Introduction to Quantum Information and Quantum Machine Learning

Project - class 5

Dr Gustaw Szawioła, docent PUT
D. Sc. Eng. Przemysław Głowacki

1. Harrow-Hassidim-Lloyd (HHL) Algorithm

Formulation of the problem and the concept of its solution: Solving linear equations.

- System of linear equations
- $\hat{A}\,\vec{x} = \vec{b}$

  Matrix $\hat{A} = \hat{A}^{\dagger}$ is a square Hermitian matrix that has an inverse matrix $\hat{A}^{-1}$
- The dimension of the matrix $\hat{A}$: $N_b \times N_b$
- Dimensions of vectors $\vec{x}, \vec{b}$: $N_b = 2^{n_b}$
- The number of qubits needed to solve the problem : n_b

- Problem example
- $\hat{A} = \begin{pmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{pmatrix}$
- $\vec{b} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$
- A classic solution
- $\vec{x} = \begin{pmatrix} \frac{3}{8} \\ \frac{9}{8} \end{pmatrix} = \frac{3}{8}\begin{pmatrix} 1 \\ 3 \end{pmatrix}$
- $N_b = 2^1 = 2, n_b = 1$

The problem solving concept

- From the spectral theorem we can write

$$\hat{A} = \sum_{i=0}^{N_b} \lambda_i |u_i\rangle\langle u_i|$$

where

$\hat{A}\,|u_i\rangle = \lambda_i |u_i\rangle\,, \quad \langle u_i|u_j\rangle = \delta_{i,j}$

- Inverse matrix $\hat{A}^{-1}$ can be presented in the form according to the spectral theorem

$$\hat{A}^{-1} = \sum_{i=0}^{N_b-1} \lambda_i^{-1} |u_i\rangle\langle u_i|$$

- Classical vector encoding $\vec{b}$ in a quantum vector $|b\rangle$

$$|b\rangle = \sum_{i=0}^{N_b-1} b_i |u_i\rangle$$

- The solution can be encoded in the form

$$|x\rangle = \hat{A}^{-1}|b\rangle = \sum_{i=0}^{N_b-1} \lambda_i^{-1} b_i |u_i\rangle$$

It's a matter of coding:

- Binary Encoding in the Base-State (Base-State Binary Encoding) - Base-state encoding converts classical information such as numbers into quantum information in the form of basis states.

- The example

$$x = 2 \xrightarrow{binary} 10 \xrightarrow{quantum\ state} |10\rangle$$

- Amplitude encoding - Amplitude encoding encodes information in the form of basis vector coefficients.

- The example

$$\vec{v} = \binom{v_0}{v_1} \xrightarrow{quantum\ state} |v\rangle = v_0|0\rangle + v_1|1\rangle$$

It's a matter of coding (contuined):

- Hamiltonian coding - one type of Hamiltonian coding involves coding a matrix as a Hamiltonian in a unitary transformation using the exponential representation theorem of the unitary operator.

- The example

  Hermitian matrix $\hat{A} \xrightarrow{quantum\ gate}$

  $\xrightarrow{quantum\ gate}$ Unitary matrix $\hat{U} = e^{i\hat{A}\,t}$,

  where $\hat{A}$ plays the role of a Hamiltonian, whereby

- $\hat{A}\,|u_i\rangle = \lambda_i|u_i\rangle$

- $\hat{U}\,|u_i\rangle = e^{i\,\hat{A}\,t}|u_i\rangle = e^{i\,\lambda_i t}|u_i\rangle$

The concept of an algorithm for solving equations

1. **State coding $|b\rangle$ - amplitude version of coding in register b with the number of n_b- qubits**

   First, $|b\rangle$ should be expressed in the basis $\{|u_i\rangle\}$ of eigenvectors $\hat{A}$ and then transformed into the computational basis $\{|x\rangle\}$ of register b.

2. **Encoding of matrix $\widehat{A}$ in unitary operation (quantum gate) $\widehat{U} = e^{i\widehat{A}\,t}$ - the Hamiltonian version of encoding**

   For the operation $\widehat{U} = e^{i\widehat{A}\,t}$ we create controlled quantum gates c$\widehat{U}$. Using the phase estimation operation, it is possible to determine the eigenvalues of $e^{i\widehat{A}\,t}$ and therefore also $\widehat{A}$ simultaneously. As control qubits in the QPE operation, we will use a register composed of n-qubits (auxiliary in QPE), which we will call **clock**. The eigenvalues $\widehat{A}$, i.e. the phase $\widehat{U}$ will be stored in the clock register, encoded in the **base states**.

