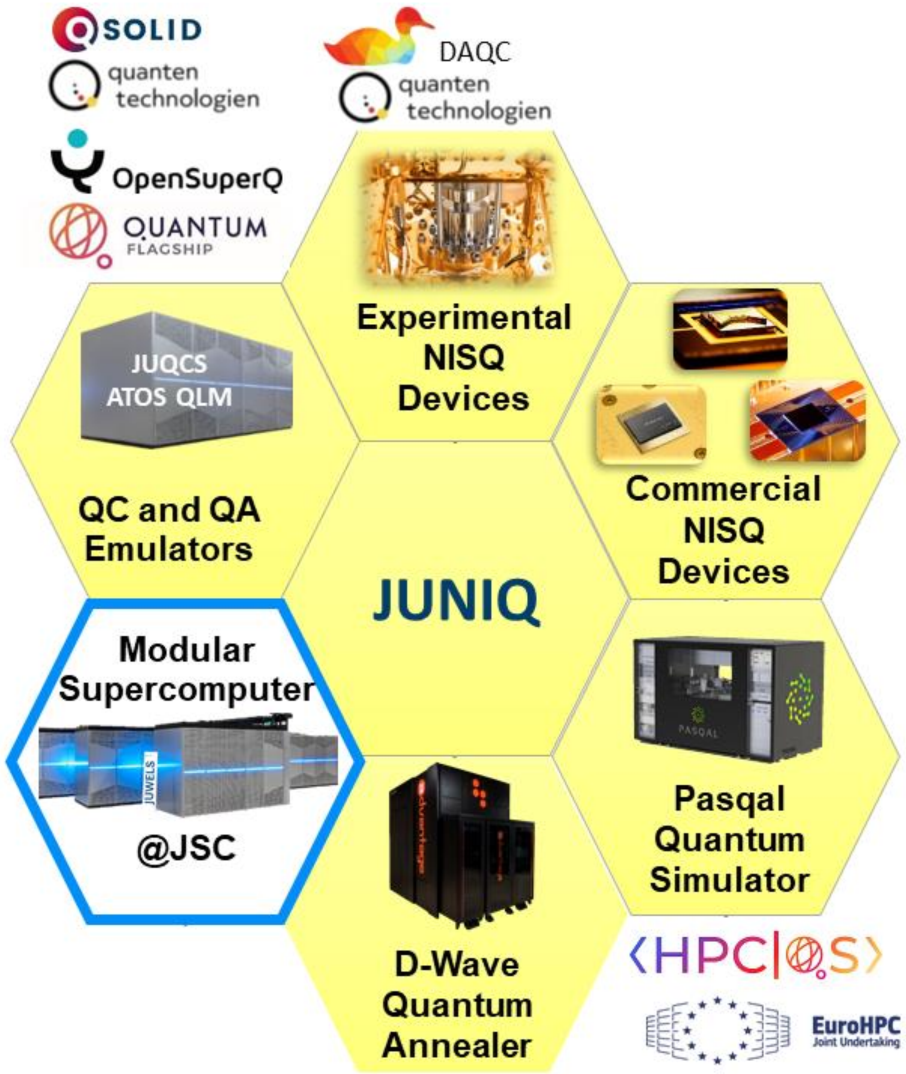
1. **Quantum Bits and Quantum Gates**



JÜLICH Forschungszentrum

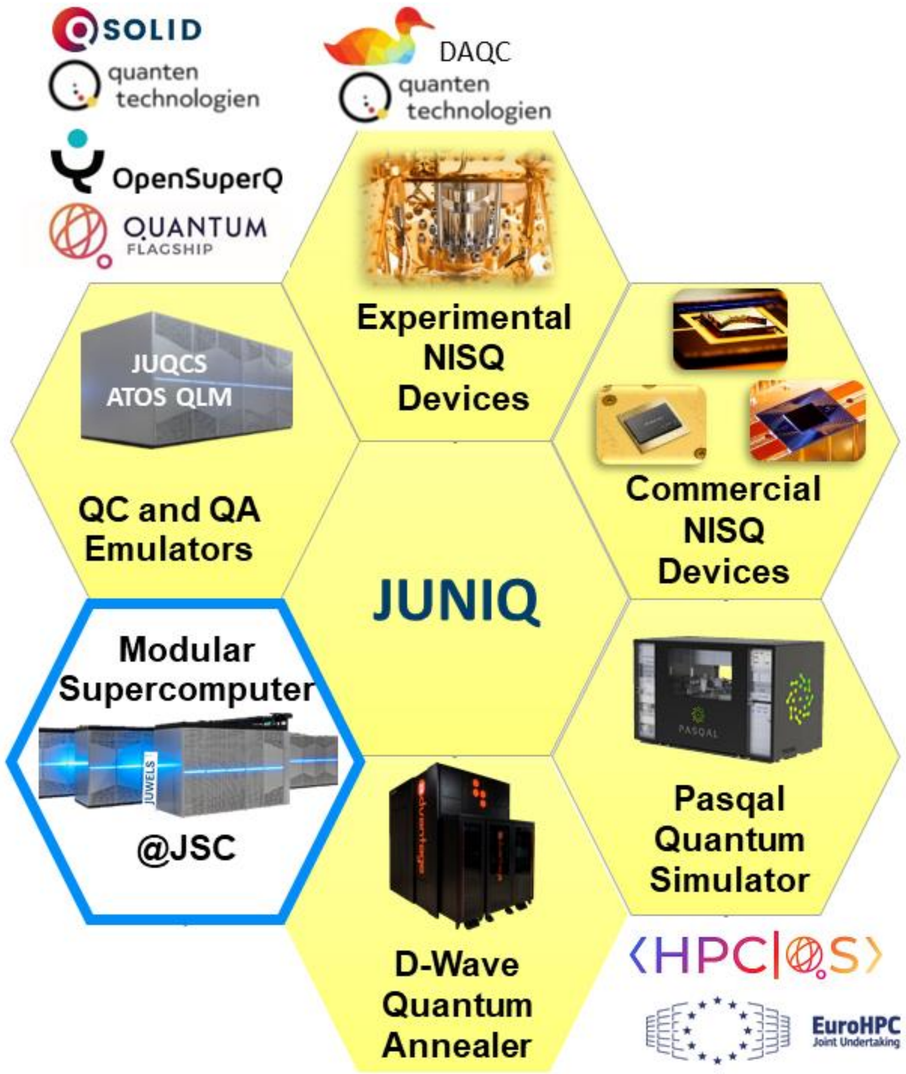
CONTENTS

1. **Quantum Bits and Quantum Gates**
2. **Programming and Simulating Quantum Circuits**



JÜLICH Forschungszentrum

CONTENTS


QSOLID
quanten technologien
OpenSuperQ
QUANTUM FLAGSHIP
DAQC
quanten technologien
Experimental NISQ Devices
JUQCS ATOS QLM
QC and QA Emulators
Commercial NISQ Devices
JUNIQ
Modular Supercomputer
@JSC
Pasqal Quantum Simulator
D-Wave Quantum Annealer
HPCQS
EuroHPC Joint Undertaking

QUANTUM BITS AND QUANTUM GATES

A single qubit

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Computational basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

A single qubit

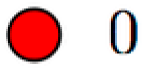
- Definition of a single qubit

$$|\psi\rangle = \psi_0 \,|0\rangle + \psi_1 \,|1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Computational basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Digital computer: Bit



QUANTUM BITS AND QUANTUM GATES

A single qubit

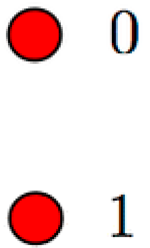
- Definition of a single qubit

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Computational basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Digital computer: Bit



QUANTUM BITS AND QUANTUM GATES

A single qubit

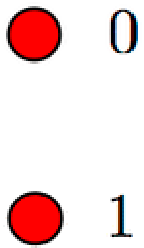
- Definition of a single qubit

$$|\psi\rangle = \psi_0 \, |0\rangle + \psi_1 \, |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Computational basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Digital computer: Bit



Quantum computer: Quantum Bit = Qubit

QUANTUM BITS AND QUANTUM GATES

A single qubit

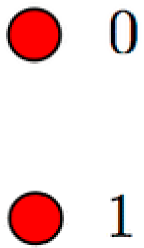
- Definition of a single qubit

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

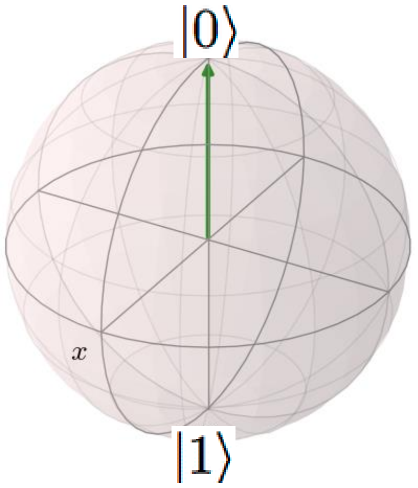
Computational basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Digital computer: Bit



Quantum computer: Quantum Bit = Qubit



QUANTUM BITS AND QUANTUM GATES

A single qubit

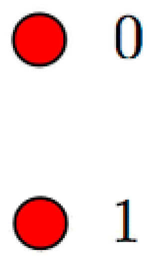
- Definition of a single qubit

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

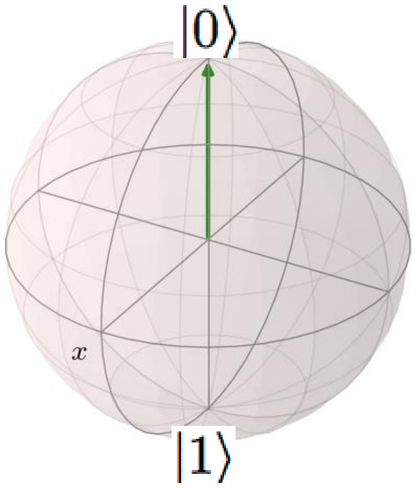
Computational basis states:

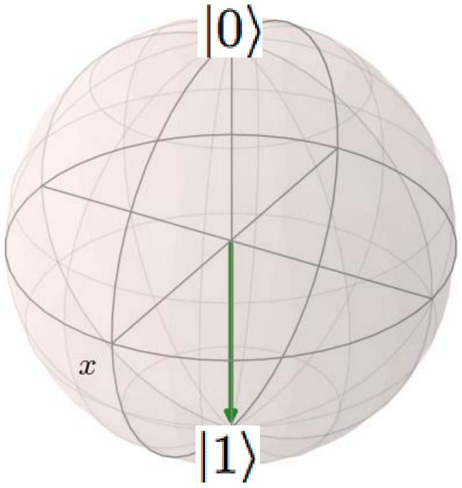
$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Digital computer: Bit



Quantum computer: Quantum Bit = Qubit





QUANTUM BITS AND QUANTUM GATES

A single qubit

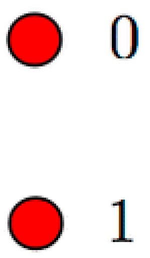
- Definition of a single qubit

$$|\psi\rangle = \psi_0 \, |0\rangle + \psi_1 \, |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

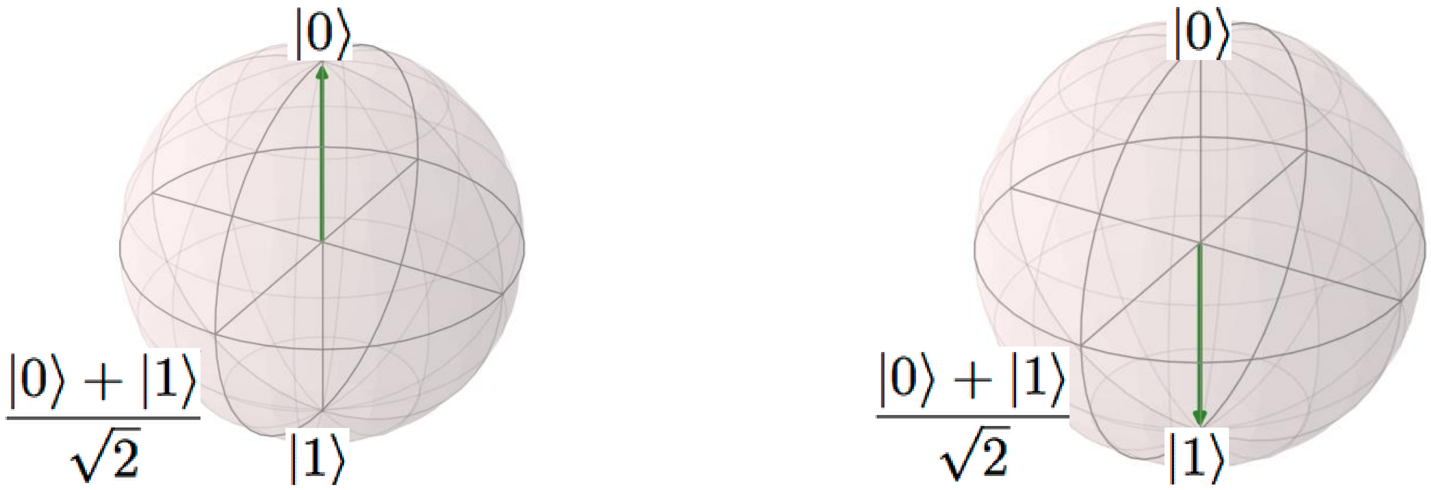
Computational basis states:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

Digital computer: Bit



Quantum computer: Quantum Bit = Qubit



QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0 \, |0\rangle + \psi_1 \, |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Normalization:

$$\langle\psi|\psi\rangle = |\psi_0|^2 + |\psi_1|^2 = 1$$

$$\langle\psi| = (\psi_0^*, \;\; \psi_1^*)$$

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0 |0\rangle + \psi_1 |1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Normalization:

$$\langle\psi|\psi\rangle = |\psi_0|^2 + |\psi_1|^2 = 1$$

$$\langle\psi| = (\psi_0^*, \ \psi_1^*)$$

Global phase:

$$|\psi\rangle \equiv e^{i\Phi}|\psi\rangle$$

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle = \begin{pmatrix}\psi_0\\ \psi_1\end{pmatrix}$$

Normalization:

$$\langle\psi|\psi\rangle = |\psi_0|^2 + |\psi_1|^2 = 1$$
$$\langle\psi| = (\psi_0^*,\ \ \psi_1^*)$$

- Representation on the Bloch sphere

$$|\psi\rangle = \cos\frac{\vartheta}{2}\,|0\rangle + e^{i\varphi}\sin\frac{\vartheta}{2}\,|1\rangle = \begin{pmatrix}\cos\frac{\vartheta}{2}\\ e^{i\varphi}\sin\frac{\vartheta}{2}\end{pmatrix}$$

Global phase:

$$|\psi\rangle \equiv e^{i\Phi}|\psi\rangle$$

JÜLICH Forschungszentrum

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle = \begin{pmatrix}\psi_0\\ \psi_1\end{pmatrix}$$

- Representation on the Bloch sphere

$$|\psi\rangle = \cos\frac{\vartheta}{2}\,|0\rangle + e^{i\varphi}\sin\frac{\vartheta}{2}\,|1\rangle = \begin{pmatrix}\cos\frac{\vartheta}{2}\\ e^{i\varphi}\sin\frac{\vartheta}{2}\end{pmatrix}$$

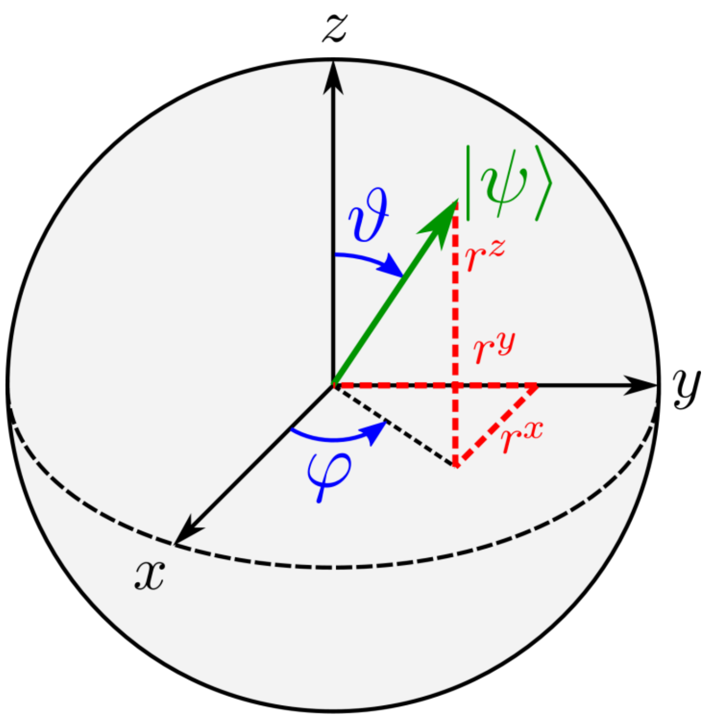
Normalization:

$$\langle\psi|\psi\rangle = |\psi_0|^2 + |\psi_1|^2 = 1$$

$$\langle\psi| = (\psi_0^*,\ \psi_1^*)$$

Global phase:

$$|\psi\rangle \equiv e^{i\Phi}|\psi\rangle$$



JÜLICH Forschungszentrum

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0 \,|0\rangle + \psi_1 \,|1\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \end{pmatrix}$$

