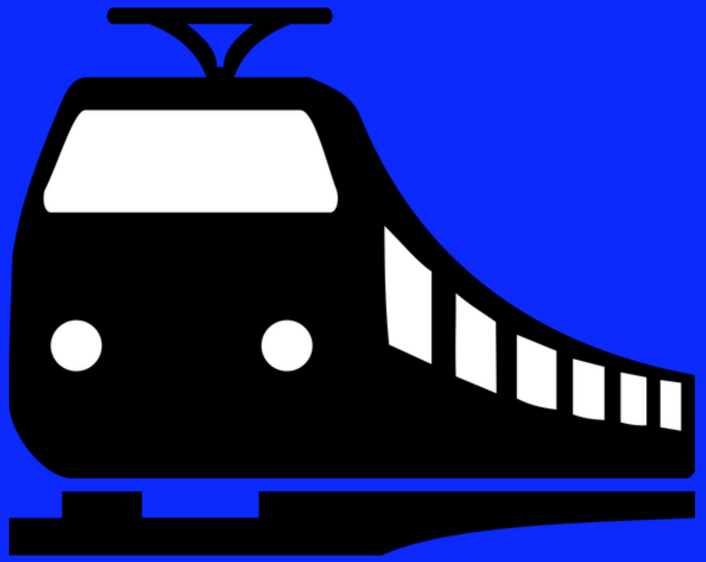
Solving Optimization Problems with Quantum Computing

Lai, Chia-Tso 3786011

Optimization problems are very common in many fields of research and industries such as computer science, biology, chemistry, logistics, energy sectors, finance and manufacturing. However, most of these problems are NP-hard.



Quantum computing is potentially a game changer for optimization!

Prevalent Paradigms of Quantum Optimization

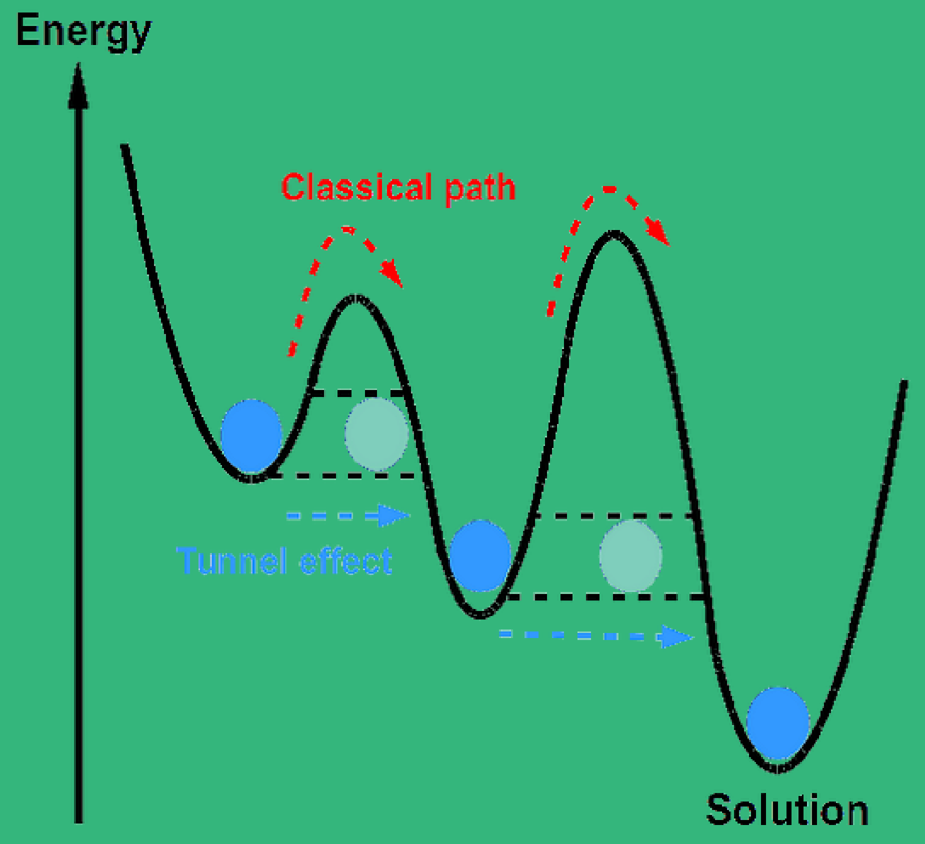
Adiabatic Quantum Computing/ Quantum Annealing