The concept of an algorithm for solving equations

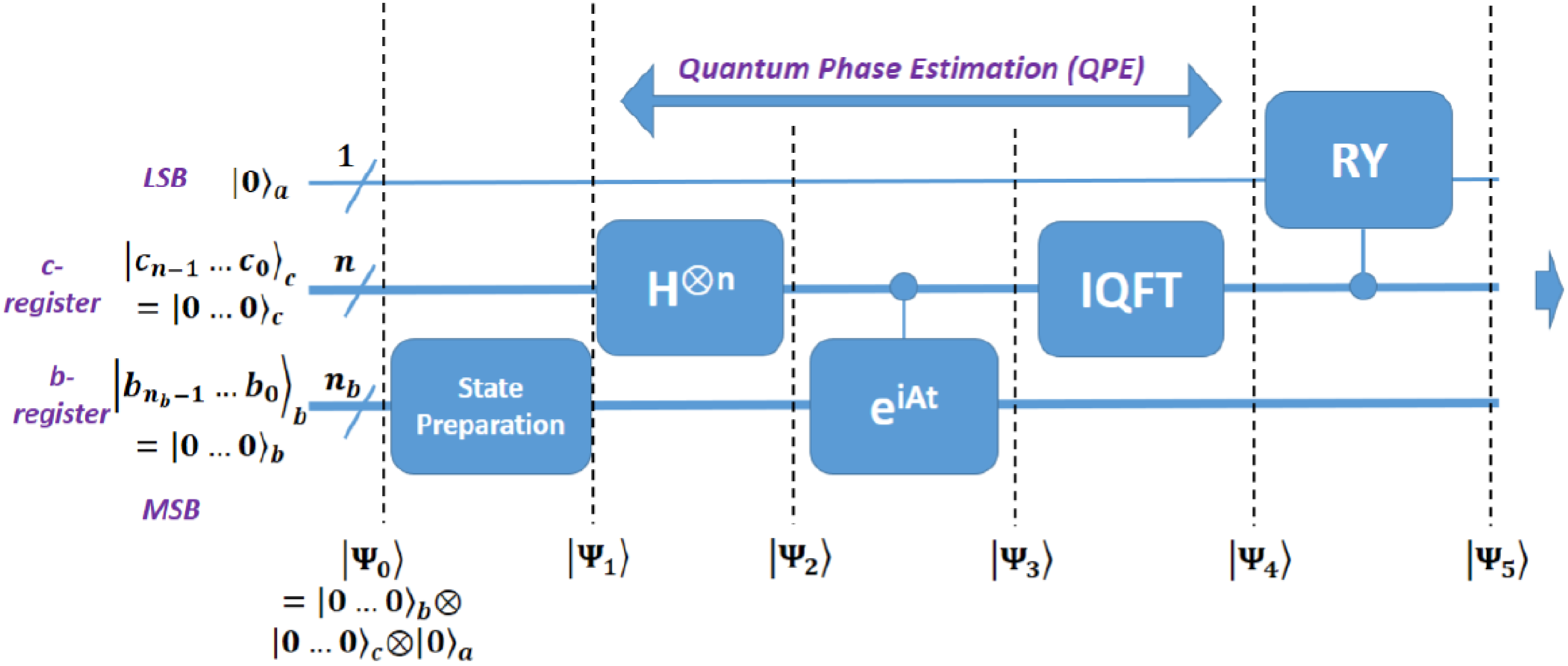
3. **Transformation $\boldsymbol{\lambda_i \to \lambda_i^{-1}}$**

   The above transformation is implemented as controlled quantum gates $cc\hat{R}_y$, with the controlling qubits being the clock register qubits storing the eigenvalues λ_i. The target qubit is an additional one ancilla qubit. After the $cc\hat{R}_y$ operation, a measurement is made, the result of which "1" on the ancilla qubit informs that the correct result (quantum solution) $|x\rangle$ has been stored in the b register.

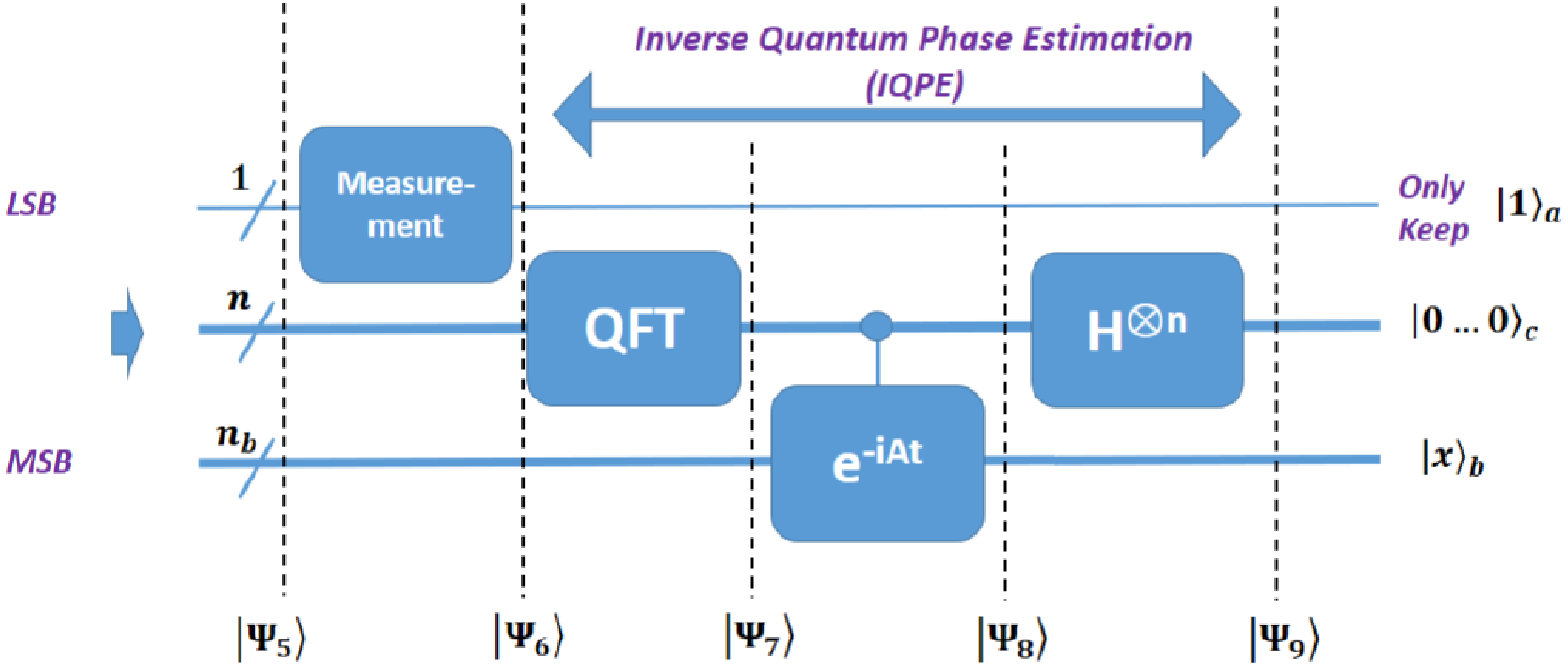
4. **Reverse calculations ("decomputation")**

   Since the register b with the result $|x\rangle$ is in the convolved state with the entangled register, an operation must be performed to separate the states of the $|x\rangle$ register and the clock register. This is accomplished by the IQEP operation inverse to QPE.