Normalization:

$$\langle\psi|\psi\rangle = |\psi_0|^2 + |\psi_1|^2 = 1$$

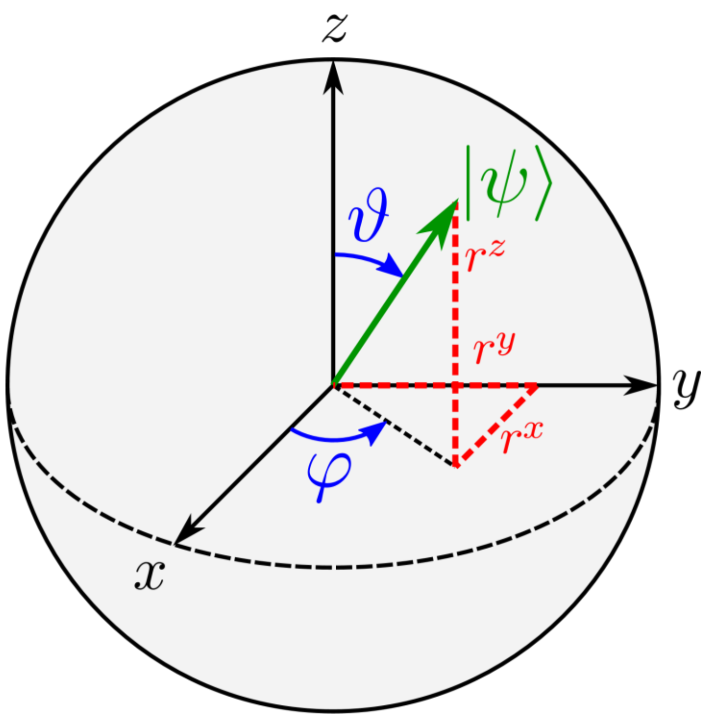
$$\langle\psi| = (\psi_0^*,\ \ \psi_1^*)$$

- Representation on the Bloch sphere

$$|\psi\rangle = \cos\frac{\vartheta}{2}\,|0\rangle + e^{i\varphi}\sin\frac{\vartheta}{2}\,|1\rangle = \begin{pmatrix} \cos\frac{\vartheta}{2} \\ e^{i\varphi}\sin\frac{\vartheta}{2} \end{pmatrix}$$

Global phase:

$$|\psi\rangle \equiv e^{i\Phi}|\psi\rangle$$



- Evaluation of the Cartesian coordinates

$$\vec{r} = \begin{pmatrix} r^x \\ r^y \\ r^z \end{pmatrix} = \begin{pmatrix} \langle\psi|\sigma^x|\psi\rangle \\ \langle\psi|\sigma^y|\psi\rangle \\ \langle\psi|\sigma^z|\psi\rangle \end{pmatrix} = \begin{pmatrix} \sin\vartheta\cos\varphi \\ \sin\vartheta\sin\varphi \\ \cos\vartheta \end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

A single qubit

- Definition of a single qubit

$$|\psi\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle = \begin{pmatrix}\psi_0 \\ \psi_1\end{pmatrix}$$

Normalization:

$$\langle\psi|\psi\rangle = |\psi_0|^2 + |\psi_1|^2 = 1$$

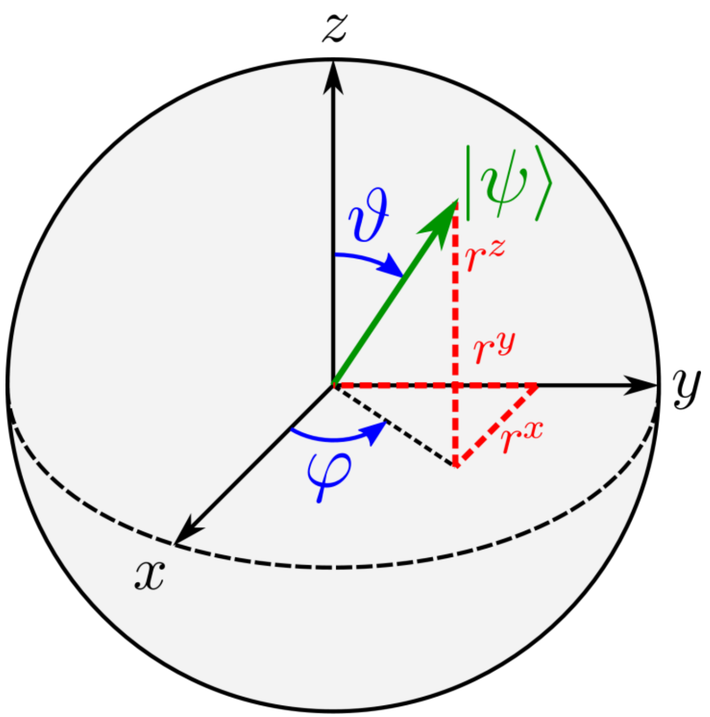
$$\langle\psi| = (\psi_0^*,\ \ \psi_1^*)$$

- Representation on the Bloch sphere

$$|\psi\rangle = \cos\frac{\vartheta}{2}\ |0\rangle + e^{i\varphi}\sin\frac{\vartheta}{2}\ |1\rangle = \begin{pmatrix}\cos\frac{\vartheta}{2} \\ e^{i\varphi}\sin\frac{\vartheta}{2}\end{pmatrix}$$

Global phase:

$$|\psi\rangle \equiv e^{i\Phi}|\psi\rangle$$



- Evaluation of the Cartesian coordinates

$$\vec{r} = \begin{pmatrix}r^x \\ r^y \\ r^z\end{pmatrix} = \begin{pmatrix}\langle\psi|\sigma^x|\psi\rangle \\ \langle\psi|\sigma^y|\psi\rangle \\ \langle\psi|\sigma^z|\psi\rangle\end{pmatrix} = \begin{pmatrix}\sin\vartheta\cos\varphi \\ \sin\vartheta\sin\varphi \\ \cos\vartheta\end{pmatrix}$$

Pauli matrices:

$$\sigma^x = \begin{pmatrix}0 & 1 \\ 1 & 0\end{pmatrix} \quad \sigma^y = \begin{pmatrix}0 & -i \\ i & 0\end{pmatrix} \quad \sigma^z = \begin{pmatrix}1 & 0 \\ 0 & -1\end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

Gates as rotations on the Bloch sphere

QUANTUM BITS AND QUANTUM GATES

Gates as rotations on the Bloch sphere

- Rotations around the axes of the Bloch sphere

$$R^x(\theta) = e^{-i\theta\sigma^x/2} = \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R^y(\theta) = e^{-i\theta\sigma^y/2} = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R^z(\theta) = e^{-i\theta\sigma^z/2} = \begin{pmatrix} \exp(-i\theta/2) & 0 \\ 0 & \exp(i\theta/2) \end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

Gates as rotations on the Bloch sphere

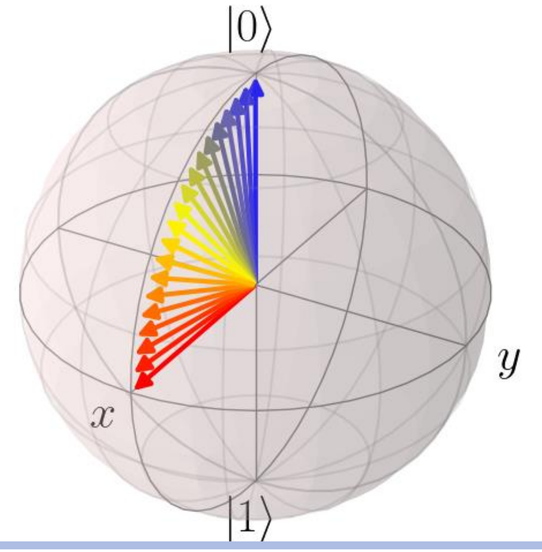
- Rotations around the axes of the Bloch sphere

$$R^x(\theta) = e^{-i\theta\sigma^x/2} = \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R^y(\theta) = e^{-i\theta\sigma^y/2} = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R^z(\theta) = e^{-i\theta\sigma^z/2} = \begin{pmatrix} \exp(-i\theta/2) & 0 \\ 0 & \exp(i\theta/2) \end{pmatrix}$$

$$R^y(\pi/2) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = -\boxed{R^y(\pi/2)}-$$



QUANTUM BITS AND QUANTUM GATES

Gates as rotations on the Bloch sphere

- Rotations around the axes of the Bloch sphere

$$R^x(\theta) = e^{-i\theta\sigma^x/2} = \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

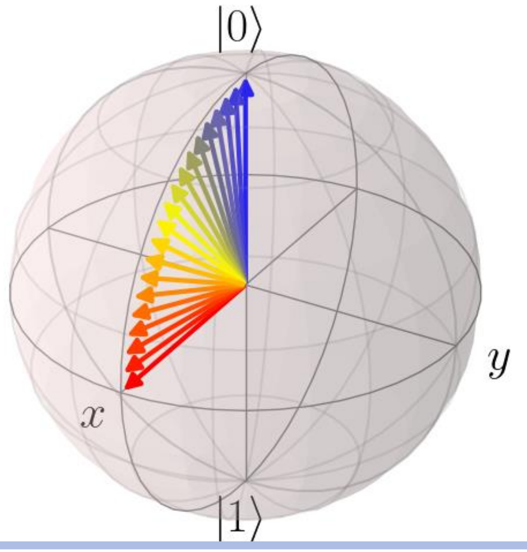
$$R^y(\theta) = e^{-i\theta\sigma^y/2} = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R^z(\theta) = e^{-i\theta\sigma^z/2} = \begin{pmatrix} \exp(-i\theta/2) & 0 \\ 0 & \exp(i\theta/2) \end{pmatrix}$$

- General rotation gate

$$R^{\vec{n}}(\theta) = e^{-i\theta\vec{n}\cdot\vec{\sigma}/2} = \cos\frac{\theta}{2}\, I - i\sin\frac{\theta}{2}\,\vec{n}\cdot\vec{\sigma}$$

$$R^y(\pi/2) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = -\boxed{R^y(\pi/2)}-$$



QUANTUM BITS AND QUANTUM GATES

Gates as rotations on the Bloch sphere

- Rotations around the axes of the Bloch sphere

$$R^x(\theta) = e^{-i\theta\sigma^x/2} = \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

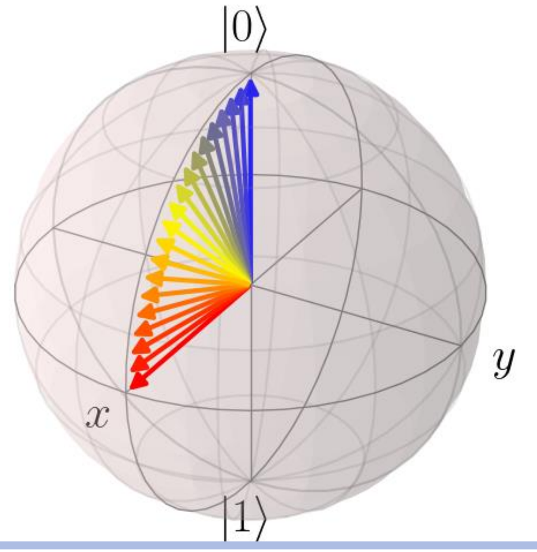
$$R^y(\theta) = e^{-i\theta\sigma^y/2} = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$

$$R^z(\theta) = e^{-i\theta\sigma^z/2} = \begin{pmatrix} \exp(-i\theta/2) & 0 \\ 0 & \exp(i\theta/2) \end{pmatrix}$$

- General rotation gate

$$R^{\vec{n}}(\theta) = e^{-i\theta\vec{n}\cdot\vec{\sigma}/2} = \cos\frac{\theta}{2}\, I - i\sin\frac{\theta}{2}\,\vec{n}\cdot\vec{\sigma}$$

$$R^y(\pi/2) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -1 \\ 1 & 1 \end{pmatrix} = -\boxed{R^y(\pi/2)}-$$



Standard gate set:

$$X = \sigma^x \qquad Y = \sigma^y \qquad Z = \sigma^z$$

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \qquad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

- Multi-qubit state

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j |j\rangle = \psi_0 |0\cdots 00\rangle + \psi_1 |0\cdots 01\rangle + \cdots + \psi_{2^n-1} |1\cdots 11\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{2^n-1} \end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

Computational basis states:

$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \quad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

- Multi-qubit state

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j \, |j\rangle = \psi_0 \, |0\cdots 00\rangle + \psi_1 \, |0\cdots 01\rangle + \cdots + \psi_{2^n-1} \, |1\cdots 11\rangle = \begin{pmatrix}\psi_0\\ \psi_1\\ \vdots \\ \psi_{2^n-1}\end{pmatrix}$$

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

- Multi-qubit state

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j |j\rangle = \psi_0 |0\cdots 00\rangle + \psi_1 |0\cdots 01\rangle + \cdots + \psi_{2^n-1} |1\cdots 11\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{2^n-1} \end{pmatrix}$$

Computational basis states:

$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Tensor product

$$|q_0 q_1\rangle = |q_0\rangle \otimes |q_1\rangle \qquad A \otimes B = \begin{pmatrix} a_{00}B & a_{01}B & \cdots \\ a_{10}B & a_{11}B & \\ \vdots & & \ddots \end{pmatrix}$$

JÜLICH Forschungszentrum

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

Computational basis states:

$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \quad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

- Multi-qubit state

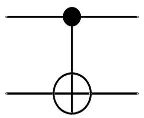
$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j |j\rangle = \psi_0 |0\cdots 00\rangle + \psi_1 |0\cdots 01\rangle + \cdots + \psi_{2^n-1} |1\cdots 11\rangle = \begin{pmatrix}\psi_0\\ \psi_1\\ \vdots \\ \psi_{2^n-1}\end{pmatrix}$$

Tensor product

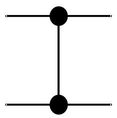
$$|q_0 q_1\rangle = |q_0\rangle \otimes |q_1\rangle \qquad A \otimes B = \begin{pmatrix} a_{00}B & a_{01}B & \cdots \\ a_{10}B & a_{11}B & \\ \vdots & & \ddots \end{pmatrix}$$

- Important two-qubit gates

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



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



QUANTUM BITS AND QUANTUM GATES

Multiple qubits

- Multi-qubit state

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j |j\rangle = \psi_0 |0\cdots 00\rangle + \psi_1 |0\cdots 01\rangle + \cdots + \psi_{2^n-1} |1\cdots 11\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{2^n-1} \end{pmatrix}$$

Computational basis states:

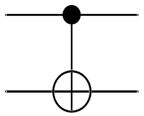
$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Tensor product

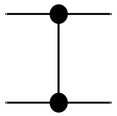
$$|q_0 q_1\rangle = |q_0\rangle \otimes |q_1\rangle \qquad A \otimes B = \begin{pmatrix} a_{00}B & a_{01}B & \cdots \\ a_{10}B & a_{11}B & \\ \vdots & & \ddots \end{pmatrix}$$

- Important two-qubit gates

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



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



JÜLICH Forschungszentrum

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

Computational basis states:

$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \quad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

- Multi-qubit state

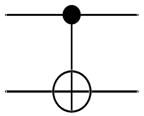
$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j\,|j\rangle = \psi_0\,|0\cdots00\rangle + \psi_1\,|0\cdots01\rangle + \cdots + \psi_{2^n-1}\,|1\cdots11\rangle = \begin{pmatrix}\psi_0\\ \psi_1\\ \vdots\\ \psi_{2^n-1}\end{pmatrix}$$