Grover's Search Algorithm

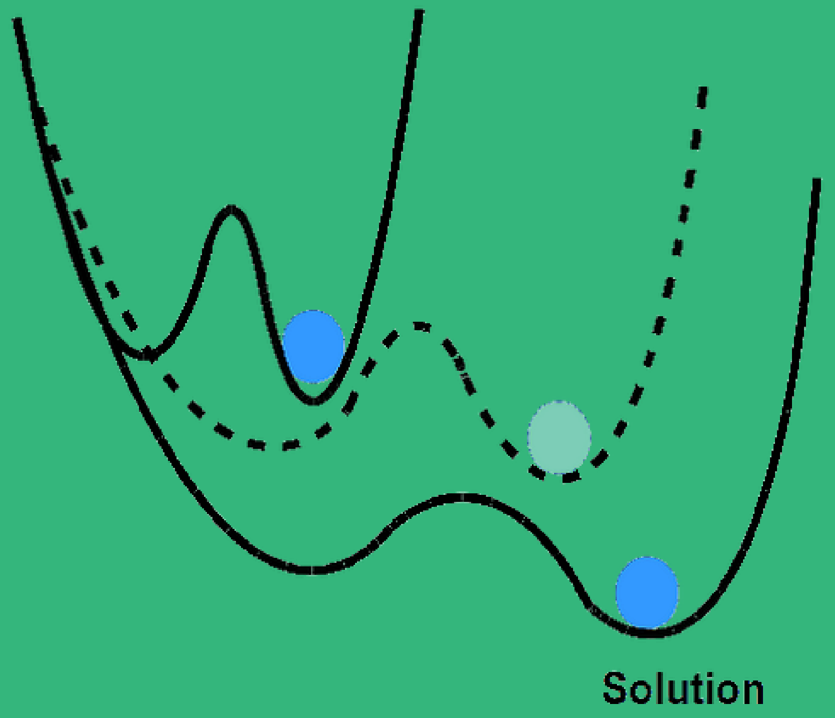
Variational Quantum Algorithm

Hybrid Algorithms

Adiabatic Quantum Computing



Quantum Tunnelling



Adiabatic evolution

$$H_c = \sum_i h_i s_i^z + \sum_i \sum_{j>i} J_{ij} s_i^z s_j^z$$

$$H_I = \sum_i h_i' s_i^x$$

$$H = (1-s)H_I + sH_c$$

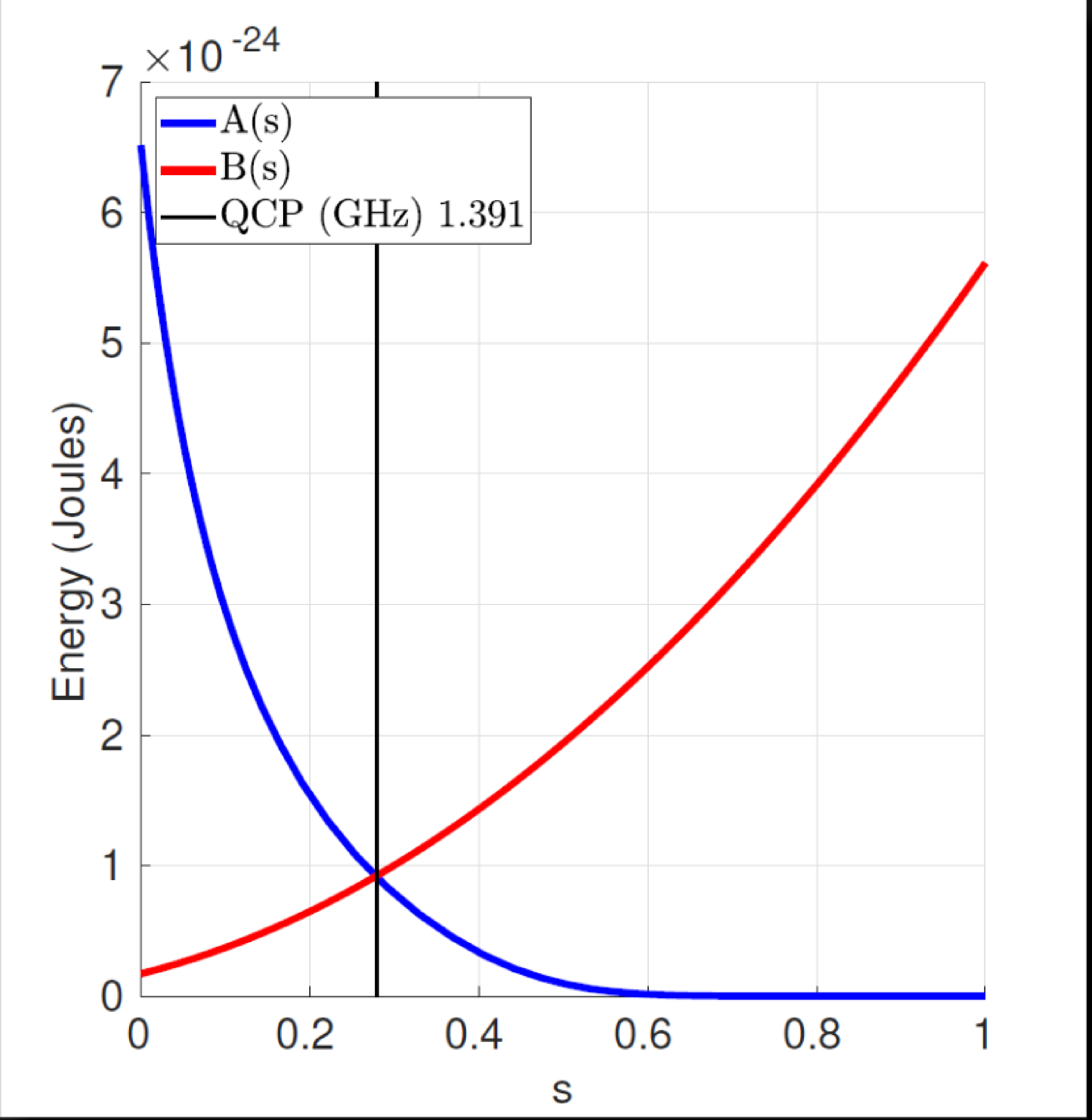
Adiabatic quantum computing (quantum annealing) is a quantum computing approach that encodes the cost function as the Hamiltonian of an Ising model, and then evolves the initial quantum state adiabatically to the ground state of this Hamiltonian.

D:Wave

|QuEra⟩ Computing Inc.

PASQAL

Annealing Protocol



Quantum annealers and analog quantum computers are perfect candidates for solving quadratic unconstrained binary optimization (QUBO) problems such as MaxCut, Maximum Independent Set, Traveling Salesman

D-Wave:

D-Wave Leap's CQM Hybrid Solver

Scalability : >5000 qubits

superconducting qubits with 15 couplers/qubit

Can easily impose quadratic constraints

Quantum Shortest-Path Algorithm

Compute the shortest distance between two nodes of a network

Each station is represent as a binary variable Y. if the station is on the path then Y = 1

The sum of the distance between any two connected stations on the path is encoded as the cost Hamiltonian.

A valid path is a connected subgraph including the start and the end points

Objective:

$$Hc = \sum_{k>s} W_{ks} A_{ks} Y_k Y_s$$

Constraints:

1.The start and the end points are on the path:

$$Y_{start} = Y_{end} = 1$$

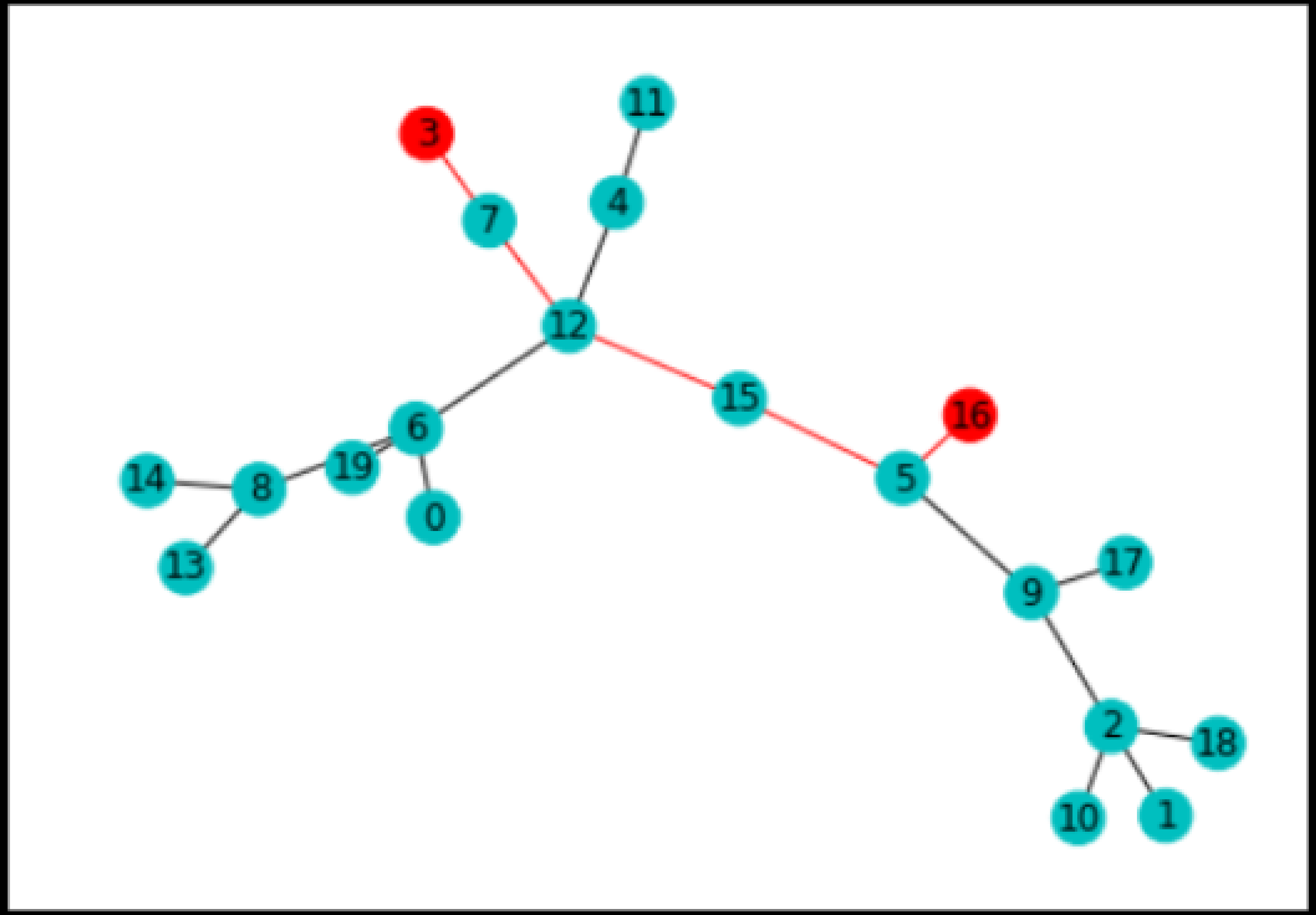
2. Both start and end points need to be connected exactly once:

$$\sum_k A_{start,k} Y_k = 1 \qquad \sum_k A_{end,k} Y_k = 1$$

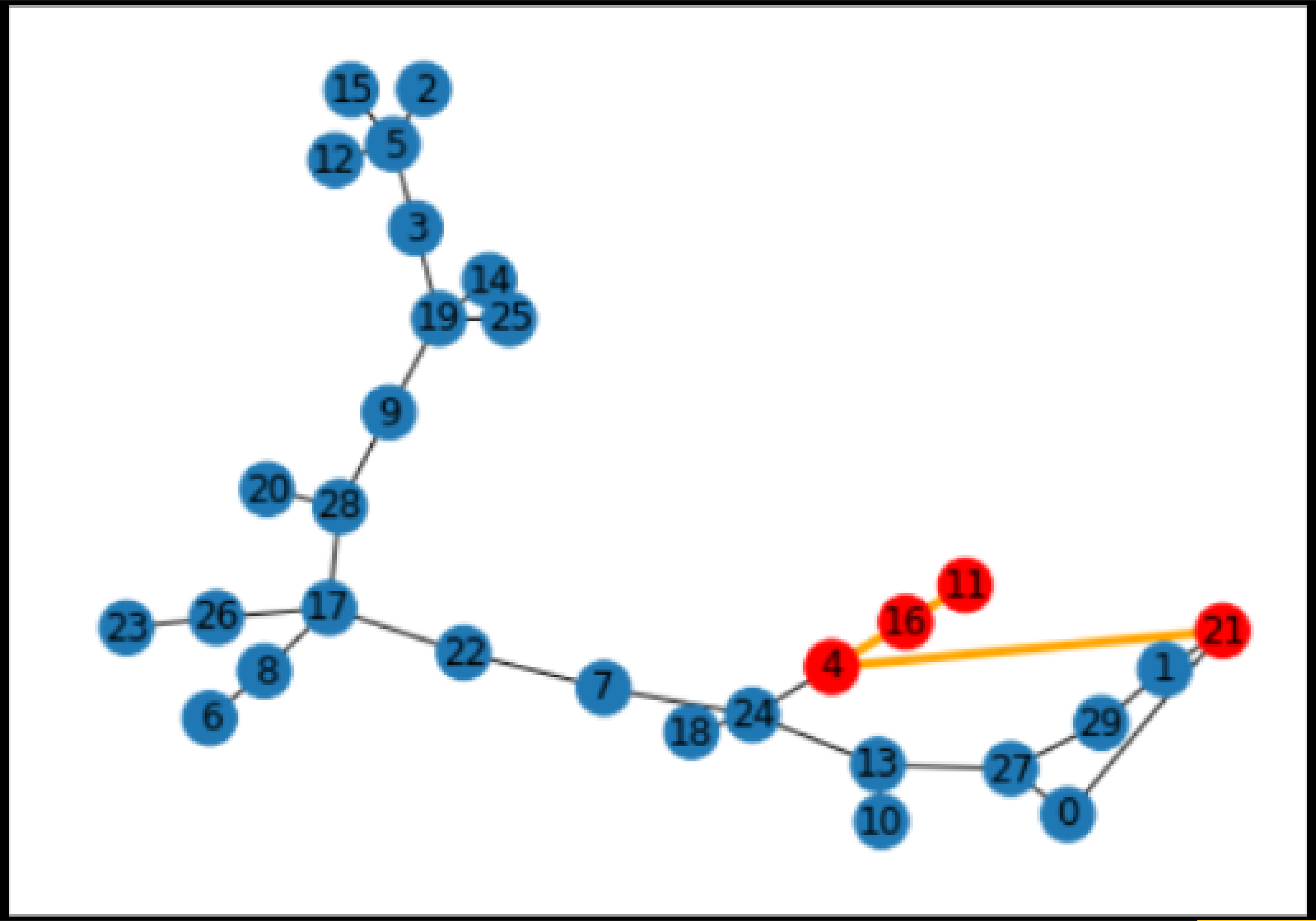
3. Number of connected edges = number of nodes(stations) -1 :

$$\sum_{k>s} A_{ks} Y_k Y_s = \sum_k Y_k - 1$$

Quantum Shortest Path Algorithm



1 possible path



multiple possible paths

QuEra/Pasqal:

Neutral atom quantum computer

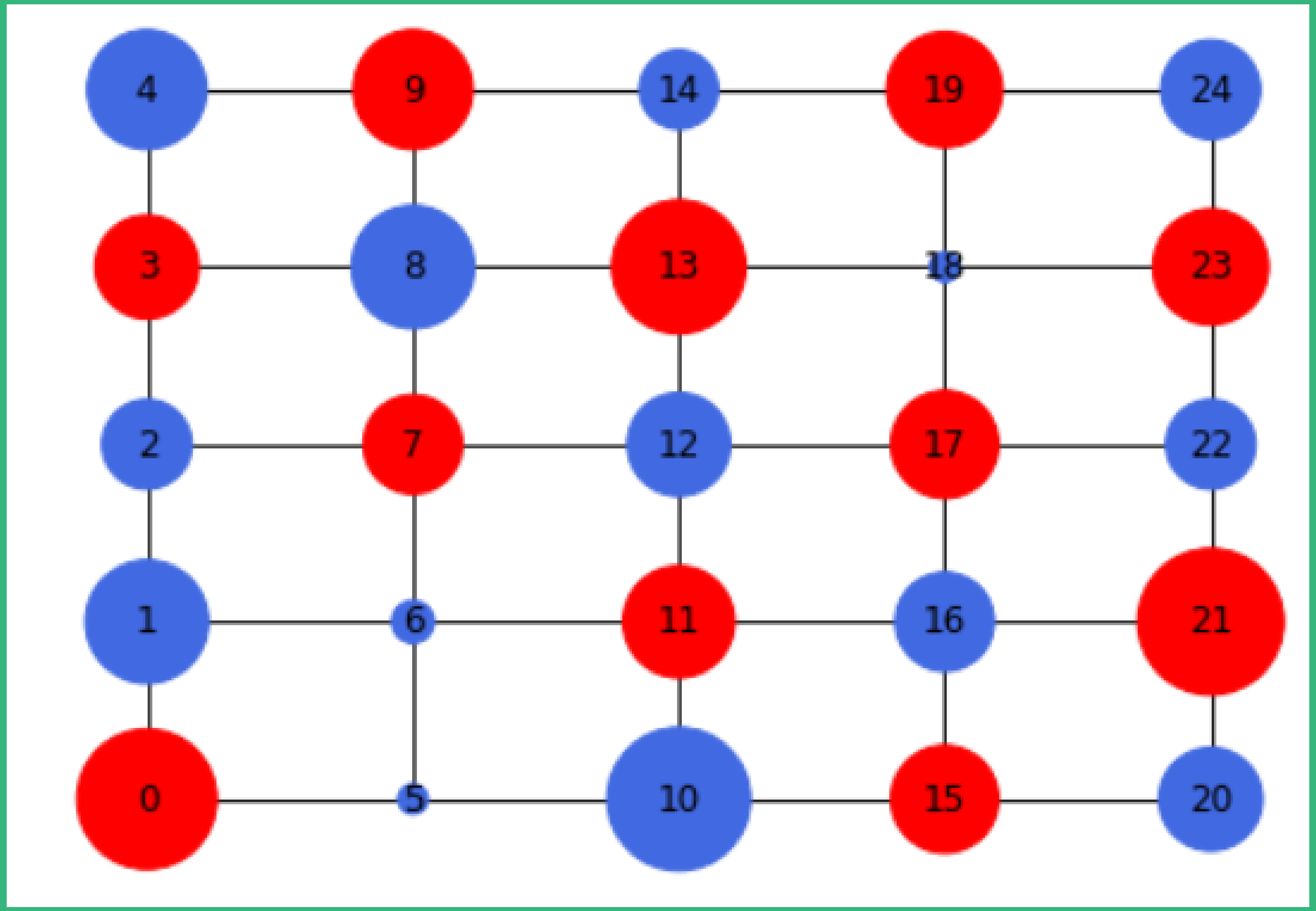
Scalability : 300 qubits

Optical tweezers to trap neutral atom arrays

High flexibility in qubits spatial arrangement

Rydberg blockade for entanglement

Maximum Weighted Independent Set (MWIS)



$$H_{Ryd}(t) = \frac{\hbar}{2}\sum_i \Omega_i(t)\sigma_i^x - \frac{\hbar}{2}\sum_i \delta_i(t)\sigma_i^z + \sum_{i<j} V_{ij}\sigma_i^x\sigma_j^z$$

Maximum Independent Set is a graph theory problem to find out the biggest independent set (nodes in the set are disconnected from each other). MWIS is to maximize the weights of nodes in the set while satisfying the independent set constraint.

MWIS encoding is inherent to Rydberg atoms Hamiltonian, in which the Rydberg blockade imposes the independent set constraint.