The general HHL quantum circuit



The general HHL quantum circuit



2. Coding states and elementary operations - example analysis

Encoding the state of register b

- $\widehat{A}\,\vec{x} = \vec{b}$
- Eigenvalues and eigenvectors $\widehat{A}$
- $\widehat{A} = \begin{pmatrix} 1 & -\frac{1}{3} \\ -\frac{1}{3} & 1 \end{pmatrix}$

$u_0 = |u_0| e^{i\,\delta}$

$\delta = \pi$ $\qquad$ $\delta = 0$

- $\lambda_0 = \frac{2}{3} \longrightarrow |u_0\rangle = -\frac{1}{\sqrt{2}}\binom{1}{1}$ $\qquad$ $|u_0\rangle = \frac{1}{\sqrt{2}}\binom{1}{1}$
- $\lambda_1 = \frac{4}{3} \longrightarrow |u_1\rangle = \frac{1}{\sqrt{2}}\binom{-1}{1}$ $\qquad$ $|u_0\rangle = \frac{1}{\sqrt{2}}\binom{1}{-1}$

- Encoding $\vec{b}$ in the database $\{|0\rangle, |1\rangle\}$
- $\vec{b} = \binom{0}{1} \longrightarrow |b\rangle = \hat{X}|0\rangle = |1\rangle = \binom{0}{1}$
- Encoding $\vec{b}$ in the database $\{|u_0\rangle, |u_1\rangle\}$
- $|b\rangle = b_0|u_0\rangle + b_1|u_1\rangle =$

$$= -\frac{1}{\sqrt{2}}|u_0\rangle + \frac{1}{\sqrt{2}}|u_1\rangle$$

Binary encoding in the base states of the eigenvalues $\{\lambda_0, \lambda_1\}$ in the clock register

- Since the clock register is related to the QPE operation and stores the eigenvalues $\{\lambda_0, \lambda_1\}$, we want to represent these values as binary-encoded integers in the computational base of the n_c qubit clock register.

$$\frac{\lambda_1}{\lambda_0} = \frac{\frac{4}{3}}{\frac{2}{3}} = 2$$

- We transform $\{\lambda_0, \lambda_1\} \longrightarrow \{\tilde{\lambda}_0, \tilde{\lambda}_1\}$

$$2\,\pi\,\frac{\tilde{\lambda}_i}{N_c} = \lambda_i\, t$$

- We want to keep the proportions, i.e. $\frac{\lambda_1}{\lambda_0} = \frac{\tilde{\lambda}_1}{\tilde{\lambda}_0}$
- but we choose as small integers as possible $\tilde{\lambda}_0 = 1, \tilde{\lambda}_1 = 2$.
- Nc=2 bits because we will represent the decimal number 2 using binary 10

Binary encoding in the base states of the eigenvalues $\{\lambda_0, \lambda_1\}$ in the clock register (contuined)...

- We note that two bits are enough to write numbers in binary $\tilde{\lambda}_0 = 1, \tilde{\lambda}_1 = 2$ therefore in binary notation $\tilde{\lambda}_0 = 01, \tilde{\lambda}_1 = 10$ and from here
- $|\tilde{\lambda}_0\rangle = |1\rangle = |01\rangle$
- $|\tilde{\lambda}_1\rangle = |2\rangle = |10\rangle$.
- Number $N_c = 2^{n_c} = 4$
- Equality $2\,\pi\,\frac{\tilde{\lambda}_i}{N_c} = \lambda_i\, t$ can be fulfilled with

$$t = \frac{3\,\pi}{4}$$

Encoding $\hat{U}$ in the database $\{|u_0\rangle, |u_1\rangle\}$

- Transformation matrix to the basis $\{|u_0\rangle, |u_1\rangle\}$

$$\hat{V} = (|u_0\rangle, |u_1\rangle) = \begin{pmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}$$

- Transformation of matrix $\hat{A}$ into the basis $\{|u_0\rangle, |u_1\rangle\}$

$$\hat{A}_u = \hat{V}^\dagger \hat{A} \hat{V} = \begin{pmatrix} \lambda_0 & 0 \\ 0 & \lambda_1 \end{pmatrix} = \begin{pmatrix} \frac{2}{3} & 0 \\ 0 & \frac{4}{3} \end{pmatrix}$$

- Matrix $\hat{U}$ in the basis $\{|u_0\rangle, |u_1\rangle\}$

$$\hat{U}_u = e^{i\hat{A}t} = \begin{pmatrix} e^{i\lambda_0 t} & 0 \\ 0 & e^{i\lambda_1 t} \end{pmatrix} =$$

$$= \begin{pmatrix} e^{i2\pi\frac{\tilde{\lambda}_0}{N_c}} & 0 \\ 0 & e^{i2\pi\frac{\tilde{\lambda}_1}{N_c}} \end{pmatrix} =$$

$$= \begin{pmatrix} e^{i\frac{\pi}{2}} & 0 \\ 0 & e^{i\pi} \end{pmatrix} = \begin{pmatrix} i & 0 \\ 0 & -1 \end{pmatrix}$$

Encoding $\hat{U}^2$ in the database $\{|u_0\rangle, |u_1\rangle\}$

- Since two qubits are enough in the clock register to represent the scaled eigenvalues $\{\tilde{\lambda}_0 = 1, \tilde{\lambda}_1 = 2\}$, the following operations are performed in the QPE operation implemented on register b:
$$\hat{U}^{2^0} = \hat{U}$$
and
$$\hat{U}^{2^1} = \hat{U}^2$$
controlled by clock register qubits

- Matrix $\hat{U}^2$ in the basis $\{|u_0\rangle, |u_1\rangle\}$
$$\hat{U}_u^2 = \begin{pmatrix} i & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} i & 0 \\ 0 & -1 \end{pmatrix} =$$
$$= \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix}$$