Tensor product

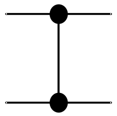
$$|q_0q_1\rangle = |q_0\rangle \otimes |q_1\rangle \qquad A \otimes B = \begin{pmatrix} a_{00}B & a_{01}B & \cdots \\ a_{10}B & a_{11}B & \\ \vdots & & \ddots \end{pmatrix}$$

- Important two-qubit gates

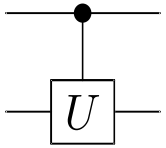
$$\mathrm{CNOT} = \mathrm{CX} = \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} =$$



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



$$\mathrm{C}U = \begin{pmatrix} I & 0 \\ 0 & U \end{pmatrix} =$$



QUANTUM BITS AND QUANTUM GATES

Multiple qubits

➢ Multi-qubit state

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j |j\rangle = \psi_0 |0\cdots00\rangle + \psi_1 |0\cdots01\rangle + \cdots + \psi_{2^n-1} |1\cdots11\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{2^n-1} \end{pmatrix}$$

Computational basis states:

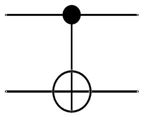
$$|00\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad |01\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix} \quad |10\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad |11\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

Tensor product

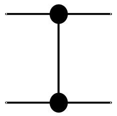
$$|q_0 q_1\rangle = |q_0\rangle \otimes |q_1\rangle \qquad A \otimes B = \begin{pmatrix} a_{00}B & a_{01}B & \cdots \\ a_{10}B & a_{11}B & \\ \vdots & & \ddots \end{pmatrix}$$

➢ Important two-qubit gates

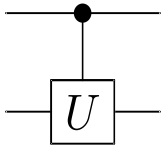
$$\mathrm{CNOT} = \mathrm{CX} = \begin{array}{c} \begin{matrix} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{matrix} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} =$$



$$CZ = \begin{array}{c} \begin{matrix} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{matrix} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \end{array} =$$



$$\mathrm{CU} = \begin{pmatrix} I & 0 \\ 0 & U \end{pmatrix} =$$



➢ General controlled gates

$$\mathrm{CU}_{i_1 i_2} |q_0 \cdots q_{n-1}\rangle = \begin{cases} |q_0 \cdots q_{n-1}\rangle & (\text{if } q_{i_1} = 0) \\ |q_0 \cdots q_{i_2-1}\rangle \left(U |q_{i_2}\rangle\right) |q_{i_2+1} \cdots q_{n-1}\rangle & (\text{if } q_{i_1} = 1) \end{cases}$$

QUANTUM BITS AND QUANTUM GATES

Multiple qubits

- Multi-qubit state

$$|\psi\rangle = \sum_{j=0}^{2^n-1} \psi_j |j\rangle = \psi_0 |0\cdots 00\rangle + \psi_1 |0\cdots 01\rangle + \cdots + \psi_{2^n-1} |1\cdots 11\rangle = \begin{pmatrix} \psi_0 \\ \psi_1 \\ \vdots \\ \psi_{2^n-1} \end{pmatrix}$$

Computational basis states:

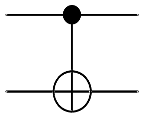
$$|00\rangle = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad |01\rangle = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \quad |10\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \quad |11\rangle = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}$$

Tensor product

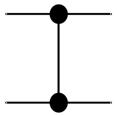
$$|q_0 q_1\rangle = |q_0\rangle \otimes |q_1\rangle \qquad A \otimes B = \begin{pmatrix} a_{00}B & a_{01}B & \cdots \\ a_{10}B & a_{11}B & \\ \vdots & & \ddots \end{pmatrix}$$

- Important two-qubit gates

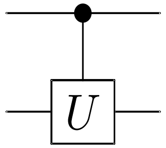
$$\mathrm{CNOT} = \mathrm{CX} = \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array} =$$



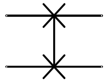
$$\mathrm{CZ} = \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} \end{array} =$$



$$\mathrm{CU} = \begin{pmatrix} I & 0 \\ 0 & U \end{pmatrix} =$$



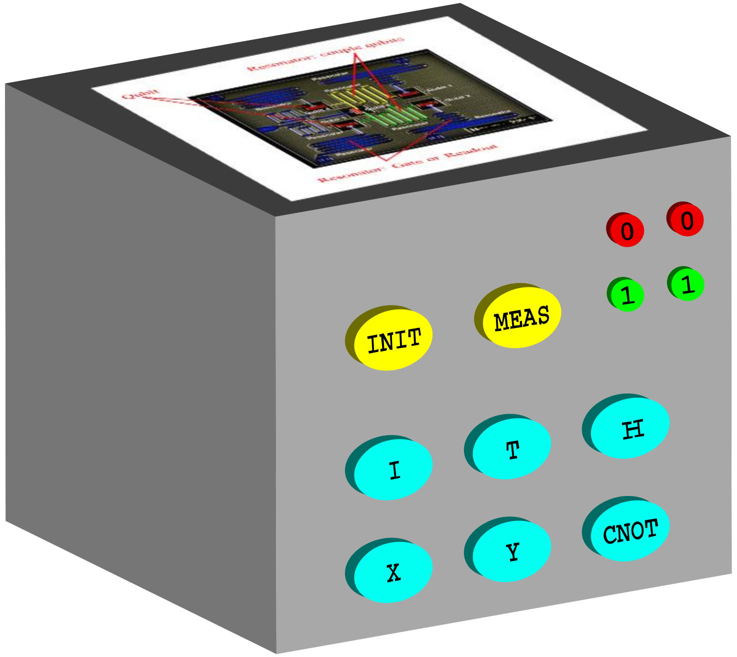
$$\mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} =$$



- General controlled gates

$$\mathrm{CU}_{i_1 i_2} |q_0 \cdots q_{n-1}\rangle = \begin{cases} |q_0 \cdots q_{n-1}\rangle & (\text{if } q_{i_1} = 0) \\ |q_0 \cdots q_{i_2-1}\rangle (U |q_{i_2}\rangle) |q_{i_2+1} \cdots q_{n-1}\rangle & (\text{if } q_{i_1} = 1) \end{cases}$$

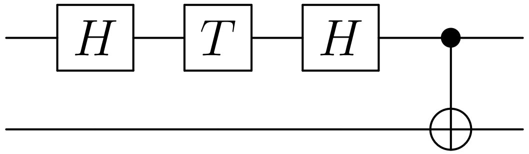
PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

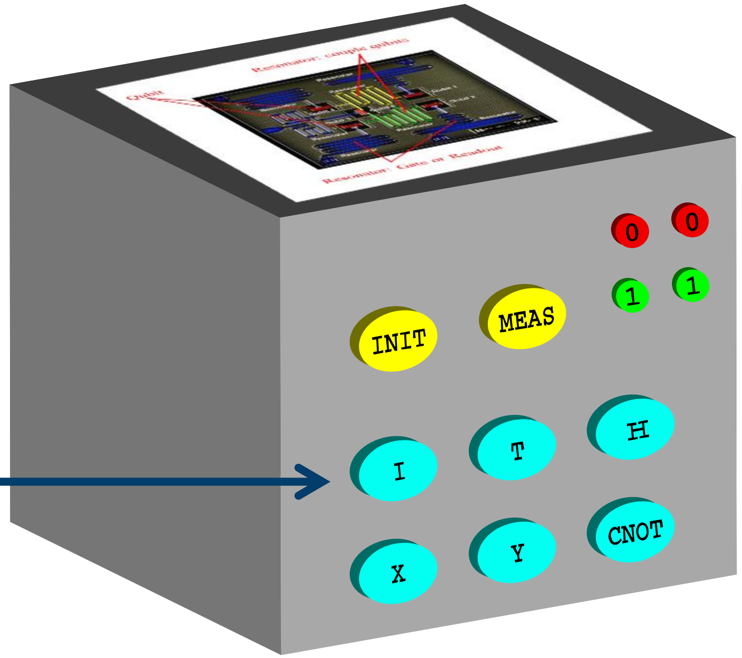


PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates

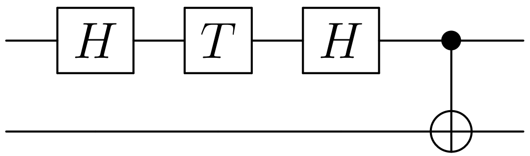


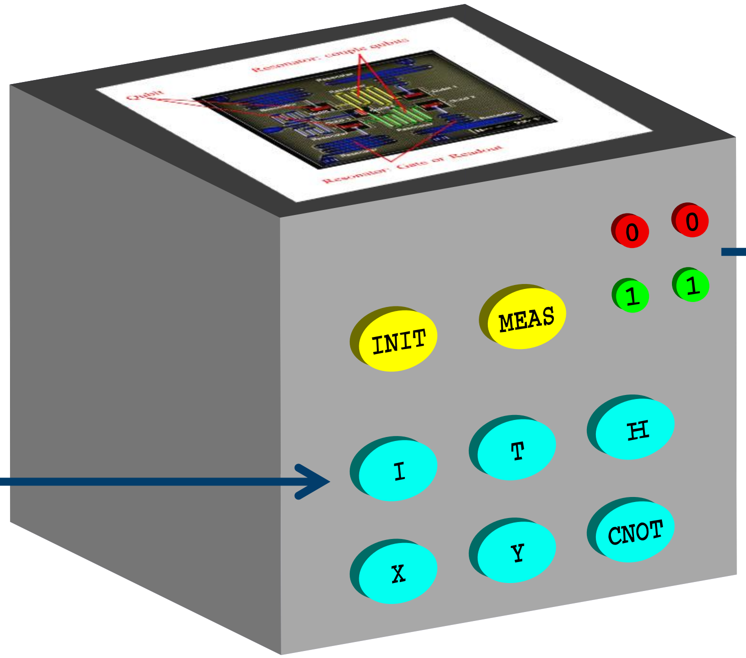


PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates





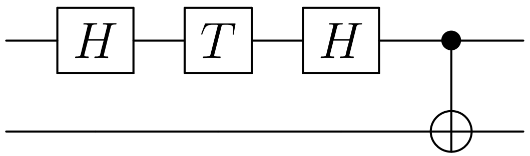
Output: Bitstrings

„00“, „01“, „10“, „11“

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

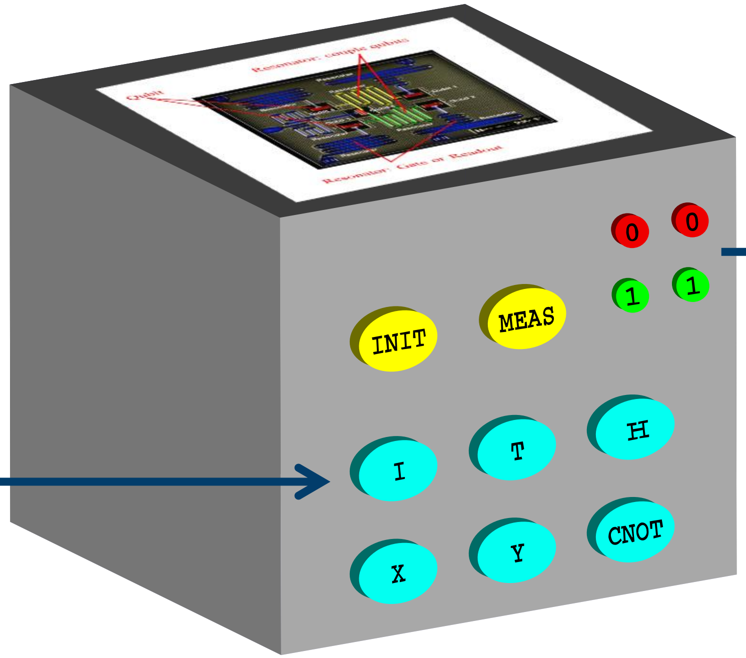
Input:

Programm =
Circuit of
Quantum Gates



Result of this program

Option 1: Compute the unitary matrix



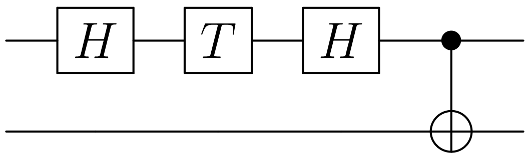
Output: Bitstrings

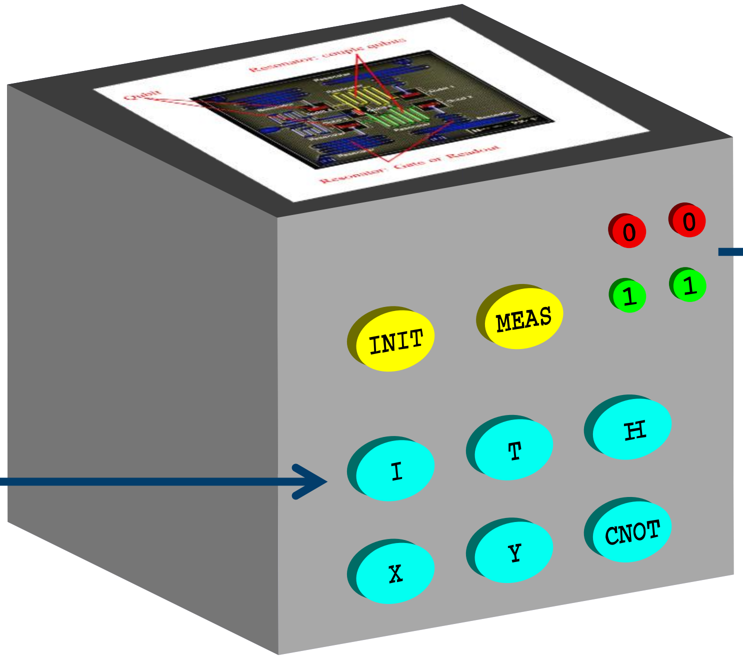
„00“, „01“, „10“, „11“

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

Result of this program

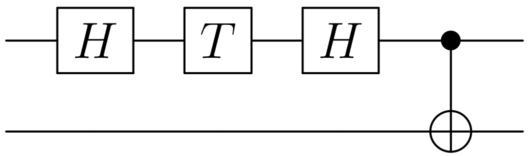
Option 1: Compute the unitary matrix

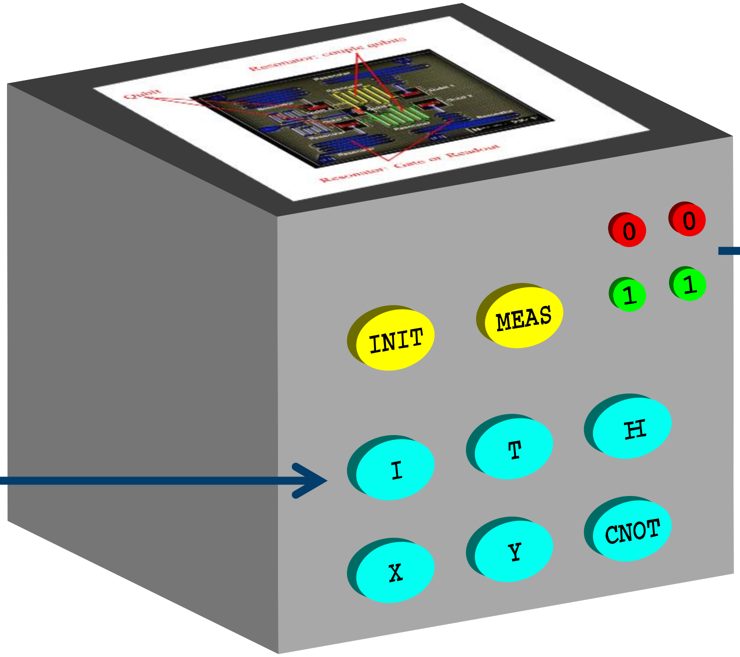
$$\mathrm{CNOT}\,(H\otimes I)\,(T\otimes I)\,(H\otimes I) = \begin{pmatrix} \frac{1+e^{i\pi/4}}{2} & 0 & \frac{1-e^{i\pi/4}}{2} & 0 & \\ 0 & \frac{1+e^{i\pi/4}}{2} & 0 & \frac{1-e^{i\pi/4}}{2} & \\ 0 & \frac{1-e^{i\pi/4}}{2} & 0 & \frac{1+e^{i\pi/4}}{2} & 0 \\ \frac{1-e^{i\pi/4}}{2} & 0 & \frac{1+e^{i\pi/4}}{2} & 0 & \end{pmatrix}$$