Gate-based QC software toolkits:

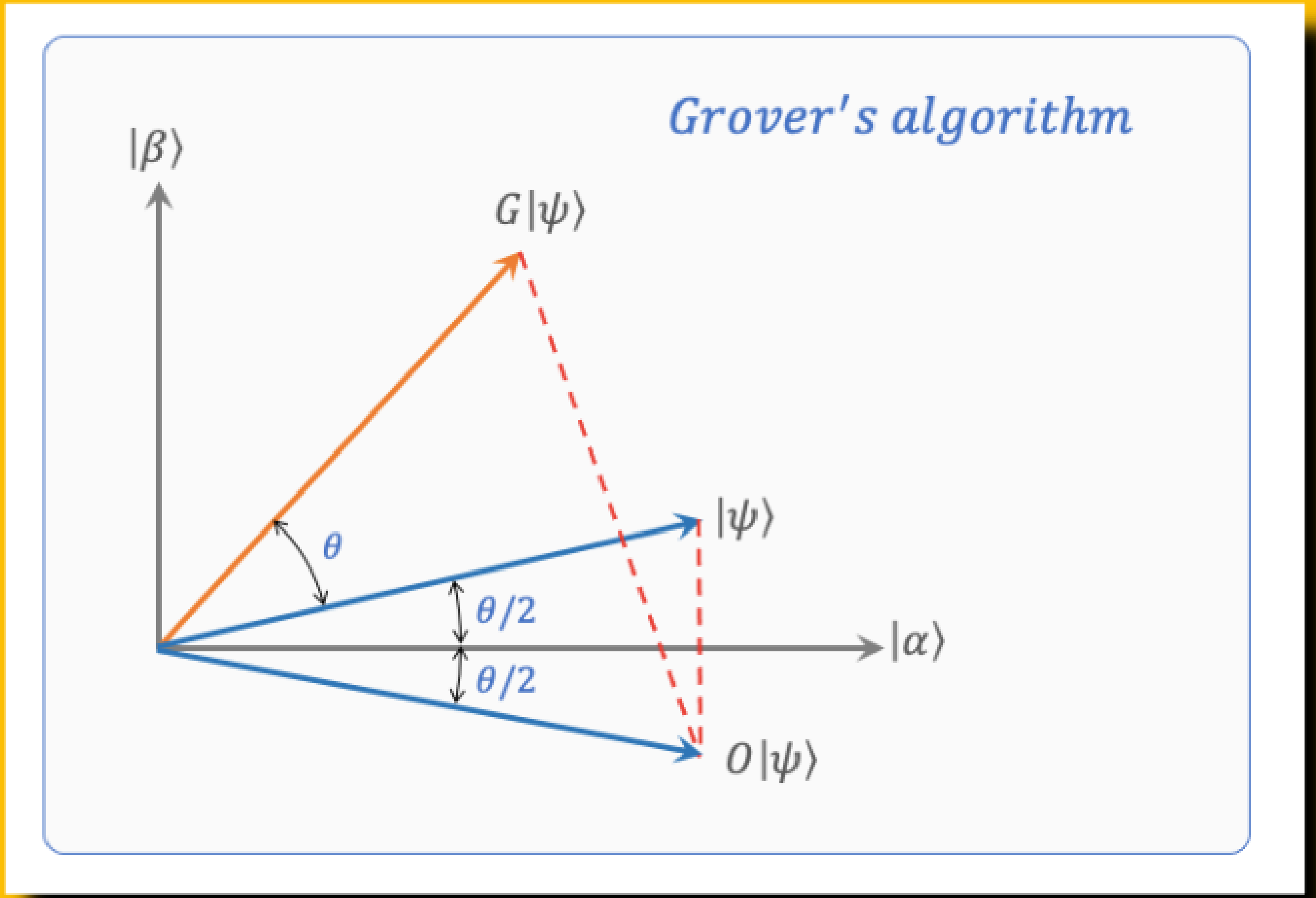
Qiskit, PennyLane, Cirq

Quantum Simulator/ Small size noisy quantum computer

Easy to program and visualize quantum circuit and gate operations

Abundant quantum algorithm librarys : QFT, Grover's algorithm, QAOA, etc.

Grover's Search Algorithm



$$G = AS_0A^{\dagger}S_x$$

Grover's search algorithm aims at finding the marked states (optimal solution) of a solution space by amplifying the amplitude of the desired states with Grover's operators

The steps can be summarized as:

1. Prepare the superposition states that encompasses the solution space
2. Reflection about the bad state (flip the phase of the marked state)
3. Reflection about the prepared state to amplify the amplitude of the marked states

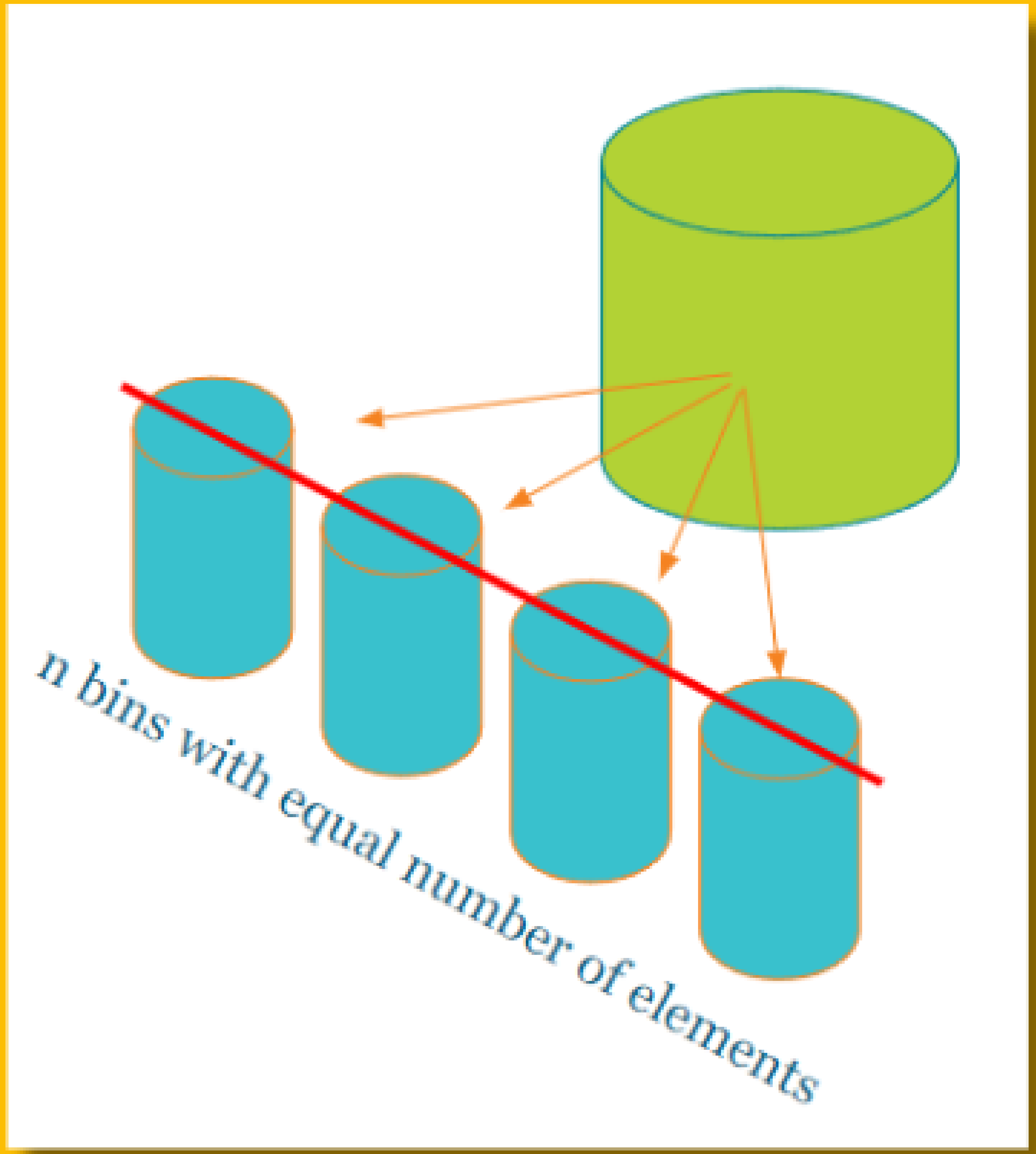
Multiway Number Partitionig

An array with N weighted elements :
W=[w1,w2, ... ,wN]