Encoding $\hat{U}$ and $\hat{U}^2$ in the database $\{|0\rangle, |1\rangle\}$

- Transformation $\hat{U}_u$ to base $\{|0\rangle, |1\rangle\}$

$$\hat{U} = \hat{V}\,\hat{U}_u\hat{V}^{\dagger} =$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} i & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} =$$

$$= \frac{1}{2}\begin{pmatrix} -1+i & 1+i \\ 1+i & -1+i \end{pmatrix}$$

- Transformation $\hat{U}_u^2$ to base $\{|0\rangle, |1\rangle\}$

$$\hat{U}^2 = \hat{V}\,\hat{U}_u^2\hat{V}^{\dagger} =$$

$$= \begin{pmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} =$$

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

Implementation $\widehat{U}$ and $\widehat{U}^2$ using Qiskit

- In Qiskit a four-parameter gate is available

$$\hat{u}(\theta,\phi,\lambda,\gamma) = = \begin{pmatrix} e^{i\gamma}\cos\frac{\theta}{2} & -e^{i(\gamma+\lambda)}\sin\frac{\theta}{2} \\ e^{i(\gamma+\phi)}\sin\frac{\theta}{2} & e^{i(\gamma+\phi+\lambda)}\cos\frac{\theta}{2} \end{pmatrix}$$

- Implementation

$$\widehat{U} = \hat{u}\left(\theta = \frac{\pi}{2}, \phi = -\frac{\pi}{2}, \lambda = \frac{\pi}{2}, \gamma = \frac{3\pi}{4}\right)$$

- Implementation

$$\widehat{U}^2 = \hat{u}(\theta = \pi, \phi = \pi, \lambda = 0, \gamma = 0)$$

The example for $\widehat{U}$:

$$\widehat{U} = \frac{1}{2}\begin{pmatrix} -1+i & 1+i \\ 1+i & -1+i \end{pmatrix}$$

$$\begin{cases} e^{i\gamma}\cos\frac{\theta}{2} = -\frac{1}{2}+\frac{1}{2}i \\ -e^{i(\gamma+\lambda)}\sin\frac{\theta}{2} = \frac{1}{2}+\frac{1}{2}i \\ e^{i(\gamma+\phi)}\sin\frac{\theta}{2} = \frac{1}{2}+\frac{1}{2}i \\ e^{i(\gamma+\phi+\lambda)}\cos\frac{\theta}{2} = -\frac{1}{2}+\frac{1}{2}i \end{cases} \Rightarrow \widehat{U} = \hat{u}\left(\theta = \frac{\pi}{2}, \phi = -\frac{\pi}{2}, \lambda = \frac{\pi}{2}, \gamma = \frac{3\pi}{4}\right)$$

Implementation $\hat{U}$ and $\hat{U}^2$ using Qiskit

- Let's note that

$$\hat{U}^{-1} = \hat{U}^{\dagger} =$$

$$= \frac{1}{2}\begin{pmatrix} -1-i & 1-i \\ 1-i & -1-i \end{pmatrix} =$$

$$= \hat{u}\left(\theta = \frac{\pi}{2}, \phi = \frac{\pi}{2}, \lambda = -\frac{\pi}{2}, \gamma = -\frac{3\pi}{4}\right)$$

- Let's note that

$$\hat{U}^{-2} = \left(\hat{U}^2\right)^{-1} = \left(\hat{U}^2\right)^{\dagger}$$

- Therefore, in the analyzed example

$$\hat{U}^{-2} = \hat{U}^2 =$$

$$= \hat{u}\left(\theta = \frac{\pi}{2}, \phi = -\frac{\pi}{2}, \lambda = \frac{\pi}{2}, \gamma = \frac{3\pi}{4}\right)$$

Comment

- To apply the QPE operation, all gates $\hat{U}^1$, $\hat{U}^2$, $\hat{U}^{-1}$, $\hat{U}^{-2}$ should be implemented as gates $c\hat{U}^1$, $c\hat{U}^2$, $c\hat{U}^{-1}$, $c\hat{U}^{-2}$ controlled by the clock qubit.
- This can be done in Qiskit using gate $cu(\theta, \phi, \lambda, \gamma, q[i])$, where i denotes the number of the clock qubit controlling.

Eigenvalue inversion operation

- We can implement $\lambda_i \to \lambda_i^{-1}$ operations on one qubit by using the rotation operation:

$$\hat{R}_y(\theta) = \begin{pmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}$$

assuming that $\theta = 2\arcsin\left(\frac{1}{\tilde{\lambda}_j}\right)$

$$\sin\frac{\theta}{2} = \sin\left(\arcsin\left(\frac{1}{\tilde{\lambda}_j}\right)\right) = \frac{1}{\tilde{\lambda}_j}$$

- Action of $\hat{R}_y(\theta)$ operator on the auxiliary qubit (ancilla qubit)

$$\hat{R}_y(\theta)|0\rangle =$$

$$= \cos\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}|1\rangle =$$

$$= \sqrt{1 - \sin^2\frac{\theta}{2}|0\rangle + \sin\frac{\theta}{2}|1\rangle} =$$

$$= \sqrt{1 - \left(\frac{1}{\tilde{\lambda}_j}\right)^2}|0\rangle + \frac{1}{\tilde{\lambda}_j}|1\rangle$$

A controlled operation $\widehat{ccR_y}(\theta)$

- The clock register in the analyzed example is two-qubit, so the basis vectors of this register can be written as $|c_1 c_0\rangle$. Binary numbers of the form $c = c_1 c_0$ they encode eigenvalues $\tilde{\lambda}_j = c$ determined by the QPE operation so they will control the operations $\widehat{ccR_y}(\theta)$.
- We can implement such controlled operations by taking this

$$\theta(c) = \theta(c_1 c_0) = 2 \arcsin\left(\frac{1}{\tilde{\lambda}_j}\right)$$

- Therefore, angle rotations should be applied to rotations that reverse eigenvalues

$$\theta(c = 1) = \theta(01) = 2 \arcsin\left(\frac{1}{\tilde{\lambda}_0 = 1}\right) = \pi$$

$$\theta(c = 2) = \theta(10) = 2 \arcsin\left(\frac{1}{\tilde{\lambda}_0 = 2}\right) = \frac{\pi}{3}$$