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

$$\sqrt{0.85\ldots}|00\rangle + \sqrt{0.15\ldots}|11\rangle$$

Result of this program

Option 1: Compute the unitary matrix

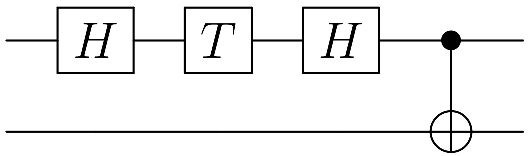
$$\mathrm{CNOT}\,(H \otimes I)\,(T \otimes I)\,(H \otimes I) = \begin{pmatrix} \boxed{\begin{matrix} \frac{1+e^{i\pi/4}}{2} \\ 0 \\ 0 \\ \frac{1-e^{i\pi/4}}{2} \end{matrix}} & \begin{matrix} 0 \\ \frac{1+e^{i\pi/4}}{2} \\ \frac{1-e^{i\pi/4}}{2} \\ 0 \end{matrix} & \begin{matrix} \frac{1-e^{i\pi/4}}{2} \\ 0 \\ 0 \\ \frac{1+e^{i\pi/4}}{2} \end{matrix} & \begin{matrix} 0 \\ \frac{1-e^{i\pi/4}}{2} \\ \frac{1+e^{i\pi/4}}{2} \\ 0 \end{matrix} & \begin{matrix} \\ \\ 0 \\ \\ \end{matrix} \end{pmatrix}$$

JÜLICH Forschungszentrum

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

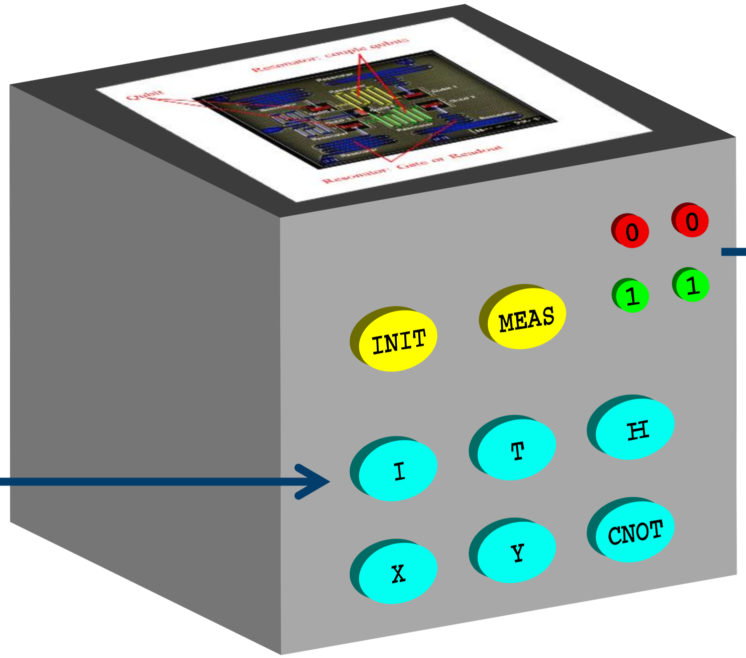
Input:

Programm =
Circuit of
Quantum Gates



Result of this program

Option 2: Track the unitary transformations



Output: Bitstrings

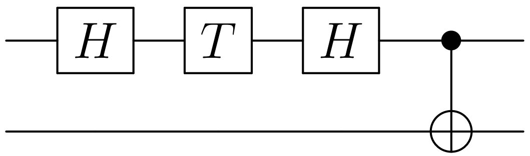
„00“, „01“, „10“, „11“

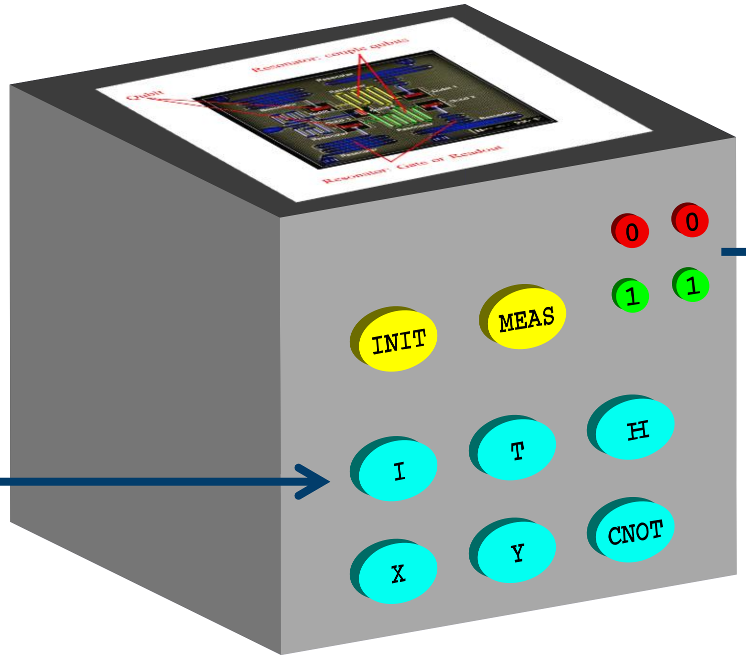
JÜLICH
Forschungszentrum

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm = Circuit of Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

Result of this program

Option 2: Track the unitary transformations

$$|00\rangle \overset{H\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \overset{T\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

$$\overset{H\otimes I}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

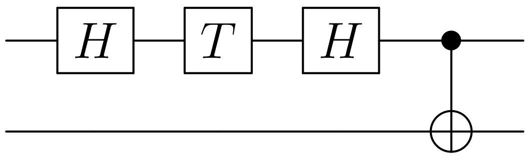
$$\overset{\text{CNOT}}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|11\rangle$$

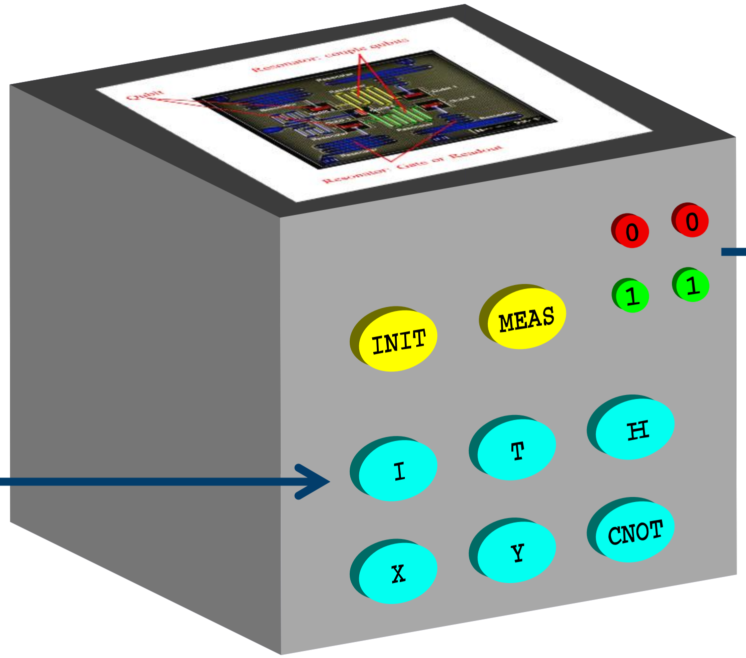
JÜLICH Forschungszentrum

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm = Circuit of Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

Result of this program

Option 2: Track the unitary transformations

$$|00\rangle \overset{H\otimes I}{\mapsto} \boxed{\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle} \overset{T\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

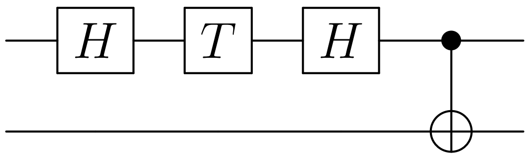
$$\overset{H\otimes I}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

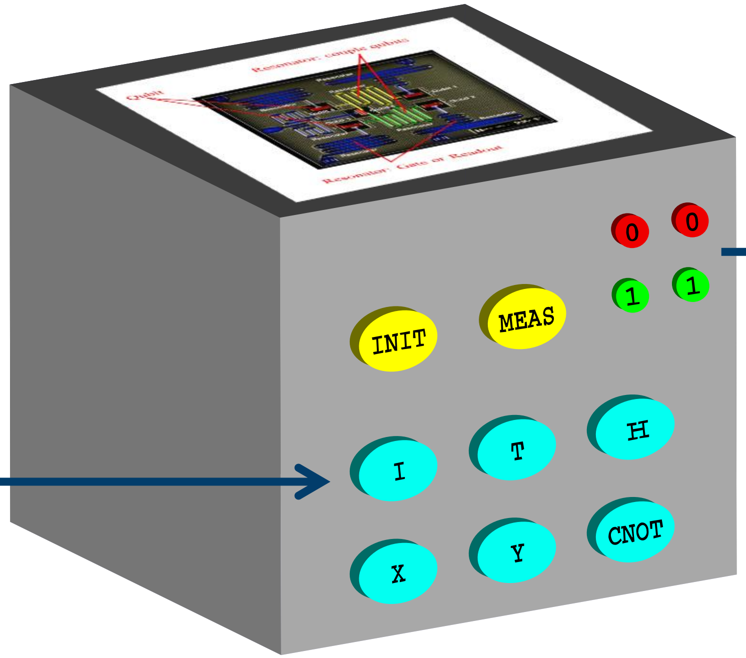
$$\overset{\text{CNOT}}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|11\rangle$$

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

Result of this program

Option 2: Track the unitary transformations

$$|00\rangle \overset{H\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \boxed{\overset{T\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|10\rangle}$$

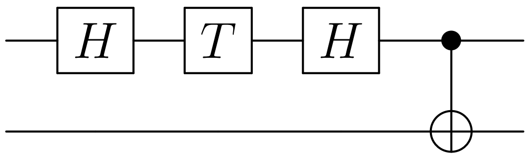
$$\overset{H\otimes I}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

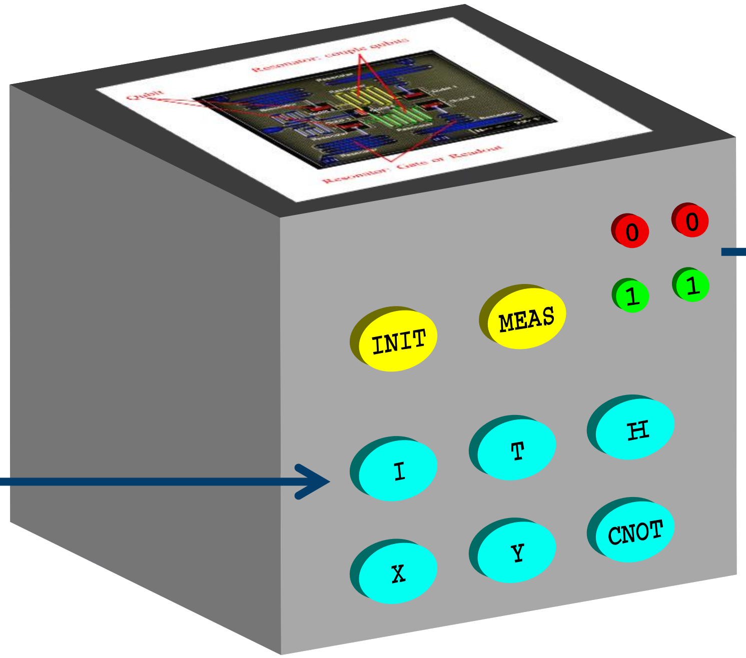
$$\overset{\text{CNOT}}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|11\rangle$$

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

Result of this program

Option 2: Track the unitary transformations

$$|00\rangle \overset{H\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \overset{T\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

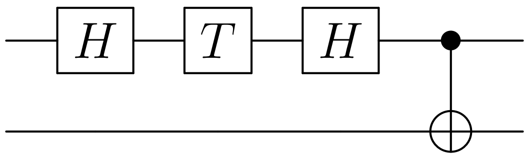
$$\boxed{\overset{H\otimes I}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|10\rangle}$$

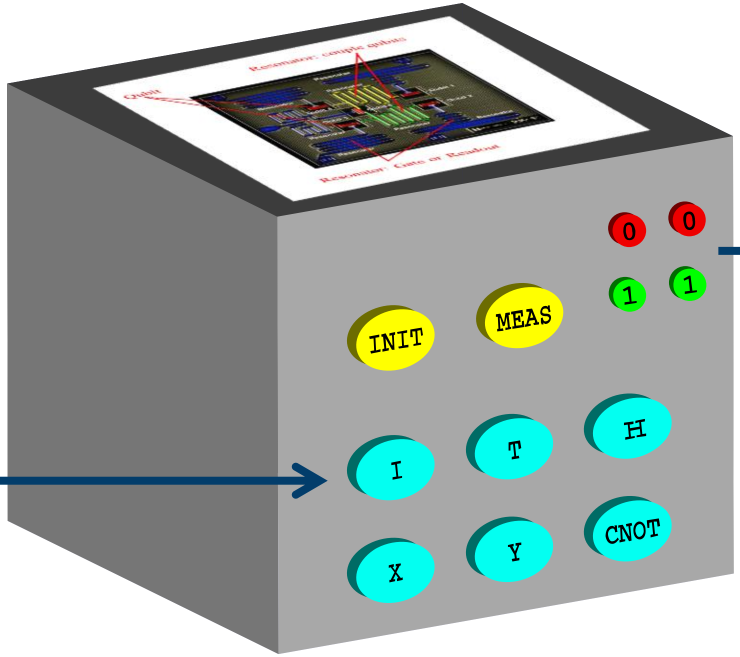
$$\overset{\text{CNOT}}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|11\rangle$$

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

Result of this program

Option 2: Track the unitary transformations

$$|00\rangle \overset{H\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \overset{T\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

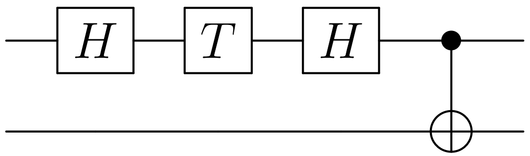
$$\overset{H\otimes I}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

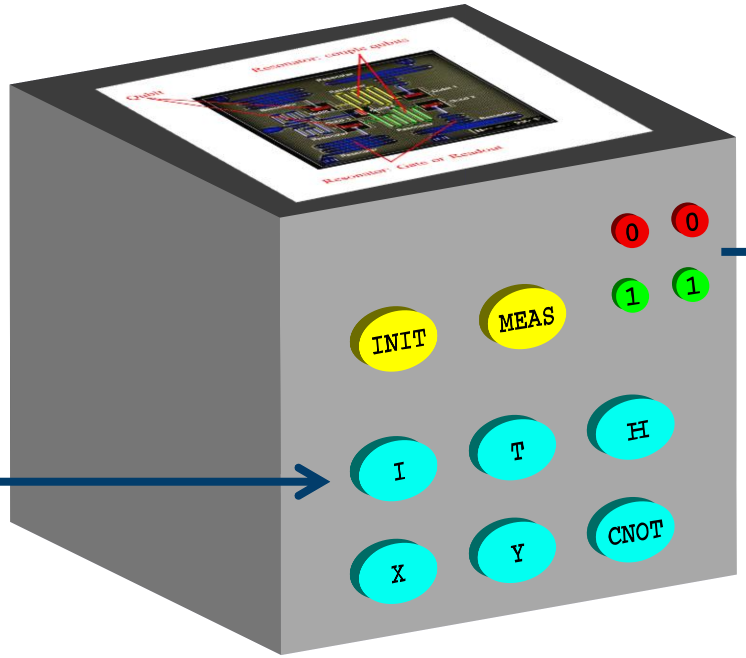
$$\boxed{\overset{\text{CNOT}}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|11\rangle}$$

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm = Circuit of Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

$$\sqrt{0.85\ldots}|00\rangle + \sqrt{0.15\ldots}|11\rangle$$

Result of this program

Option 2: Track the unitary transformations

$$|00\rangle \overset{H\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle \overset{T\otimes I}{\mapsto} \frac{1}{\sqrt{2}}|00\rangle + \frac{e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

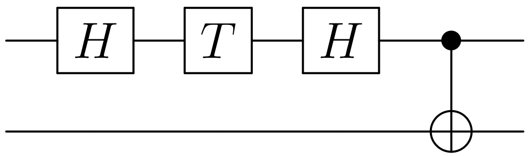
$$\overset{H\otimes I}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|10\rangle$$

$$\overset{\mathrm{CNOT}}{\mapsto} \frac{1+e^{i\pi/4}}{\sqrt{2}}|00\rangle + \frac{1-e^{i\pi/4}}{\sqrt{2}}|11\rangle$$