Assign the elements uniformly to n bins

Compute the total weight of each bin

Objective:
Minimize the standard deviation of the total weights



Multiway Number Partitionig

1.At least N+log2(N)+1 qubits required

2.Each qubit represents an element (i = 1,2,...,N)

3.Initialize the state by applying Hadamard gates to every element qubit

Oracle Design:

4. Set up the objective by C-Ry and Ry gates. The rotational angles of C-Ry gates correspond to the elements' weights (mapped to $[0, \pi]$), while the angle of Ry gate encodes the mean value of each bin

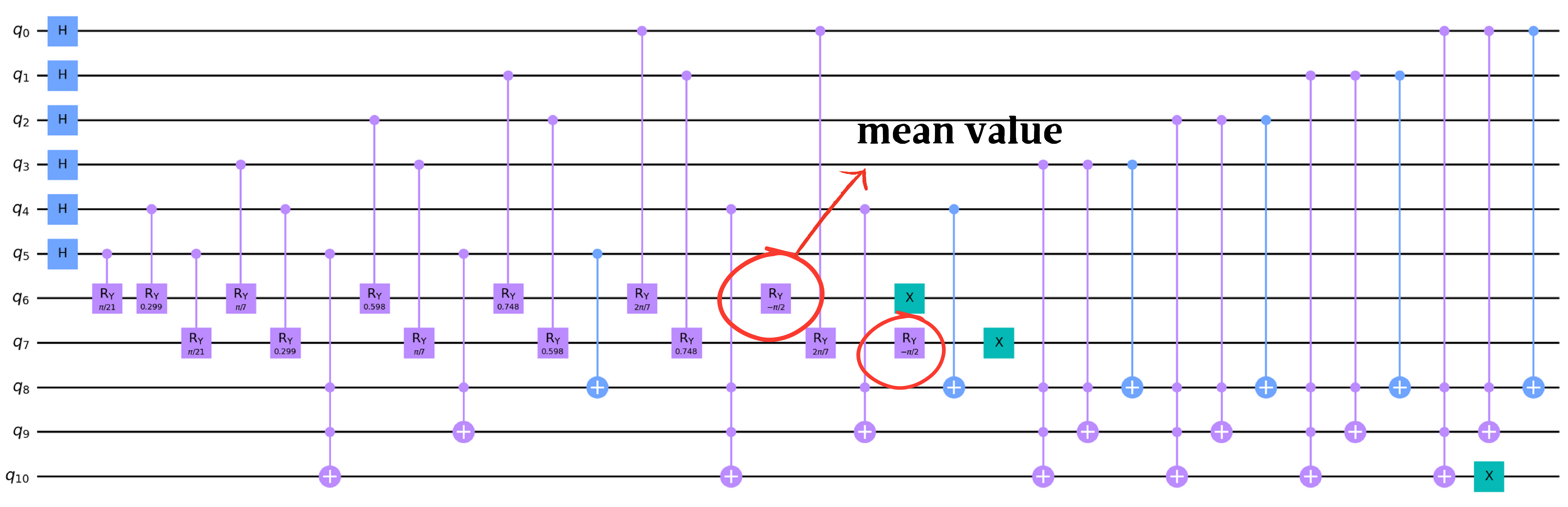
5. The constraint of N/n elements in each bin is imposed by CNOT gates, which implement addition of 1 if the element qubit is chosen

6. Apply phase flip gate to the first qubit if the ancilla qubit is marked as 1

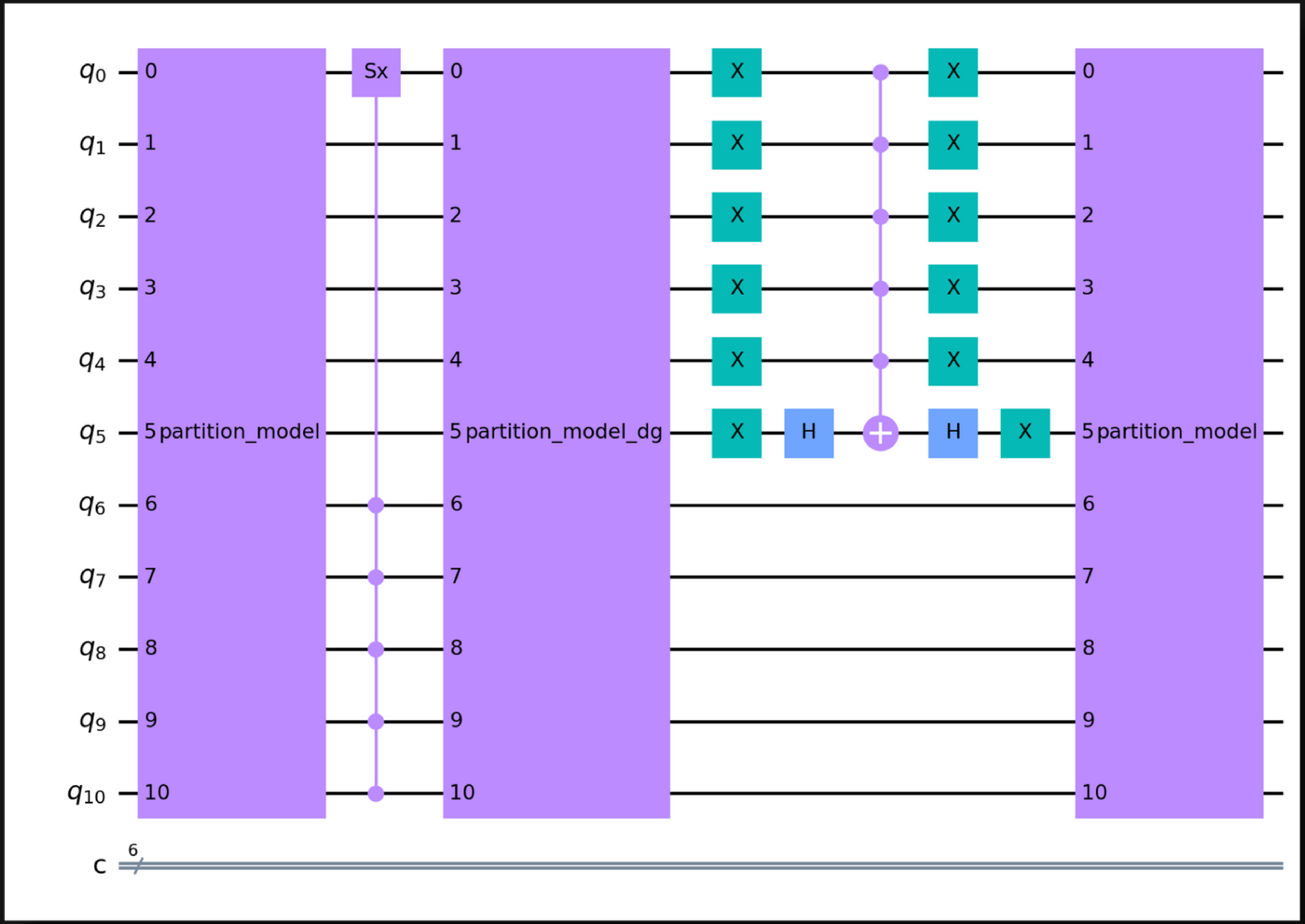
Oracle

Example of 6 elements [1,2, ..,6] into 2 bins

q0-q5: elements
q6-q7: objective
q8-q10: constraint



Grover's Search Algorithm



q0-q5: elements
q6-q7: objective
q8-q10: constraint

Benchmark

samples weight: W = np.ones(N) + np.random.normal(0,1,N)