- Summarizing

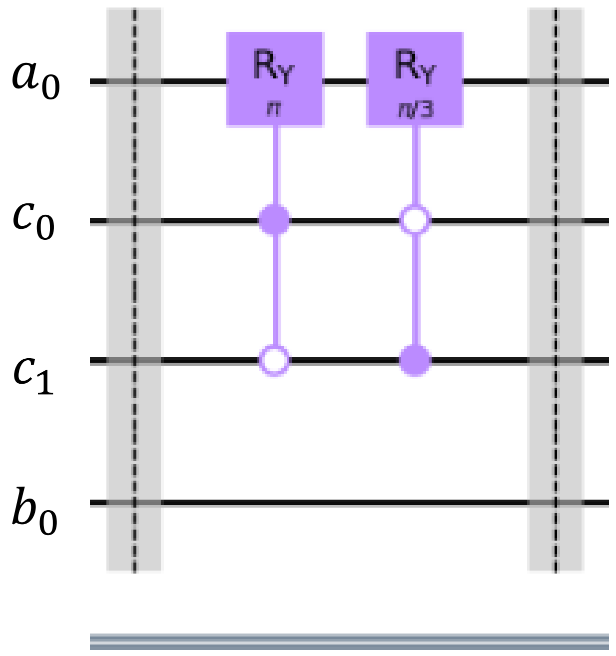
$$\theta(c) = \theta(c_1 c_0) = \frac{\pi}{3} c_1 + \pi c_0$$

A controlled operation $\widehat{ccR}_y(\theta)$

- Therefore, the operation $\lambda_i \to \lambda_i^{-1}$ implemented through controlled gates $\widehat{ccR}_y(\theta)$ can be determined by the truth table

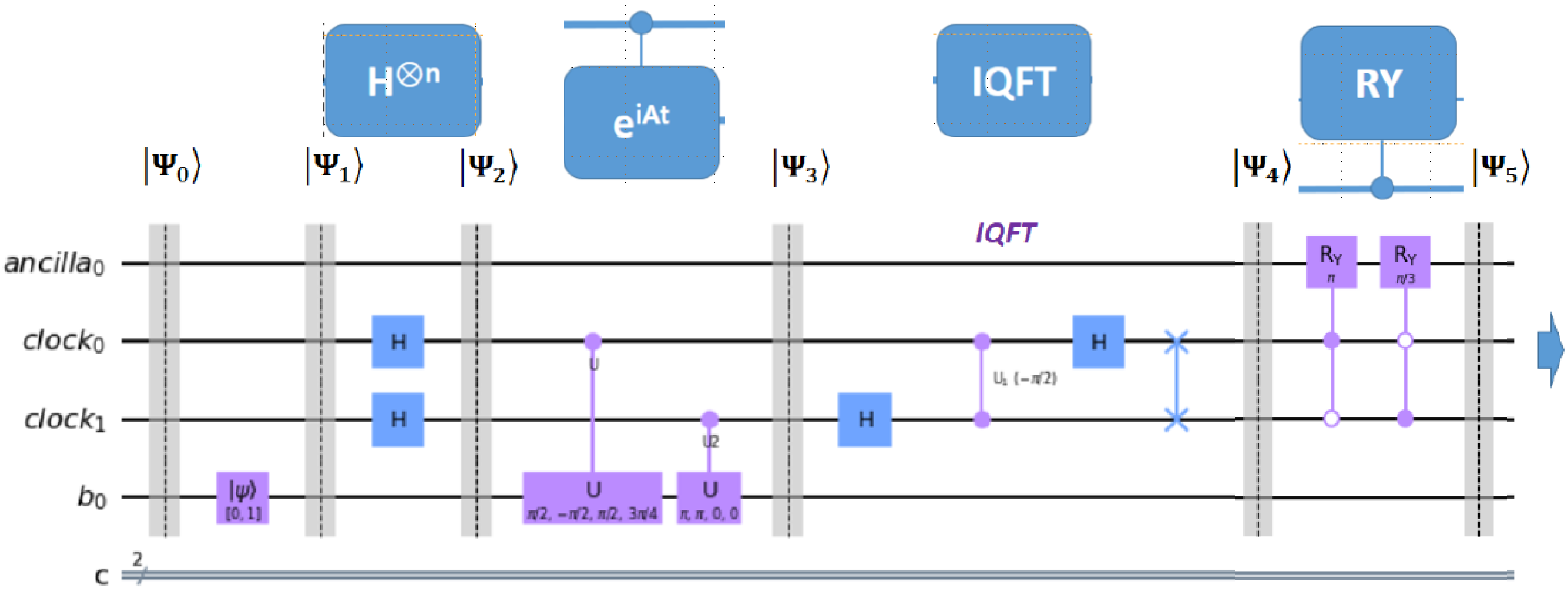
c_1	c_0	$\theta(c)=\theta(c_1c_0)$	Rotation on ancilla
0	0	0	0
0	1	π	Ry(π)
1	0	π/3	Ry(π/3)
1	1	0	0