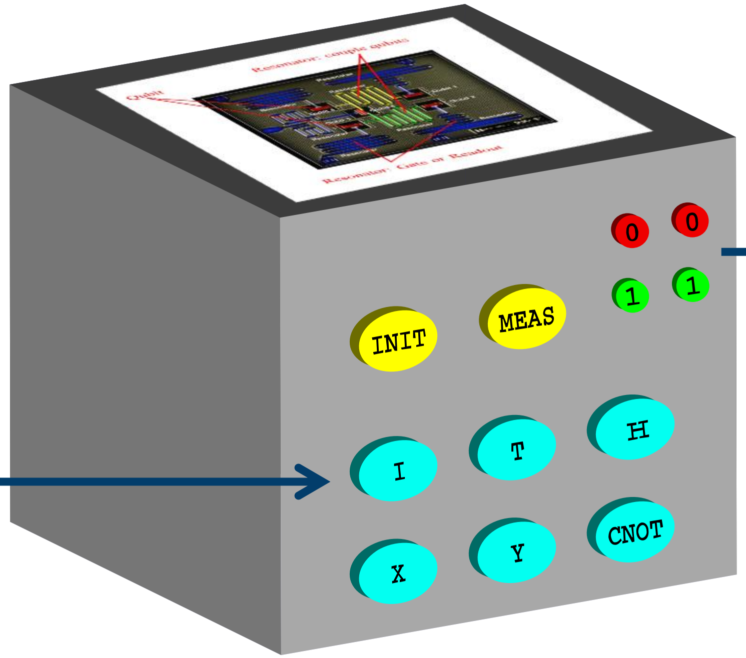
PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =
Circuit of
Quantum Gates



Option 3: Simulate the circuit



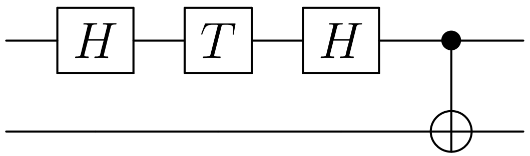
Output: Bitstrings

„00“, „01“, „10“, „11“

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

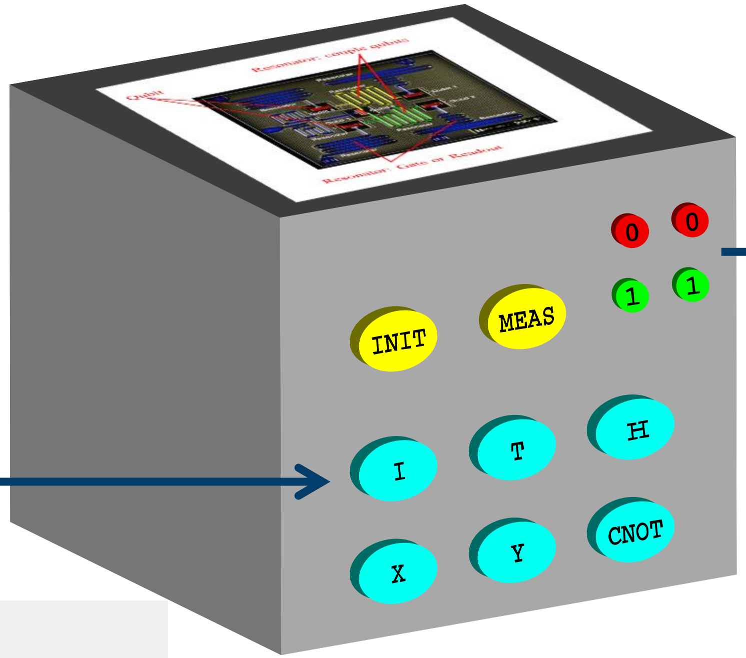
Input:

Programm =
Circuit of
Quantum Gates



Option 3: Simulate the circuit

```
import qiskit

circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)

from juqcs import Juqcs
```



Output: Bitstrings

„00“, „01“, „10“, „11“

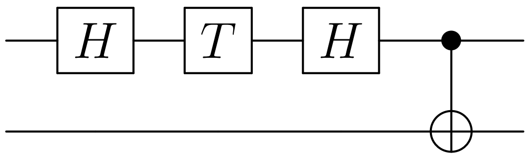
PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

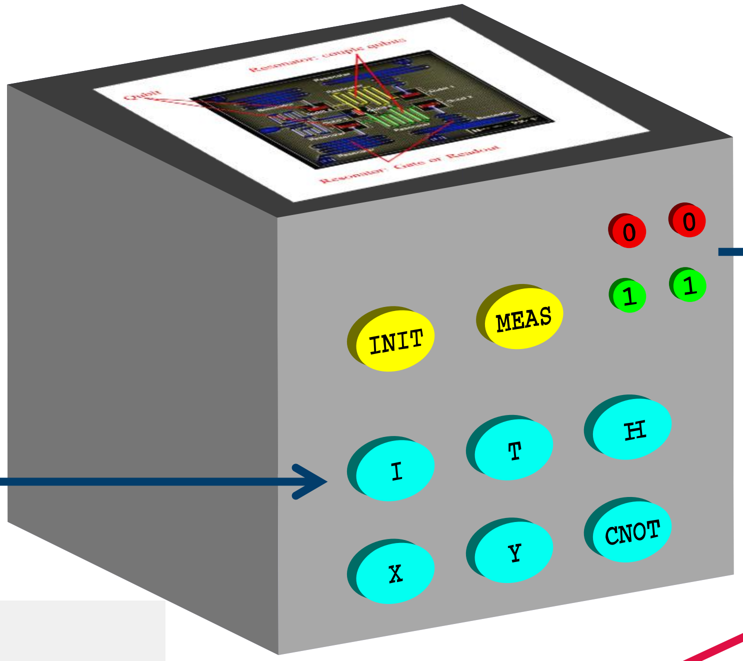
Input:

Programm =

Circuit of

Quantum Gates





Output: Bitstrings

„00“, „01“, „10“, „11“

$$\sqrt{0.85\ldots}|00\rangle + \sqrt{0.15\ldots}|11\rangle$$

Option 3: Simulate the circuit

```
import qiskit

circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)

from juqcs import Juqcs
```

JUQCS (Jülich Universal Quantum Computer Simulator)

Lecture Notes:
Programming Quantum Computers
https://arxiv.org/pdf/2201.02051.pdf

JÜLICH Forschungszentrum

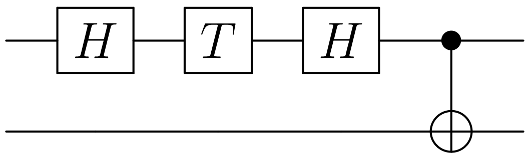
PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Input:

Programm =

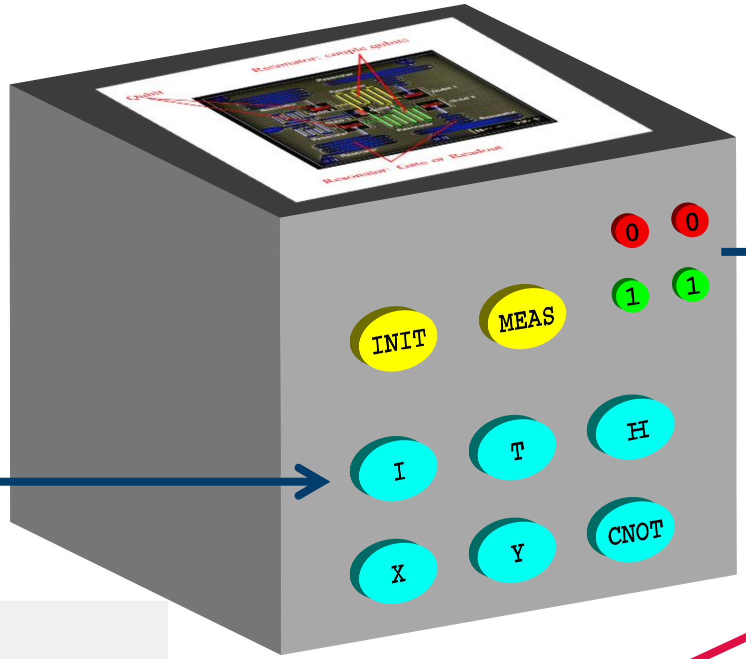
Circuit of

Quantum Gates



Option 3: Simulate the circuit

```
import qiskit

circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)

from juqcs import Juqcs
```



Output: Bitstrings

„00“, „01“, „10“, „11“

$$\sqrt{0.85\ldots}|00\rangle + \sqrt{0.15\ldots}|11\rangle$$

JUQCS
(Jülich Universal Quantum Computer Simulator)

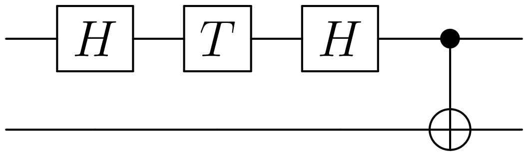
Measure **00** with 85 %

and **11** with 15 %

Lecture Notes:
Programming Quantum Computers
https://arxiv.org/pdf/2201.02051.pdf

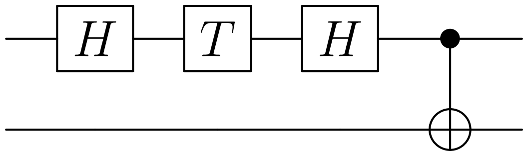
PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Simulation with Qiskit Aer



PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Simulation with Qiskit Aer



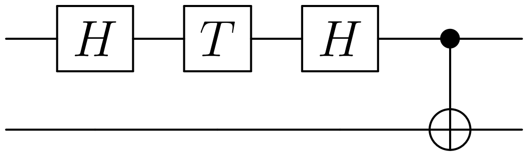
qasm_simulator

```
import qiskit
circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)
circuit.measure_all()

backend = qiskit.Aer.get_backend('qasm_simulator')
result = qiskit.execute(circuit.reverse_bits(),
                        backend=backend,
                        shots=1000).result()
result.get_counts()
```

```
{'00': 856, '11': 144}
```

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Simulation with Qiskit Aer



qasm_simulator

```
import qiskit
circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)
circuit.measure_all()

backend = qiskit.Aer.get_backend('qasm_simulator')
result = qiskit.execute(circuit.reverse_bits(),
                        backend=backend,
                        shots=1000).result()
result.get_counts()
```

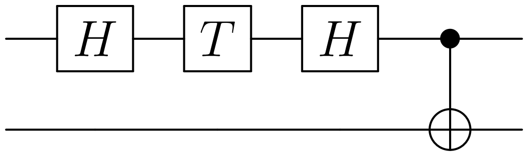
```
{'00': 856, '11': 144}
```

Measure **00** with 85 %

and **11** with 15 %

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Simulation with Qiskit Aer



qasm_simulator

```
import qiskit
circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)
circuit.measure_all()

backend = qiskit.Aer.get_backend('qasm_simulator')
result = qiskit.execute(circuit.reverse_bits(),
                        backend=backend,
                        shots=1000).result()
result.get_counts()
```

```
{'00': 856, '11': 144}
```

Measure **00** with 85 %

and **11** with 15 %

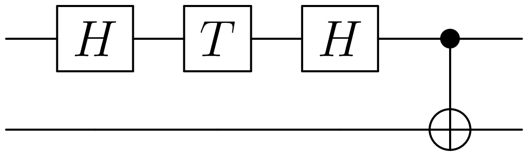
statevector_simulator

```
import qiskit
circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)


backend = qiskit.Aer.get_backend('statevector_simulator')
result = qiskit.execute(circuit.reverse_bits(),
                        backend=backend).result()

result.get_statevector()
```

```
array([0.85355339+0.35355339j, 0.        +0.j        ,
       0.        +0.j        , 0.14644661-0.35355339j])
```

PROGRAMMING AND SIMULATING QUANTUM CIRCUITS

Simulation with Qiskit Aer



qasm_simulator

```python
import qiskit
circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)
circuit.measure_all()

backend = qiskit.Aer.get_backend('qasm_simulator')
result = qiskit.execute(circuit.reverse_bits(),
                        backend=backend,
                        shots=1000).result()
result.get_counts()
```

```
{'00': 856, '11': 144}
```

Measure **00** with 85 %

and **11** with 15 %

statevector_simulator

```python
import qiskit
circuit = qiskit.QuantumCircuit(2)
circuit.h(0)
circuit.t(0)
circuit.h(0)
circuit.cx(0,1)


backend = qiskit.Aer.get_backend('statevector_simulator')
result = qiskit.execute(circuit.reverse_bits(),
                        backend=backend).result()

result.get_statevector()
```

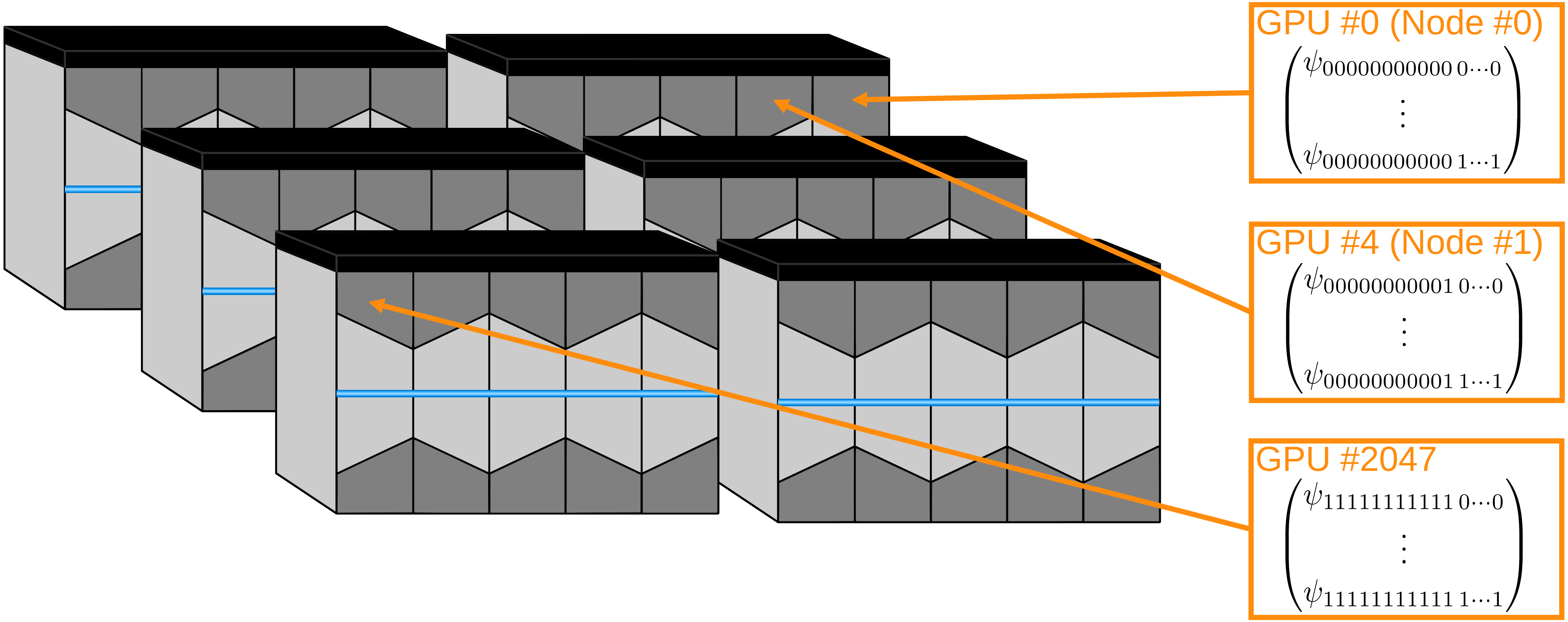
```
array([0.85355339+0.35355339j, 0.        +0.j        ,
       0.        +0.j        , 0.14644661-0.35355339j])
```

$$\sqrt{0.85\ldots}|00\rangle + \sqrt{0.15\ldots}|11\rangle$$

De Raedt et al., CPC **176**, 121 (2007)
De Raedt et al., CPC **237**, 47 (2019)
Willsch et al., CPC **278**, 108411 (2022)