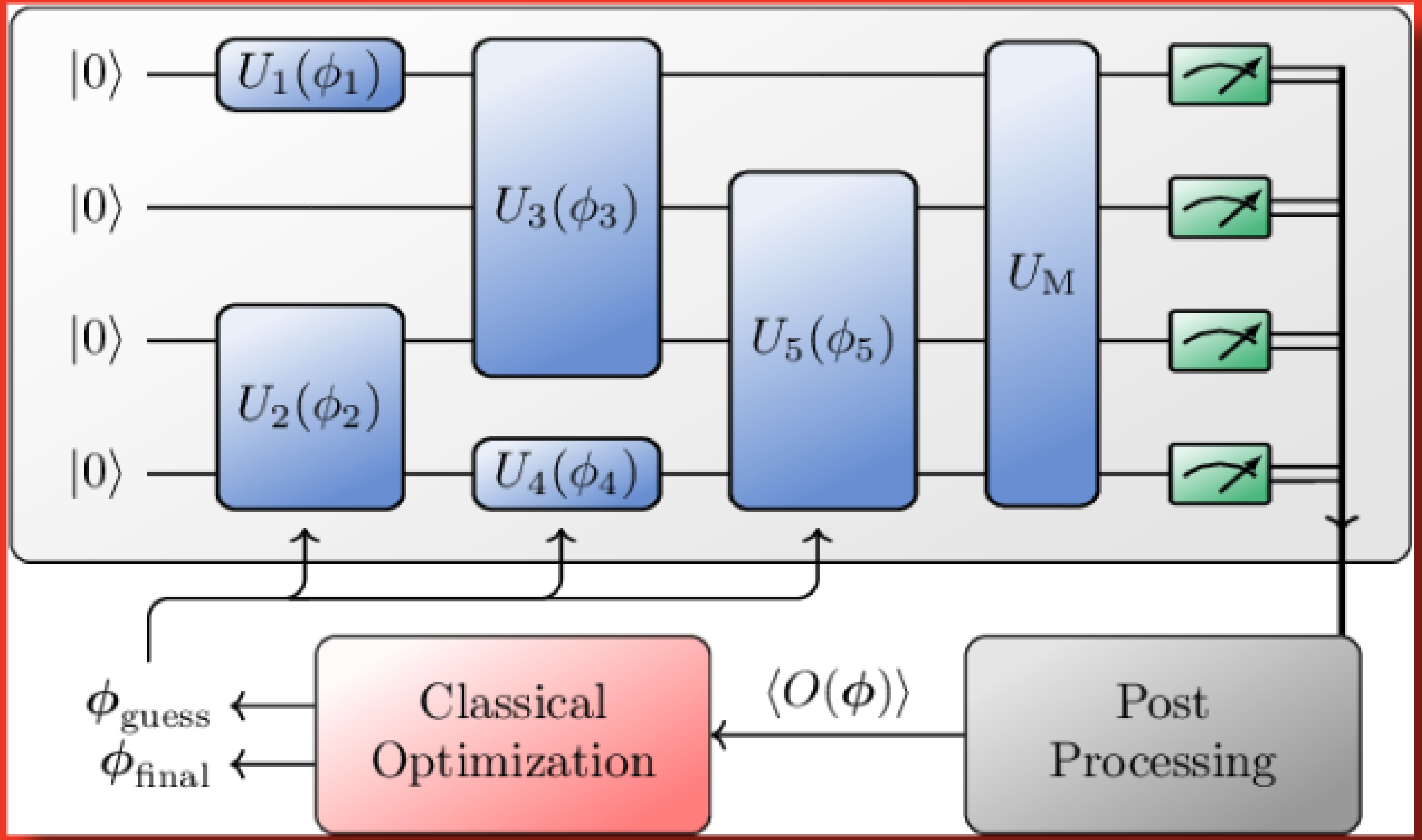
Samples	Heuristic	Grover's Search
9 elements to 3 bins	0.199	0.079
16 elements to 4 bins	0.2095	0.018
20 elements to 5 bins	0.535	0.212
22 elements to 2 bins	0.39	0.025

Benchmark

samples weight: W = np.random.random(N)

Samples	Heuristic	Grover's Search
12 elements to 6 bins	0.1032	0.1046
14 elements to 7 bins	0.065	0.071
20 elements to 4 bins	0.0512	0.0469
22 elements to 2 bins	0.0364	0.0042

Variational Quantum Algorithm

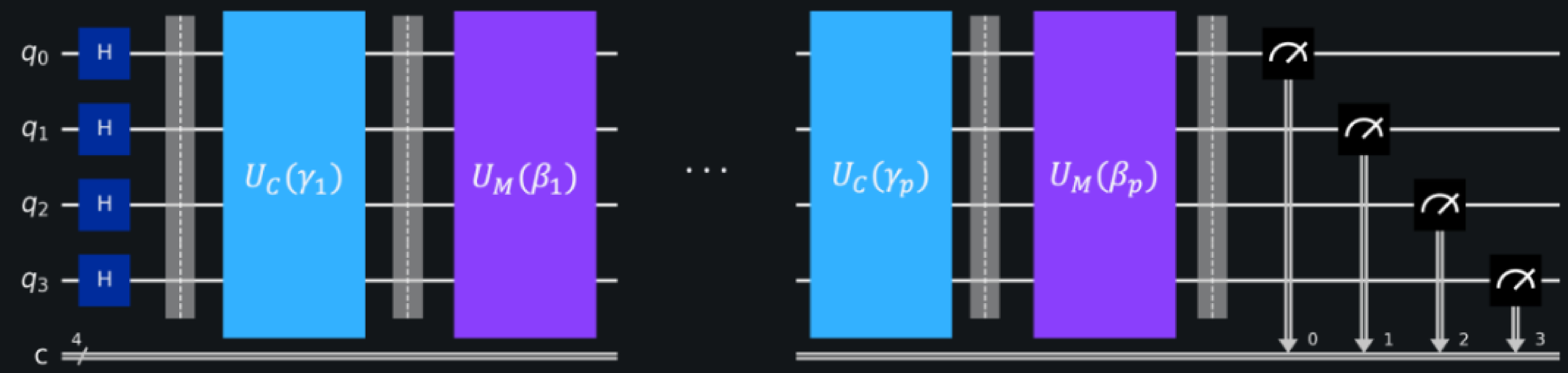


Ex. Quantum Approximate Optimization algorithm (QAOA), Quantum Neural Network(QNN)

A variational quantum algorithm is composed of an ansatz circuit with variables that parameterize the quantum gates. The objective function is encoded in the circuit (i.e. QUBO matrix) and the goal is to train the circuit for a set of parameters that optimize the expectation value of the cost Hamiltonian.

The training of the circuit is usually done by gradient descents, which requires the inclusion of classical optimizers