- Quantum circuit $\widehat{ccR}_y(\theta)$

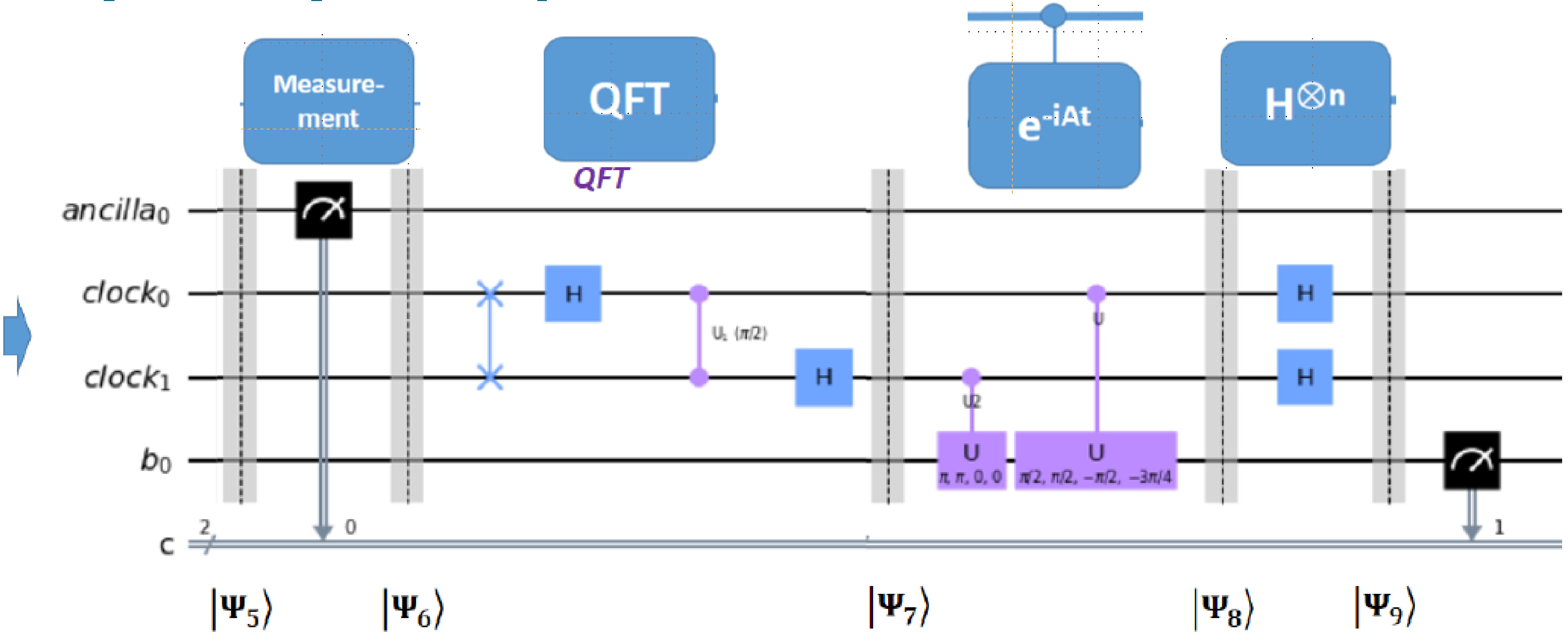


3. HHL implementation and result analysis - example

Computational part of the quantum HHL circuit - example



Computational part of the quantum HHL circuit - example



The general form of the final calculations

- The form of the result assuming the measurement result is "1" on the ancilla qubit

$$|\Psi_9\rangle =$$

$$= \frac{1}{\sqrt{\sum_{i=0}^{N_b-1}\left|\lambda_i^{-1} b_i\right|^2}} |x\rangle \otimes |00\rangle \otimes |1\rangle =$$

$$= |x\rangle \otimes |00\rangle \otimes |1\rangle$$

- Reminder:

$$|x\rangle = \hat{A}^{-1}|b\rangle = \sum_{i=0}^{N_b-1} \frac{b_i}{\lambda_i} |u_i\rangle$$

$$|b\rangle = \sum_{i=0}^{N_b-1} b_i |u_i\rangle$$

The general form of the final calculations

- The form of the result assuming the measurement result is "1" on the ancilla qubit in base $\{|u_0\rangle, |u_1\rangle\}$

$$|\Psi_9\rangle = \frac{2}{3}\sqrt{\frac{8}{5}}\left(-\frac{1}{\frac{2}{3}\sqrt{2}}|u_0\rangle + \frac{1}{\frac{4}{3}\sqrt{2}}|u_1\rangle\right)\otimes|00\rangle\otimes|1\rangle$$

- The form of the result assuming the measurement result is "1" on the ancilla qubit in base $\{|0\rangle, |1\rangle\}$

- $|\Psi_9\rangle = \frac{1}{2}\sqrt{\frac{2}{5}}\,(|0\rangle + 3|1\rangle)\otimes|00\rangle\otimes|1\rangle$

- Reminder:

$$|b\rangle = b_0|u_0\rangle + b_1|u_1\rangle = -\frac{1}{\sqrt{2}}|u_0\rangle + \frac{1}{\sqrt{2}}|u_1\rangle$$

$$|u_0\rangle = -\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$$

$$|u_1\rangle = -\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$$

- Therefore

$$|x\rangle = \frac{b_0}{\lambda_0}|u_i\rangle + \frac{b_1}{\lambda_1}|u_i\rangle = -\frac{1}{\frac{2}{3}\sqrt{2}}|u_0\rangle + \frac{1}{\frac{4}{3}\sqrt{2}}|u_1\rangle$$

Probability ratios for the result on the ancilla qubit "1"

- Result obtained on the simulator (previously, the measurement was performed on the auxiliary register with a probabilistically obtained result of "1") :
  [0. +0.j 0.316-0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j 0.949-0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j 0. +0.j]

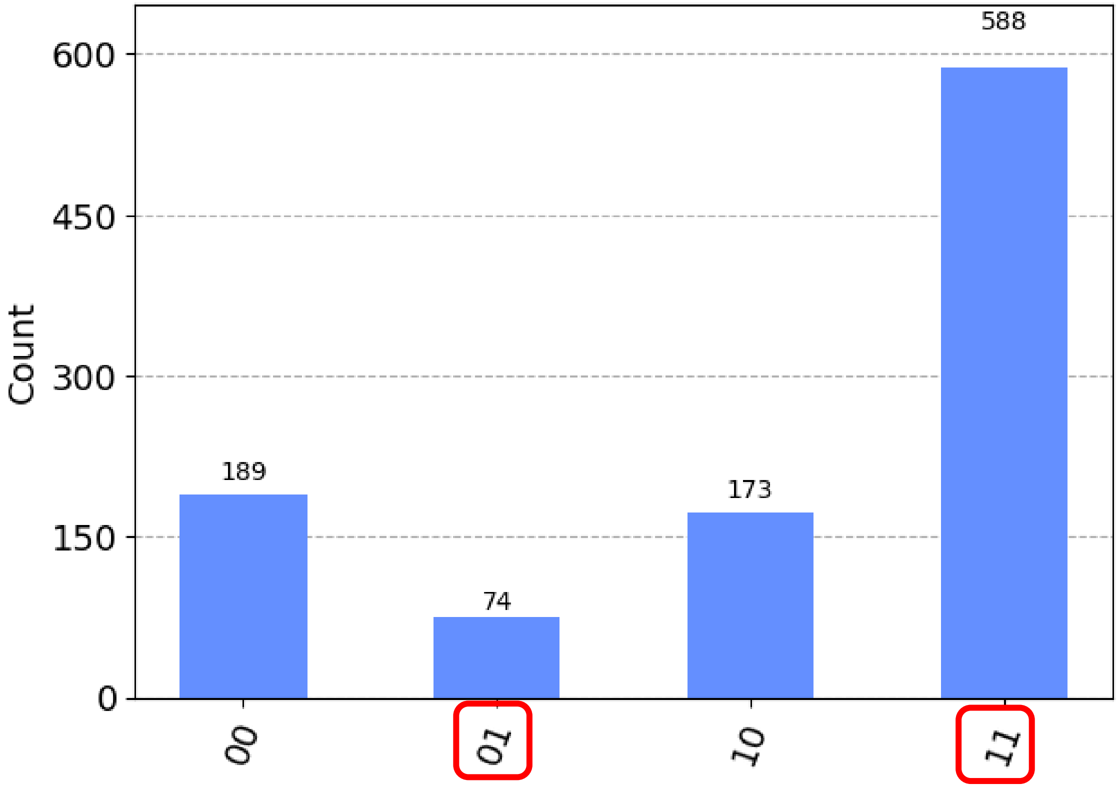
- Probability ratio:

$$\frac{p(01)}{p(11)} = \frac{(0.316)^2}{(0.949)^2} \approx \frac{1}{9}$$

- Counts ratio

$$\frac{n(01)}{n(11)} = \frac{74}{588} \approx \frac{1}{9}$$

- Histogram of counts after measurement on the auxiliary register and b



Comparison with the classic result

- The obtained quantum result has the form

$$|\Psi_9\rangle = C\,(x_0|0\rangle + x_1|1\rangle)\otimes|00\rangle\otimes|1\rangle$$

$$= \frac{1}{2}\sqrt{\frac{2}{5}}\,(|0\rangle + 3|1\rangle)\otimes|00\rangle\otimes|1\rangle$$

- Probability ratio:

$$\frac{p(0)}{p(1)} = \frac{|C\,x_0|^2}{|C\,x_1|^2} = \frac{\left(\frac{1}{2}\sqrt{\frac{2}{5}}\cdot 1\right)^2}{\left(\frac{1}{2}\sqrt{\frac{2}{5}}\cdot 3\right)^2} = \frac{1}{9} = \left(\frac{1}{3}\right)^2$$

- Classical result $\vec{x} = \begin{pmatrix} x_0^c \\ x_1^c \end{pmatrix} = \begin{pmatrix} \frac{3}{8} \\ \frac{9}{8} \end{pmatrix}$

- Comparison of classical and quantum result:

$$\text{quantumly} = \frac{x_0}{x_1} = \frac{x_0^c}{x_1^c} = \frac{3}{9} = \frac{1}{3}$$

4. Tasks
Coding states and elementary operations - example analysis

The report tasks:

1. Implement the HHL algorithm for the case:

$$\hat{A} = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \text{ and } \vec{b} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

2. Compare classical and quantum solutions $\hat{A}\,\vec{b} = \vec{x}$ i.e. calculate coordinate quotients for vectors $\vec{x}$ and $|x\rangle$ constituting the classical and quantum solutions, respectively.
3. Analyze the ratios of the probabilities of obtaining the base states in the final measurement on the b and auxiliary registers.

Use a template file: *Proj_5_2025_Q_2_2_1.ipynb*

TIP 1

Implementation $\hat{U}$ and $\hat{U}^2$ using Qiskit

- In Qiskit a four-parameter gate is available

$$\hat{u}(\theta,\phi,\lambda,\gamma) = $$

$$= \begin{pmatrix} e^{i\gamma}\cos\frac{\theta}{2} & -e^{i(\gamma+\lambda)}\sin\frac{\theta}{2} \\ e^{i(\gamma+\phi)}\sin\frac{\theta}{2} & e^{i(\gamma+\phi+\lambda)}\cos\frac{\theta}{2} \end{pmatrix}$$

For $\hat{U}$ use the angles listed below:

$\hat{U} = \hat{u}\left(\theta = \pi, \phi = 0, \lambda = \pi, \gamma = \frac{\pi}{2}\right)$

$\hat{U} = \hat{u}\left(\theta = \pi, \phi = \frac{\pi}{2}, \lambda = \frac{3\pi}{2}, \gamma = 0\right)$

- Implementation

$$\hat{U} = \hat{u}(\theta = \pi, \phi = 0, \lambda = -\pi/2, \gamma = 0)$$

- Implementation

$$\hat{U}^2 = \hat{u}(\theta = 0, \phi = 0, \lambda = 0, \gamma = \pi)$$

For $\widehat{U}^2$ use the angles listed below:

$\theta = 0$, $\gamma = \pi$, $\phi + \lambda = 0$

$\hat{U}^2 = \hat{u}(\theta = 0, \phi = 0, \lambda = 0, \gamma = \pi)$

$\hat{U}^2 = \hat{u}\left(\theta = 0, \phi = -\frac{\pi}{2}, \lambda = \frac{\pi}{2}, \gamma = \pi\right)$

$\hat{U}^2 = \hat{u}\left(\theta = 0, \phi = \frac{\pi}{2}, \lambda = -\frac{\pi}{2}, \gamma = \pi\right)$

TIP 2

Implementation $\hat{U}$ and $\hat{U}^2$ using Qiskit

- Let's note that

$$\hat{U}^{-1} = \hat{U}^{\dagger} =$$

$$= \hat{u}(\theta = \pi, \phi = 0, \lambda = -\pi, \gamma = 3/2\,\pi)$$

For $\hat{U}^{-1}$ use the angles listed below:

$\theta = \pi, \lambda - \phi = -\pi, \gamma = \frac{\pi}{2} - \lambda$ *or* $\gamma = \frac{3\pi}{2} - \phi$

$\hat{U}^{-1} = \hat{u}\left(\theta = \pi, \phi = 0, \lambda = -\pi, \gamma = \frac{3}{2}\pi\right)$