SIMULATING QUANTUM CIRCUITS

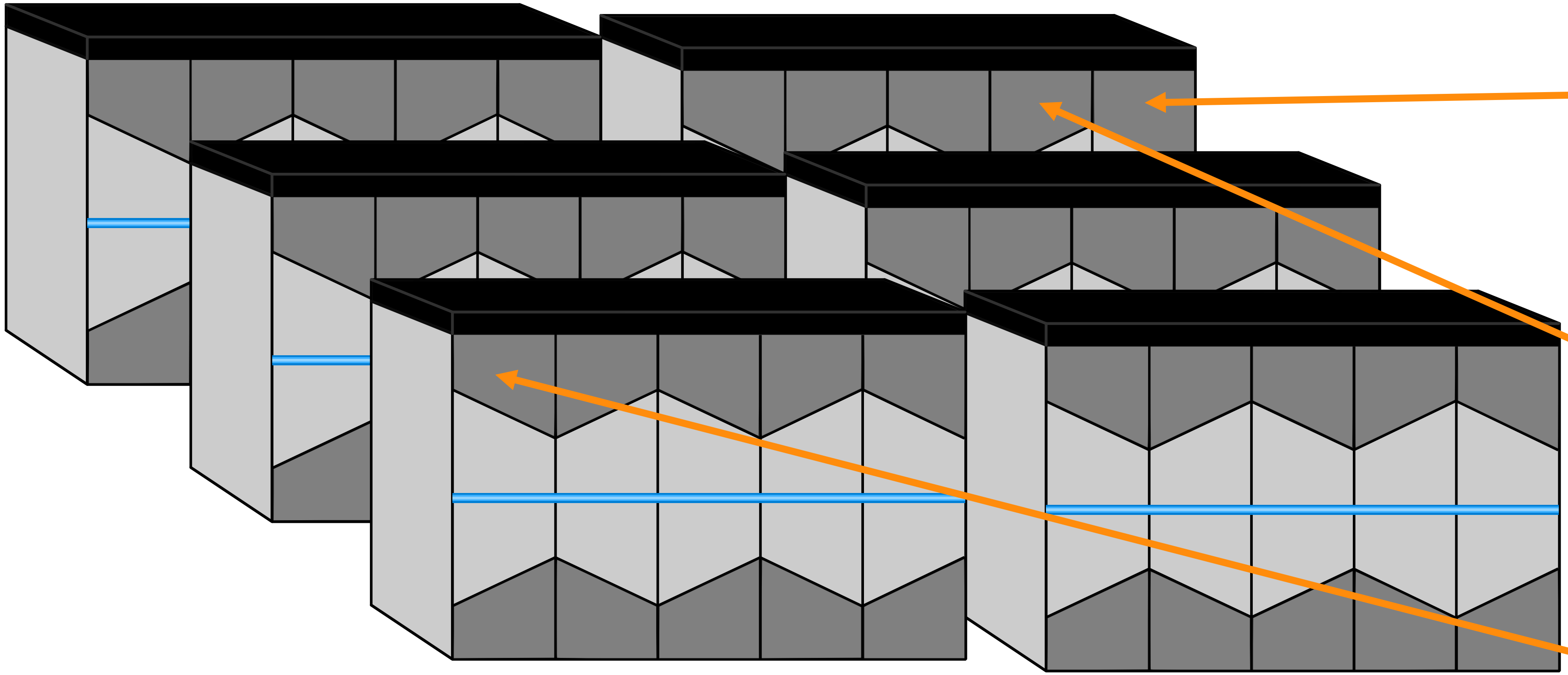
Simulation with JUQCS (large-scale simulations on a supercomputer)



De Raedt et al., CPC **176**, 121 (2007)
De Raedt et al., CPC **237**, 47 (2019)
Willsch et al., CPC **278**, 108411 (2022)

SIMULATING QUANTUM CIRCUITS

Simulation with JUQCS (large-scale simulations on a supercomputer)



GPU #0 (Node #0)

$$\begin{pmatrix} \psi_{00000000000\,0\cdots0} \\ \vdots \\ \psi_{00000000000\,1\cdots1} \end{pmatrix}$$

GPU #4 (Node #1)

$$\begin{pmatrix} \psi_{00000000001\,0\cdots0} \\ \vdots \\ \psi_{00000000001\,1\cdots1} \end{pmatrix}$$

GPU #2047

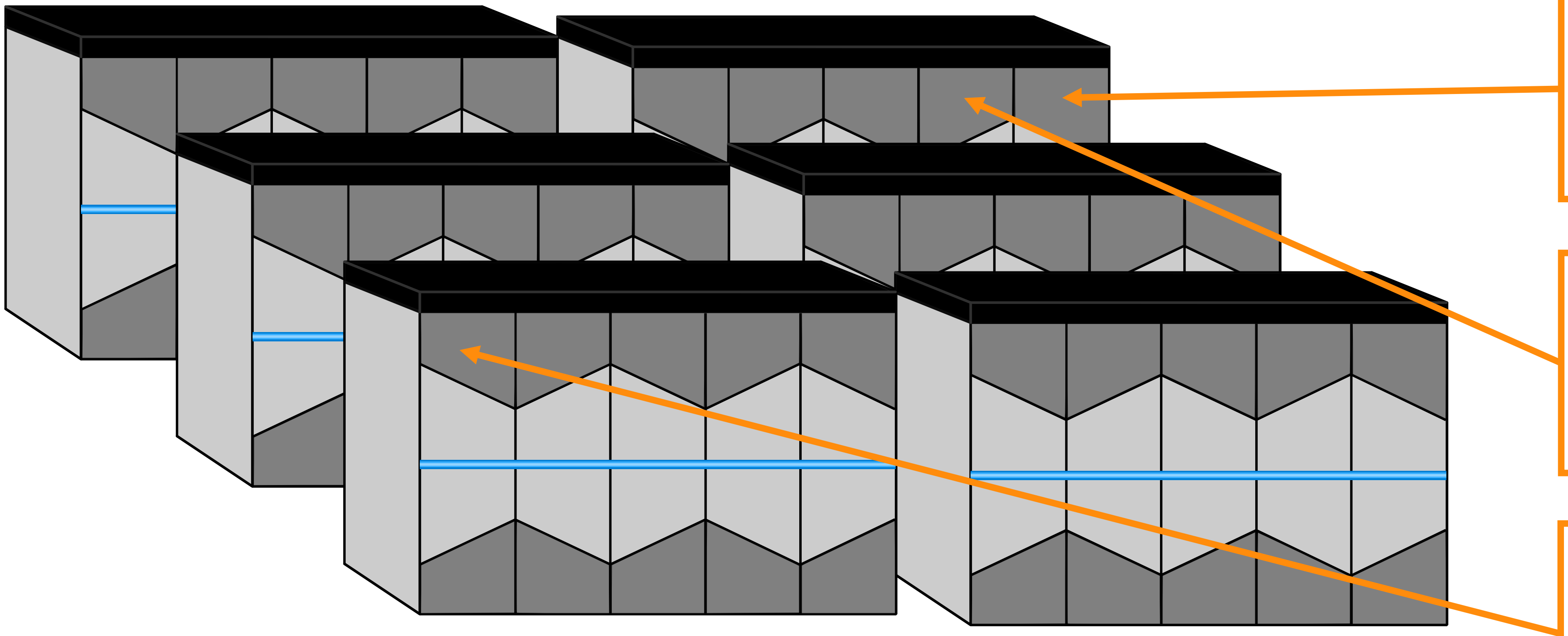
$$\begin{pmatrix} \psi_{11111111111\,0\cdots0} \\ \vdots \\ \psi_{11111111111\,1\cdots1} \end{pmatrix}$$

- **Video:** Simulation of Quantum Computers on GPUs
  https://youtu.be/JUSEYCkh2ac

De Raedt et al., CPC **176**, 121 (2007)
De Raedt et al., CPC **237**, 47 (2019)
Willsch et al., CPC **278**, 108411 (2022)

SIMULATING QUANTUM CIRCUITS

Simulation with JUQCS (large-scale simulations on a supercomputer)



GPU #0 (Node #0)

$$\begin{pmatrix} \psi_{00000000000\,0\cdots 0} \\ \vdots \\ \psi_{00000000000\,1\cdots 1} \end{pmatrix}$$

GPU #4 (Node #1)

$$\begin{pmatrix} \psi_{00000000001\,0\cdots 0} \\ \vdots \\ \psi_{00000000001\,1\cdots 1} \end{pmatrix}$$

GPU #2047

$$\begin{pmatrix} \psi_{11111111111\,0\cdots 0} \\ \vdots \\ \psi_{11111111111\,1\cdots 1} \end{pmatrix}$$

- **Video:** Simulation of Quantum Computers on GPUs
  https://youtu.be/JUSEYCkh2ac
- **JUNIQ:** Possible to simulate around 40 qubits (45 with JUQCS)
  https://juniq.fz-juelich.de

JÜLICH Forschungszentrum

QUANTUM FOURIER TRANSFORM

An operation to move registers into phases and back

QUANTUM FOURIER TRANSFORM

An operation to move registers into phases and back

- The formal definition of the QFT is:

$$\boxed{|q_0 q_1 q_2 \cdots\rangle \overset{\text{QFT}}{\mapsto} \sum_{q'_0 q'_1 q'_2 \cdots} e^{2\pi i\,(q_0 q_1 q_2 \cdots)\,(q'_0 q'_1 q'_2 \cdots)/2^N}\,|q'_0 q'_1 q'_2 \cdots\rangle}$$

JÜLICH Forschungszentrum

QUANTUM FOURIER TRANSFORM

An operation to move registers into phases and back

- The formal definition of the QFT is:

$$\boxed{|q_0 q_1 q_2 \cdots\rangle \overset{\text{QFT}}{\mapsto} \sum_{q_0' q_1' q_2' \cdots} e^{2\pi i\,(q_0 q_1 q_2 \cdots)\,(q_0' q_1' q_2' \cdots)/2^N}\, |q_0' q_1' q_2' \cdots\rangle}$$

- Intuition:

$$|j\rangle \leftrightarrow \sum_{j'} e^{2\pi i\, j\, j'/2^n}\, |j'\rangle$$

QUANTUM FOURIER TRANSFORM

An operation to move registers into phases and back

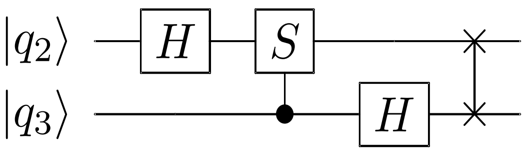
- The formal definition of the QFT is:

$$\boxed{|q_0 q_1 q_2 \cdots\rangle \overset{\text{QFT}}{\mapsto} \sum_{q'_0 q'_1 q'_2 \cdots} e^{2\pi i\,(q_0 q_1 q_2 \cdots)\,(q'_0 q'_1 q'_2 \cdots)/2^N}\,|q'_0 q'_1 q'_2 \cdots\rangle}$$

- Intuition:

$$|j\rangle \leftrightarrow \sum_{j'} e^{2\pi i\, j\, j'/2^n}\,|j'\rangle$$

- Implementation for two qubits:



QUANTUM FOURIER TRANSFORM

An operation to move registers into phases and back

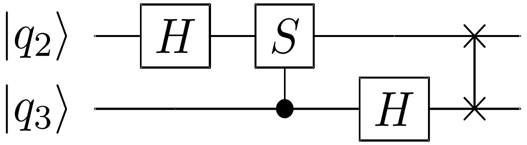
- The formal definition of the QFT is:

$$\boxed{|q_0 q_1 q_2 \cdots\rangle \overset{\mathrm{QFT}}{\mapsto} \sum_{q_0' q_1' q_2' \cdots} e^{2\pi i\,(q_0 q_1 q_2 \cdots)\,(q_0' q_1' q_2' \cdots)/2^N}\,|q_0' q_1' q_2' \cdots\rangle}$$

- Intuition:

$$|j\rangle \leftrightarrow \sum_{j'} e^{2\pi i\, j\, j'/2^n}\,|j'\rangle$$

- Implementation for two qubits:



- Verification:

$$\mathrm{SWAP}\,(I\otimes H)\,CS\,(H\otimes I)\,|q_2 q_3\rangle \propto \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \end{array} |q_2 q_3\rangle = \sum_{q_2' q_3'} e^{2\pi i\,(q_2 q_3)\,(q_2' q_3')/4}|q_2' q_3'\rangle$$

QUANTUM FOURIER TRANSFORM

An operation to move registers into phases and back

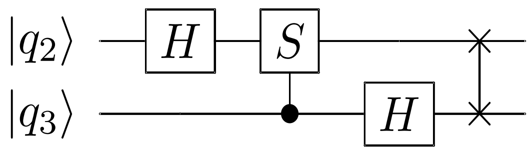
- The formal definition of the QFT is:

$$|q_0 q_1 q_2 \cdots\rangle \overset{\text{QFT}}{\mapsto} \sum_{q_0' q_1' q_2' \cdots} e^{2\pi i\,(q_0 q_1 q_2 \cdots)\,(q_0' q_1' q_2' \cdots)/2^N}\, |q_0' q_1' q_2' \cdots\rangle$$

- Intuition:

$$|j\rangle \leftrightarrow \sum_{j'} e^{2\pi i\, j\, j'/2^n} |j'\rangle$$

- Implementation for two qubits:



$$\begin{aligned} & e^{2\pi i\,(q_2 q_3)\,0/4}|0\rangle \\ + \; & e^{2\pi i\,(q_2 q_3)\,1/4}|1\rangle \\ + \; & e^{2\pi i\,(q_2 q_3)\,2/4}|2\rangle \\ + \; & e^{2\pi i\,(q_2 q_3)\,3/4}|3\rangle \end{aligned}$$

- Verification:

$$\mathrm{SWAP}\,(I \otimes H)\,CS\,(H \otimes I)\,|q_2 q_3\rangle \propto \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \end{array} |q_2 q_3\rangle = \sum_{q_2' q_3'} e^{2\pi i\,(q_2 q_3)\,(q_2' q_3')/4} |q_2' q_3'\rangle$$

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- We are looking for a quantum circuit to implement the following unitary operation

$$|q_0 q_1\rangle |q_2 q_3\rangle \qquad \mapsto \qquad |q_0 q_1\rangle |q_0 q_1 + q_2 q_3\rangle$$

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- We are looking for a quantum circuit to implement the following unitary operation

$$|q_0q_1\rangle|q_2q_3\rangle \qquad \mapsto \qquad |q_0q_1\rangle|q_0q_1+q_2q_3\rangle$$

- Such a modulo-4 adder would also work on superpositions

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- We are looking for a quantum circuit to implement the following unitary operation

$$|q_0 q_1\rangle|q_2 q_3\rangle \qquad \mapsto \qquad |q_0 q_1\rangle|q_0 q_1 + q_2 q_3\rangle$$

- Such a modulo-4 adder would also work on superpositions
- Examples:

JÜLICH Forschungszentrum

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- We are looking for a quantum circuit to implement the following unitary operation

$$|q_0 q_1\rangle |q_2 q_3\rangle \quad \mapsto \quad |q_0 q_1\rangle |q_0 q_1 + q_2 q_3\rangle$$

- Such a modulo-4 adder would also work on superpositions
- Examples:

$$\begin{aligned}
|2\rangle\,|1\rangle \quad &\mapsto \quad |2\rangle\,|3\rangle\,, \\
|2\rangle\,\frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad &\mapsto \quad |2\rangle\,\frac{|2\rangle + |3\rangle}{\sqrt{2}}, \\
|2\rangle\,\frac{|0\rangle + |1\rangle + |2\rangle}{\sqrt{3}} \quad &\mapsto \quad |2\rangle\,\frac{|2\rangle + |3\rangle + |0\rangle}{\sqrt{3}}, \\
\frac{|0\rangle + |1\rangle}{\sqrt{2}}\,\frac{|2\rangle + |3\rangle + |0\rangle}{\sqrt{3}} \quad &\mapsto \quad ?
\end{aligned}$$