QAOA



Preparation of equal superposition state

$$|+\rangle^n = \sum_{x\in\{0,1\}^n} \frac{1}{\sqrt{2^n}} |x\rangle$$

p repetitions of alternating cost and mixer layers

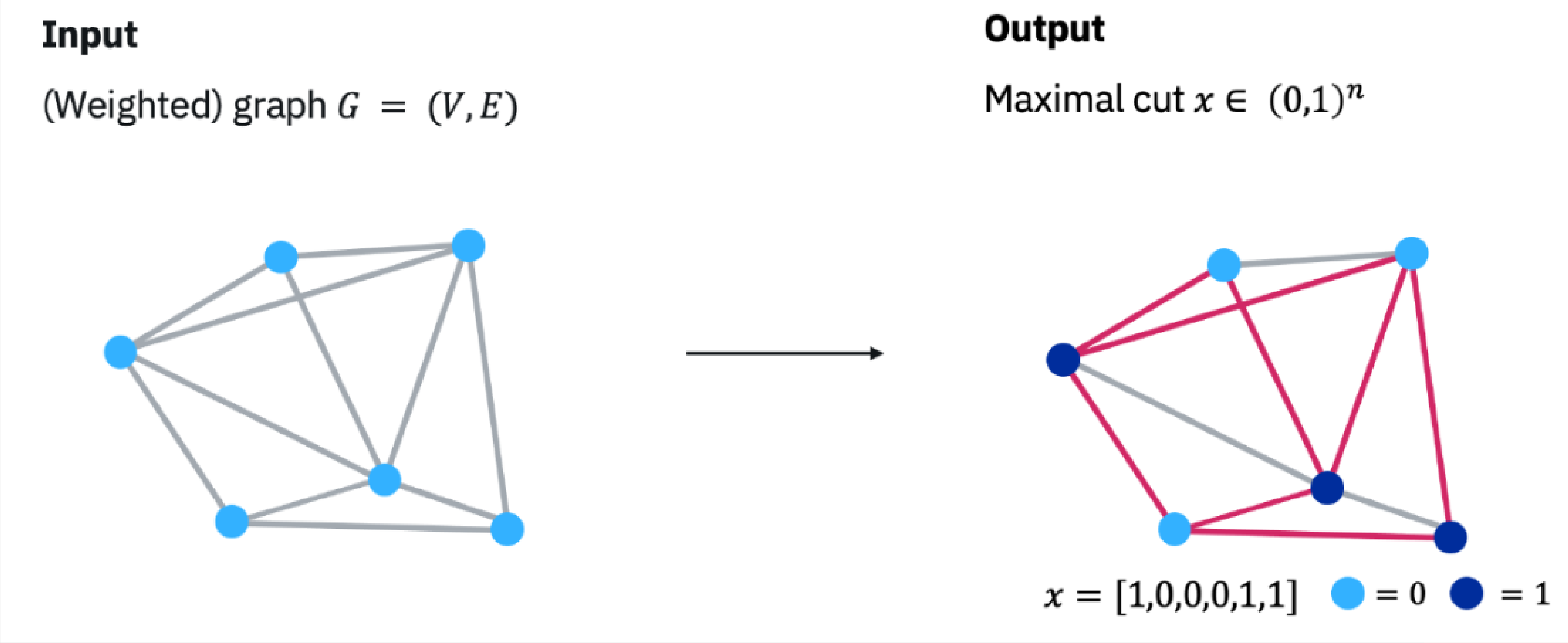
$$U_C(\gamma_i) = e^{-i\gamma_i H_C}$$

$$U_M(\beta_i) = e^{-i\beta_i H_M}$$

Measurement in computational basis

MaxCut Problem

Cut through edges of a graph with several connected nodes such that the sum of these edges is maximized



The MaxCut problem can be mapped onto the Ising model, whose unitary propagator can be decomposed into Rz and Rzz gates

QUBO form

$$H_c = \sum_{i,j} W_{ij} x_i (1 - x_j)$$

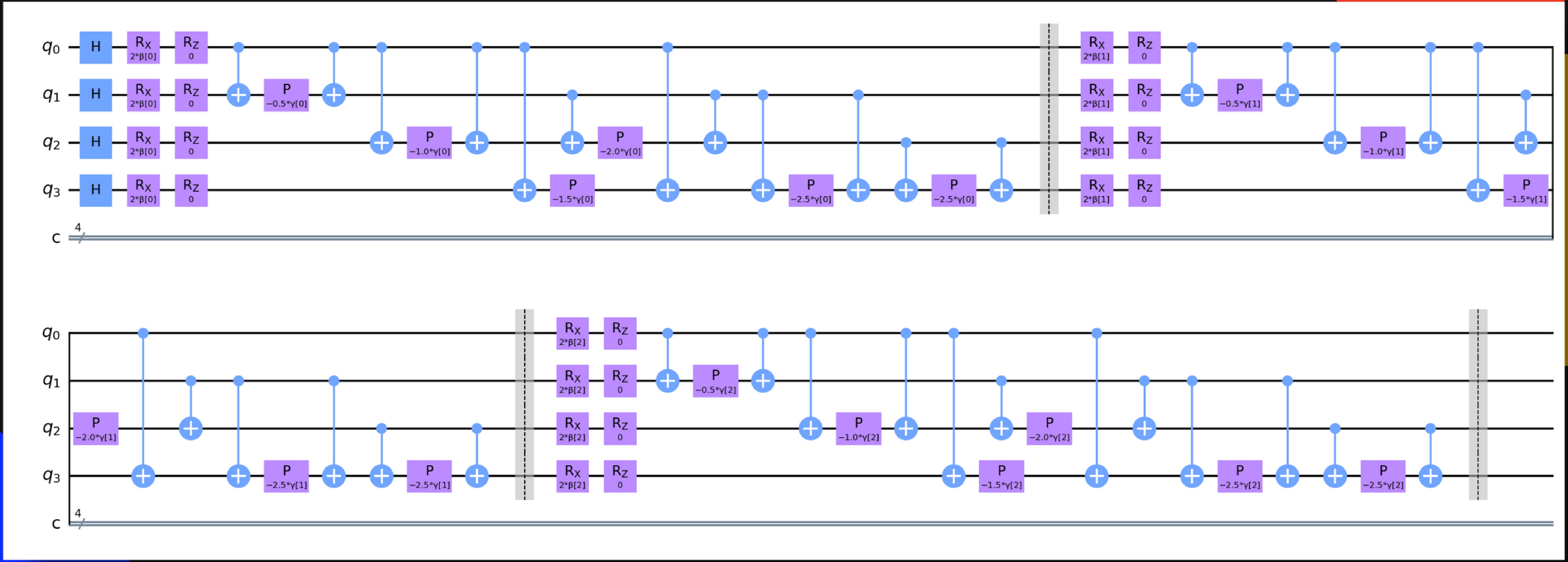
$$H_c = \sum_i \left(\sum_j W_{ij}\right) x_i - \sum_{i,j} W_{ij} x_i x_j$$

Ising form

$$H_c = -\frac{1}{4} \sum_{ij} W_{ij} s_i^z s_j^z + \frac{1}{4} \sum_{ij} W_{ij}$$

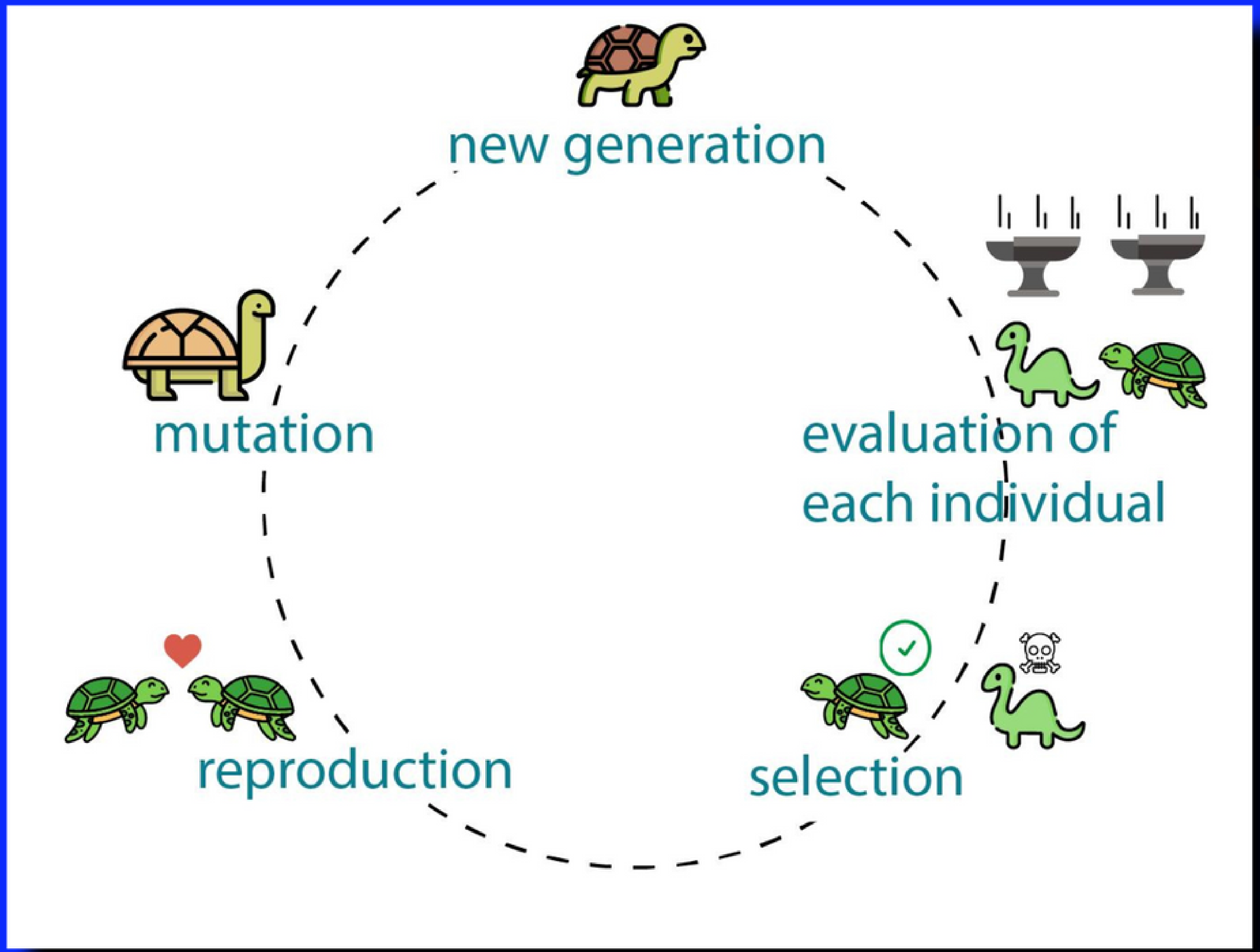
$$x_i = \frac{1 - s_i^z}{2}$$

QAOA for MaxCut:



Hybrid quantum-classical algorithm exploits the advantage of both methods. One example is the quantum genetic algorithm, which expands the scope of problems we can solve to continuous and black-box optimization.