$\hat{U}^{-1} = \hat{u}\left(\theta = \pi, \phi = \frac{\pi}{2}, \lambda = -\frac{\pi}{2}, \gamma = \pi\right)$

- Let's note that

$$\hat{U}^{-2} = \left(\hat{U}^2\right)^{-1} = \left(\hat{U}^2\right)^{\dagger}$$

- Therefore, in the analyzed example

$$\hat{U}^{-2} = \hat{U}^2 =$$

$$= \hat{u}(\theta = 0, \phi = 0, \lambda = 0, \gamma = \pi)$$

Just a reminder

$$\hat{U} = \hat{u}\left(\theta = \pi, \phi = 0, \lambda = \pi, \gamma = \frac{\pi}{2}\right)$$

$$\hat{U}^2 = \hat{u}(\theta = 0, \phi = 0, \lambda = 0, \gamma = \pi)$$

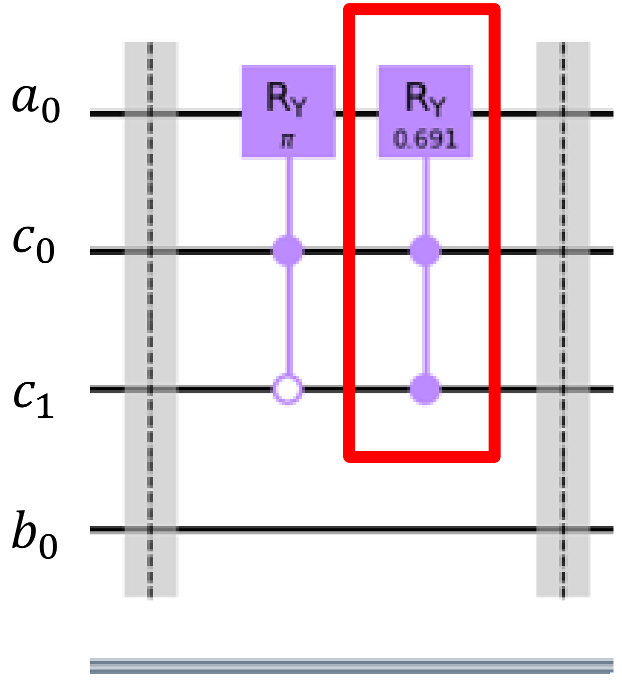
TIP 3

A controlled operation $\widehat{ccR}_y(\theta)$

- Therefore, the operation $\lambda_i \to \lambda_i^{-1}$ implemented through controlled gates $\widehat{ccR}_y(\theta)$ can be determined by the truth table

c_1	c_0	$\theta(c)=\theta(c_1c_0)$	Rotation on ancilla
0	0	0	0
0	1	π	Ry(π)
1	0	0	0
1	1	11/50π	Ry(11π/50)

- Quantum circuit $\widehat{ccR}_y(\theta)$



In file *Proj_5_2025_Q_2_2_1.ipynb* change code in cel:

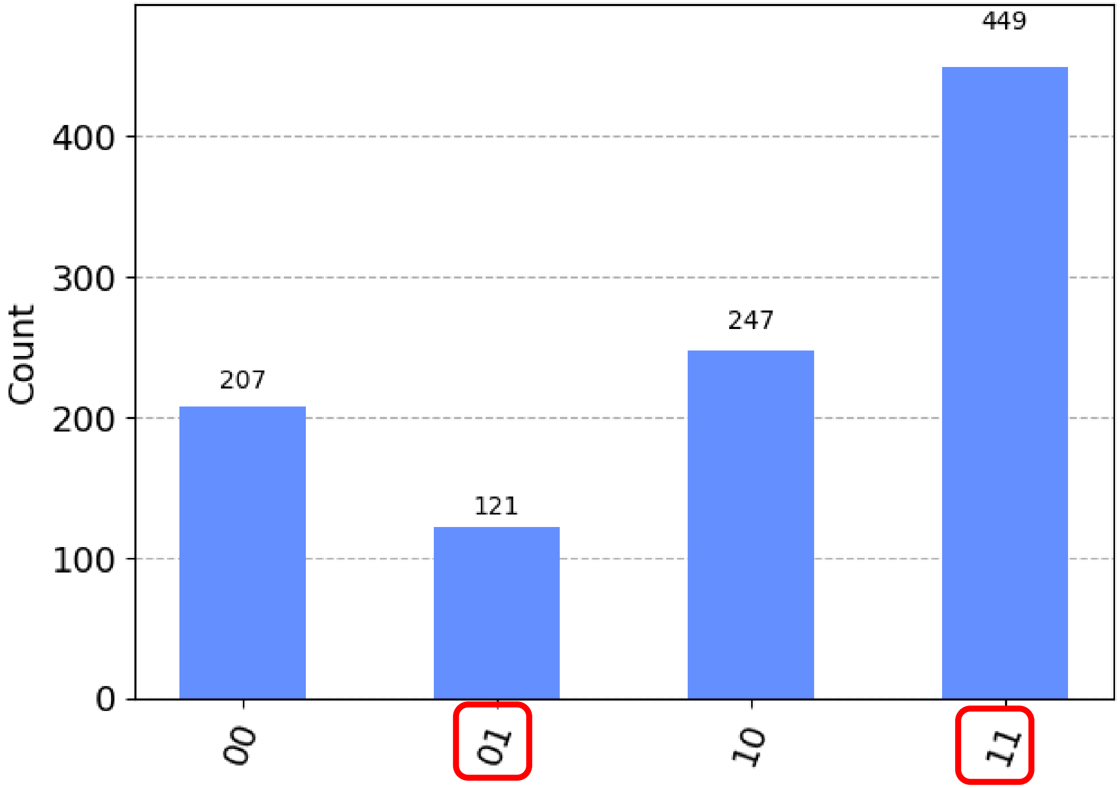
```
def hhl(circ, ancilla, clock, input, measurement):
ccRy2=qulib.RYGate((11/50)*np.pi).control(2, ctrl_state='11')
```

TIP 4

Probability ratios for the result on the ancilla qubit "1"

You can expect these results in a histogram

- Histogram of counts after measurement on the auxiliary register and b



The End