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

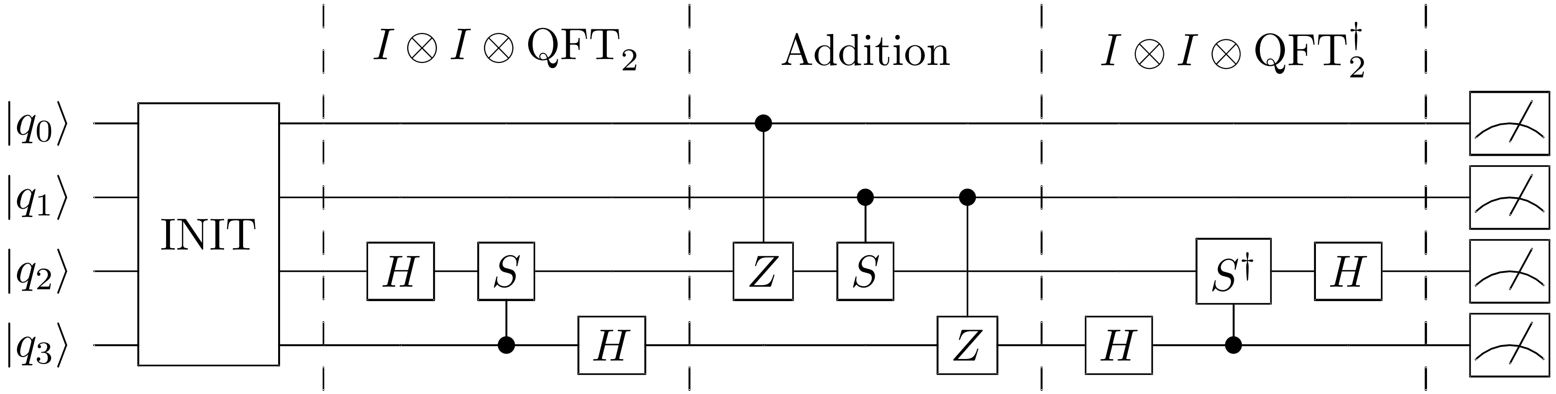
  → This is automatically modulo 4

QUANTUM ADDER

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

A nice application of the QFT

- Idea: → The QFT converts between registers and phases
  → By using bitwise phase shifts, we can implement the addition in the exponent
  → This is automatically modulo 4
- A circuit to realize this idea is



Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

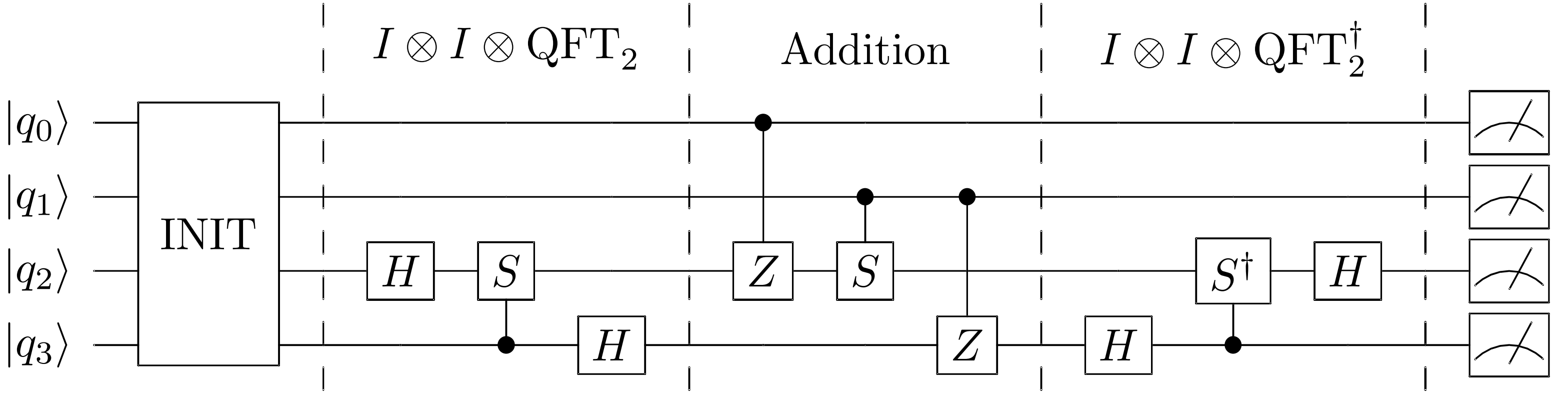
QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

  → This is automatically modulo 4
- A circuit to realize this idea is



- Intuition:

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

QUANTUM ADDER

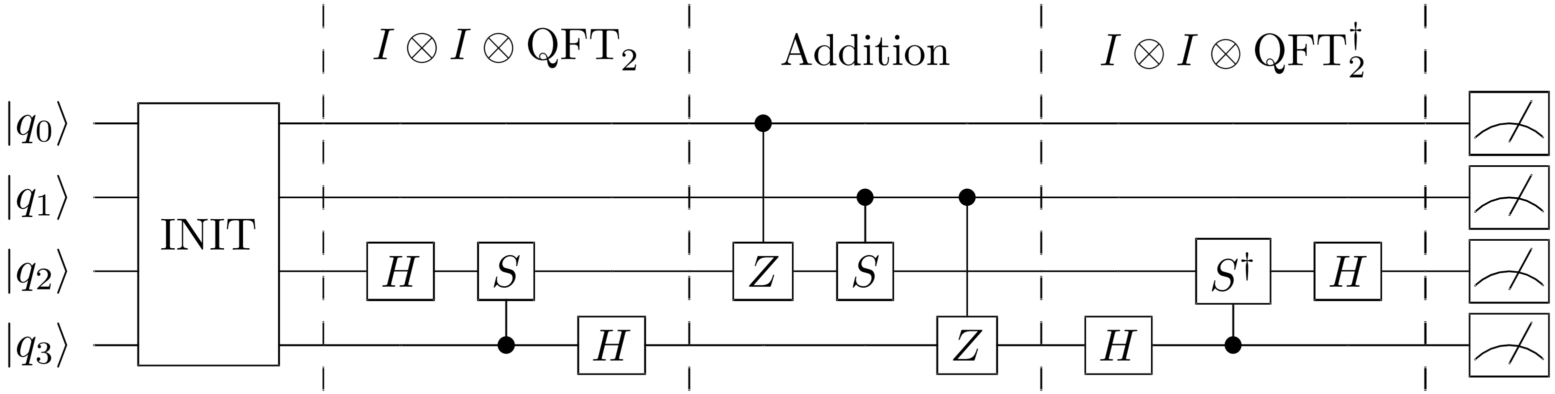
A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

  → This is automatically modulo 4

- A circuit to realize this idea is



- Intuition:

$|j\rangle|k\rangle$

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

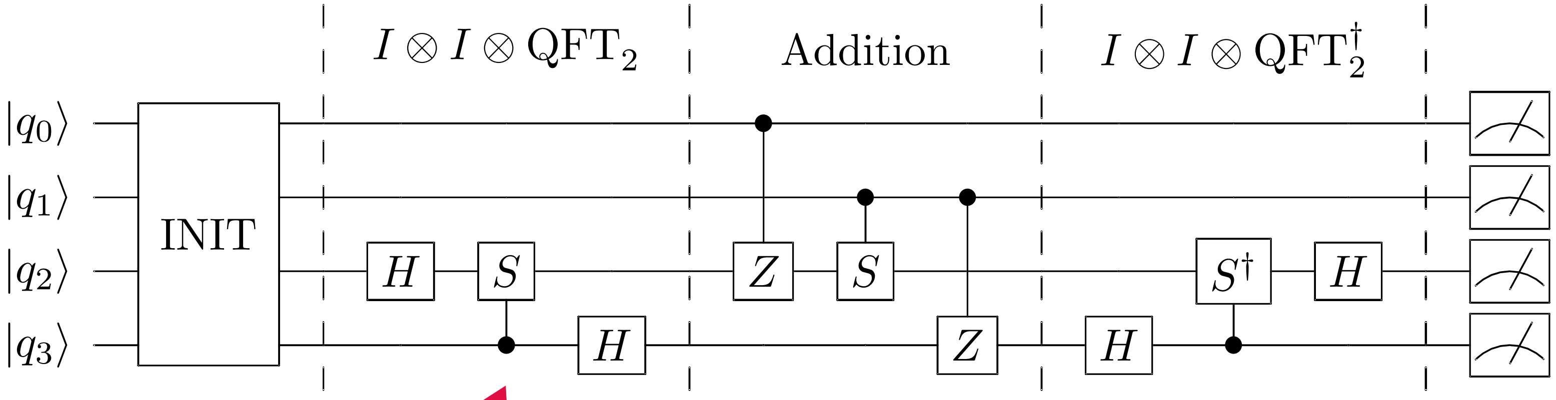
QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

  → This is automatically modulo 4
- A circuit to realize this idea is



- Intuition:

$$|j\rangle|k\rangle \mapsto |j\rangle \left(\sum_{k'} e^{2\pi i\, k\, k'/2^n} |k'\rangle \right)$$

JÜLICH Forschungszentrum

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

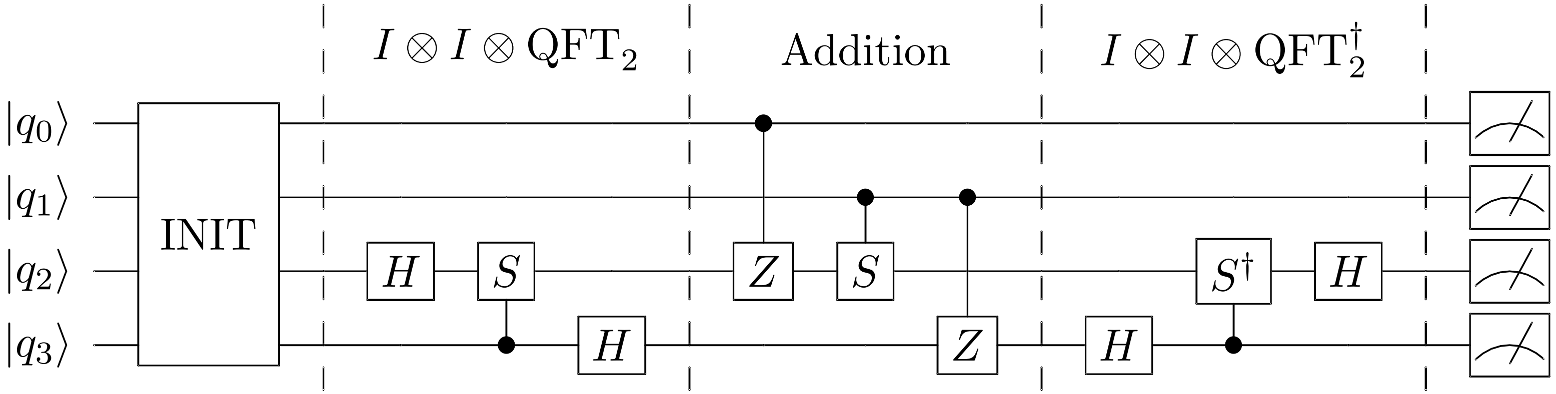
QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

  → This is automatically modulo 4
- A circuit to realize this idea is



- Intuition:

$$|j\rangle|k\rangle \mapsto |j\rangle \left(\sum_{k'} e^{2\pi i\, k\, k'/2^n} |k'\rangle \right) = \sum_{k'} e^{2\pi i\, k\, k'/2^n} |j\rangle|k'\rangle$$

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

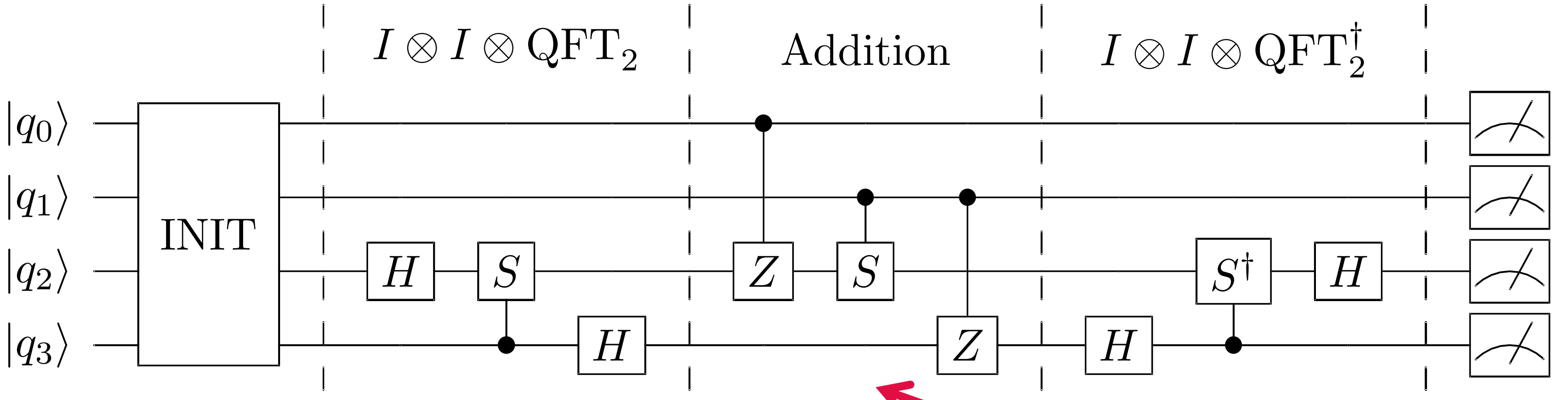
QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

  → This is automatically modulo 4
- A circuit to realize this idea is



- Intuition:

$$|j\rangle|k\rangle \mapsto |j\rangle \left(\sum_{k'} e^{2\pi i\, k\, k'/2^n} |k'\rangle \right) = \sum_{k'} e^{2\pi i\, k\, k'/2^n} |j\rangle|k'\rangle \mapsto \sum_{k'} e^{2\pi i\, (j+k)\, k'/2^n} |j\rangle|k'\rangle$$

Draper, arXiv:quant-ph/0008033 (2000)
https://arxiv.org/pdf/2201.02051.pdf

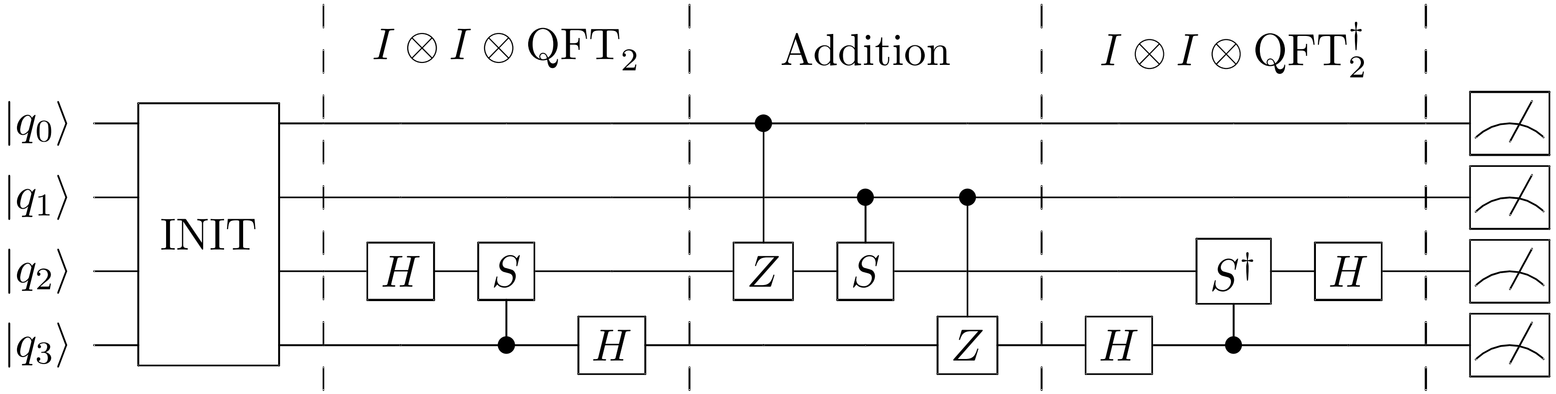
QUANTUM ADDER

A nice application of the QFT

- Idea: → The QFT converts between registers and phases

  → By using bitwise phase shifts, we can implement the addition in the exponent

  → This is automatically modulo 4
- A circuit to realize this idea is



- Intuition:

$$|j\rangle|k\rangle \mapsto |j\rangle\left(\sum_{k'} e^{2\pi i\, k\, k'/2^n}|k'\rangle\right) = \sum_{k'} e^{2\pi i\, k\, k'/2^n}|j\rangle|k'\rangle \mapsto \sum_{k'} e^{2\pi i\,(j+k)\,k'/2^n}|j\rangle|k'\rangle \mapsto |j\rangle|j+k\rangle$$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j$$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j$$

“QUBO” (q = 0,1)

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \qquad \Leftrightarrow \qquad E(s_0, \ldots, s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

"QUBO" (q = 0,1)

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \quad \Leftrightarrow \quad E(s_0, \ldots, s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

“QUBO” (q = 0,1)

“Ising” (s = +1,-1)

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0,\ldots,q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \qquad \Leftrightarrow \qquad E(s_0,\ldots,s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

"QUBO" (q = 0,1) **"Ising" (s = +1,-1)**

- Combinatorial optimization problem

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \qquad \Leftrightarrow \qquad E(s_0, \ldots, s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

“QUBO” (q = 0,1) **“Ising” (s = +1,-1)**

- Combinatorial optimization problem

  → discrete optimization is hard!

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \qquad \Leftrightarrow \qquad E(s_0, \ldots, s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

"QUBO" (q = 0,1) **"Ising" (s = +1,-1)**

- Combinatorial optimization problem

  → discrete optimization is hard!

- Idea: With a quantum circuit, we can create a superposition

$$|\psi\rangle = \psi_0 |0 \cdots 00\rangle + \cdots + \psi_{q_0^* \cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle + \cdots + \psi_{2^n-1} |1 \cdots 11\rangle$$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0,\ldots,q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \qquad \Leftrightarrow \qquad E(s_0,\ldots,s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

"QUBO" (q = 0,1) "Ising" (s = +1,-1)

- Combinatorial optimization problem

→ discrete optimization is hard!

- Idea: With a quantum circuit, we can create a superposition

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \underbrace{\boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^*\cdots q_{n-1}^*\rangle}}_{\text{Solution to the problem}} + \cdots + \psi_{2^n-1}|1\cdots 11\rangle$$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Say we want to find the minimum of the function

$$E(q_0, \ldots, q_{n-1}) = \sum_{i,j} q_i Q_{ij} q_j \qquad \Leftrightarrow \qquad E(s_0, \ldots, s_{n-1}) = \sum h_i s_i + \sum_{i<j} J_{ij} s_i s_j$$

"QUBO" (q = 0,1)

"Ising" (s = +1,-1)

- Combinatorial optimization problem

  → discrete optimization is hard!

- Idea: With a quantum circuit, we can create a superposition

$$|\psi\rangle = \psi_0 |0 \cdots 00\rangle + \cdots + \boxed{\psi_{q_0^* \cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1 \cdots 11\rangle$$

Solution to the problem

- Goal: Enhance this term!

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

- Make the quantum circuit dependent on real parameters $(\beta_1, \ldots, \beta_p, \gamma_1 \ \ldots, \gamma_p)$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

- Make the quantum circuit dependent on real parameters $(\beta_1, \ldots, \beta_p, \gamma_1 \ \ldots, \gamma_p)$
- Hope: these 2p parameters are easier searchable than the original discrete variables

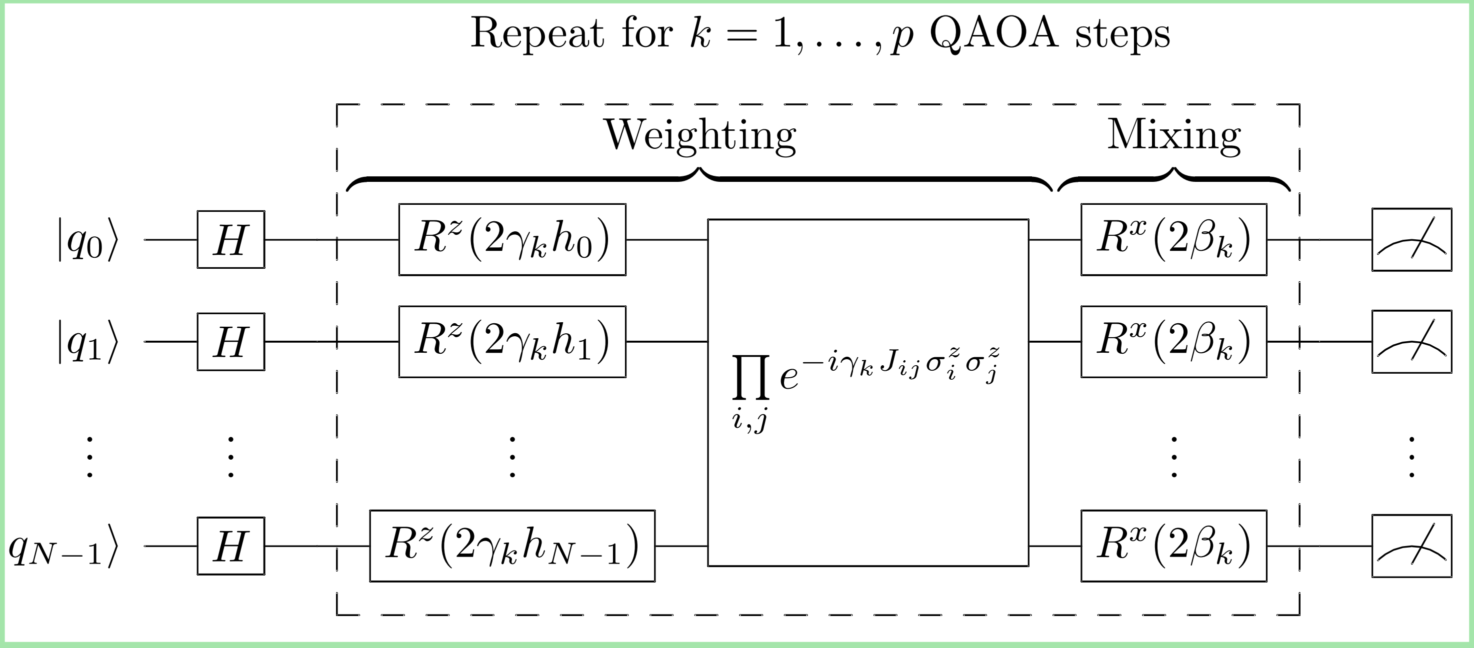
QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

- Make the quantum circuit dependent on real parameters $(\beta_1, \ldots, \beta_p, \gamma_1 \ \ldots, \gamma_p)$
- Hope: these 2p parameters are easier searchable than the original discrete variables



QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

- Look at p=1 QAOA step

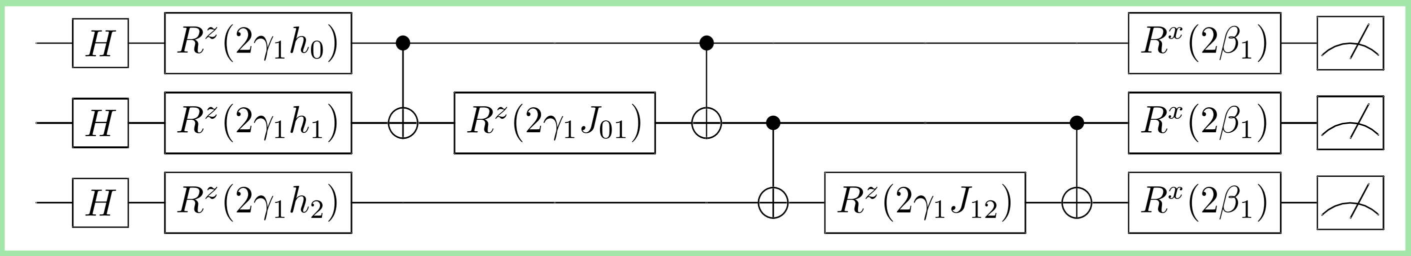
QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} |1\cdots 11\rangle$$

- Look at p=1 QAOA step



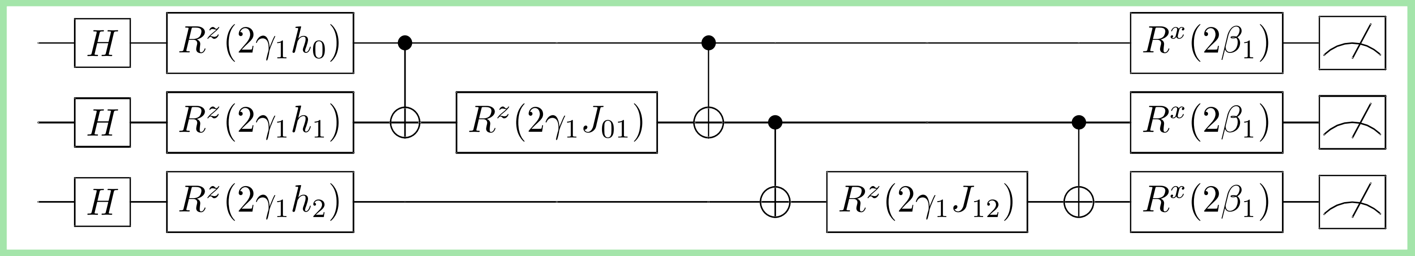
QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

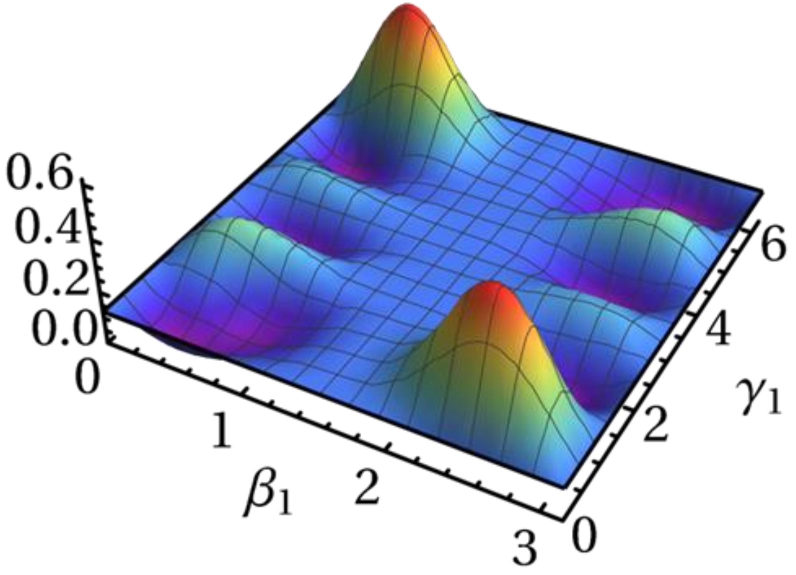
- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 \, |0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} \, |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} \, |1\cdots 11\rangle$$

- Look at p=1 QAOA step



Success probability $|\psi_{q_0^*\cdots q_{n-1}^*}|^2$



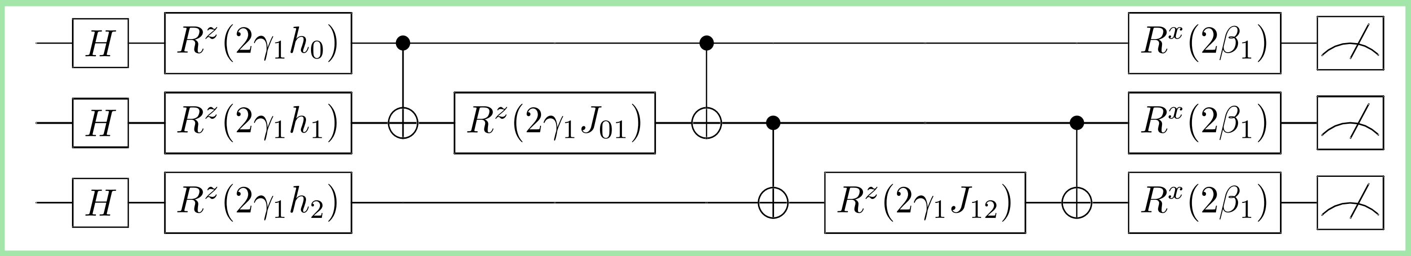
QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

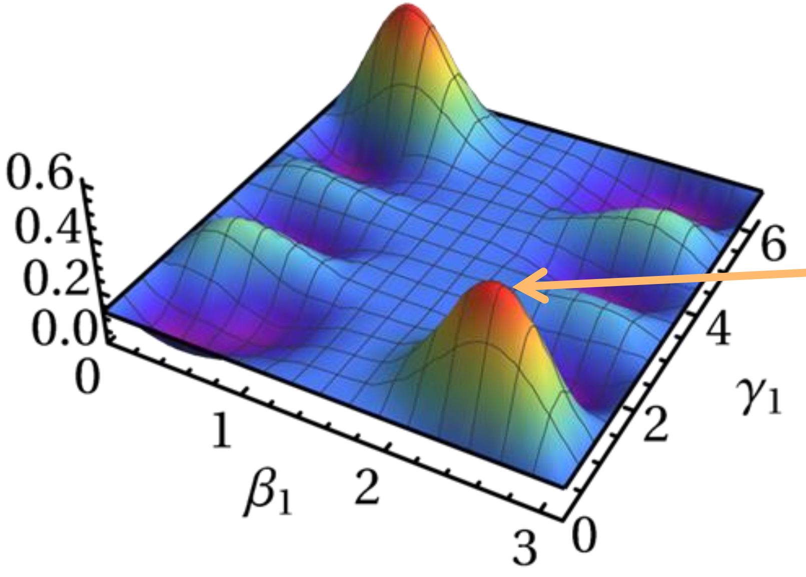
- Goal: Enhance this term!

$$|\psi\rangle = \psi_0 \, |0\cdots00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*} \, |q_0^* \cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1} \, |1\cdots11\rangle$$

- Look at p=1 QAOA step



Success probability $|\psi_{q_0^*\cdots q_{n-1}^*}|^2$



There are regions where the success probability is enhanced! How to find them?

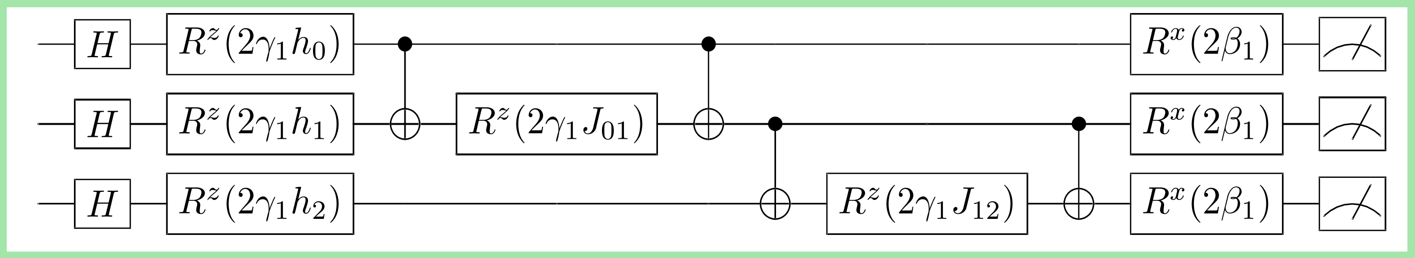
QUANTUM APPROXIMATE OPTIMIZATION ALGORITHM

A very brief overview of the QAOA

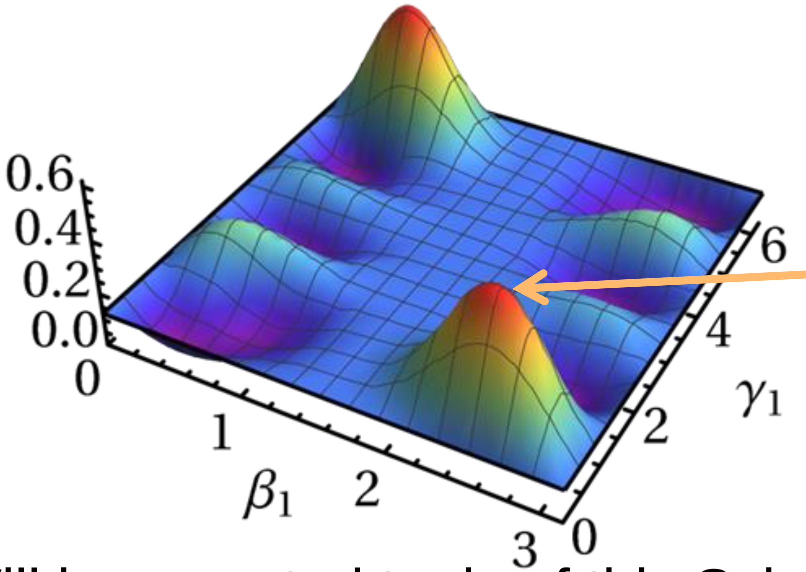
- Goal: Enhance this term!

$$|\psi\rangle = \psi_0\,|0\cdots 00\rangle + \cdots + \boxed{\psi_{q_0^*\cdots q_{n-1}^*}\,|q_0^*\cdots q_{n-1}^*\rangle} + \cdots + \psi_{2^n-1}\,|1\cdots 11\rangle$$

- Look at p=1 QAOA step



Success probability $|\psi_{q_0^*\cdots q_{n-1}^*}|^2$



There are regions where the success probability is enhanced! How to find them?

- Will be a central topic of this School ☺

THANK YOU FOR YOUR ATTENTION

- Further information:
  - **Programming Quantum Computers:** https://arxiv.org/pdf/2201.02051.pdf
  - **JUQCS:** Video https://youtu.be/JUSEYCkh2ac
  - **JUQCS:** Paper https://doi.org/10.1016/j.cpc.2022.108411