Genetic algorithm



Genetic algorithm is a classical evolutionary algorithm for solving optimization problems. It mimics the evolution theory of nature and evolves the population by selecting the fittest individuals according to the target function.

Compared to gradient-based optimization algorithms, genetic algorithm is more suitable for discrete optimization and black-box function optimization

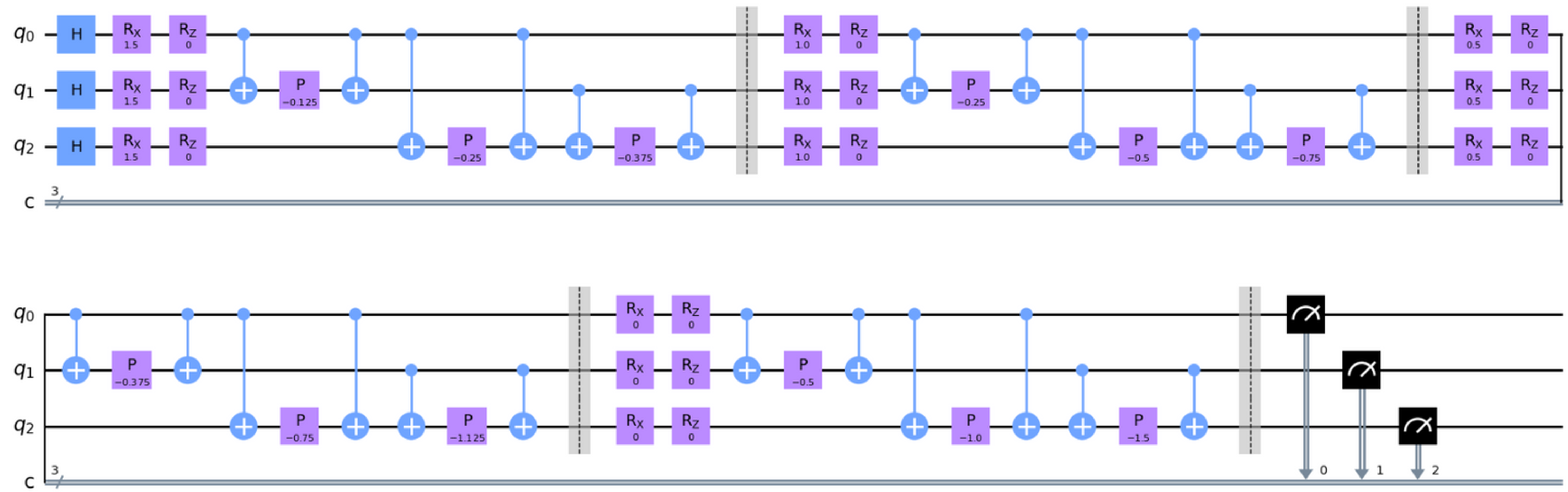
Population generation → Fitness evaluation → Parents Selection → Crossover → Mutation

Quantum genetic algorithm replaces some of the evolutionary steps with a quantum subroutine to speed up the searching process or to assist in attaining certain goals.

QAOA + Genetic Algorithm

Replace the population generation with a fixed QAOA circuit to reduce the search space in the beginning.

The measurement result is taken as the first generation of the genetic algorithm

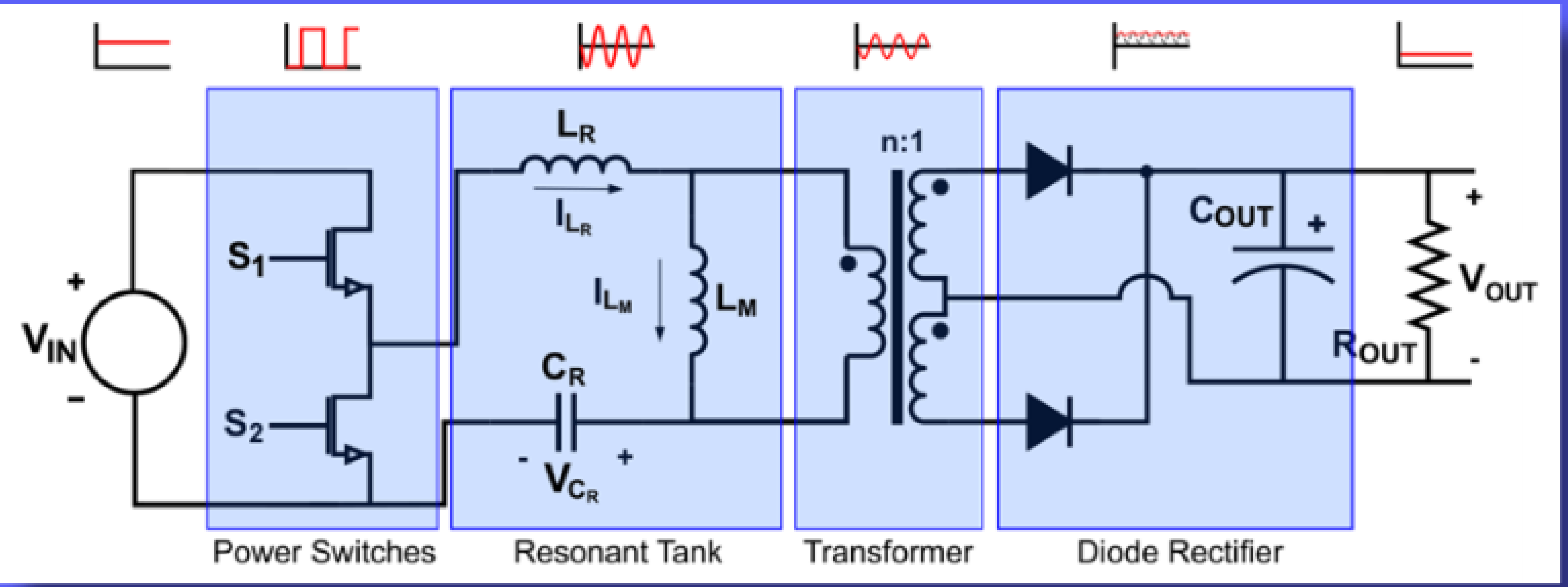


Benchmark

Size		Hybrid	Classical
5 nodes	std	0	0
	iteration	3	11
10 nodes	std	0	9.35
	iteration	2664	5000
15 nodes	std	0	5.52
	iteration	7356	10000

Hybrid algorithm converges faster!

LLC Circuit Optimization



Genetic algorithms can also be applied to optimizing power electronics devices

1. Maximize the following function:

$$-(LLC(L1, L2, c1, f) - 200)^2$$

2. Find a diverse set of good solutions

Continuous-variable black-box function optimization is a fairly complicated problem. Not only does the lack of analytical information limit the use of gradient-based methods, but also the infinite solution space of continuous variables makes the exploration cumbersome for genetic algorithms.

Quantum computing might help with compressing the variables' domains and diversifying the solution set, while genetic algorithms can evaluate the fitness values and performs crossover.

Quantum-Inspired Encoding

Map the continuous variables onto the Bloch ball;
Encode the continuous variables with r,θ, Φ

n qubits can encode 3n variables

Encoding scheme:

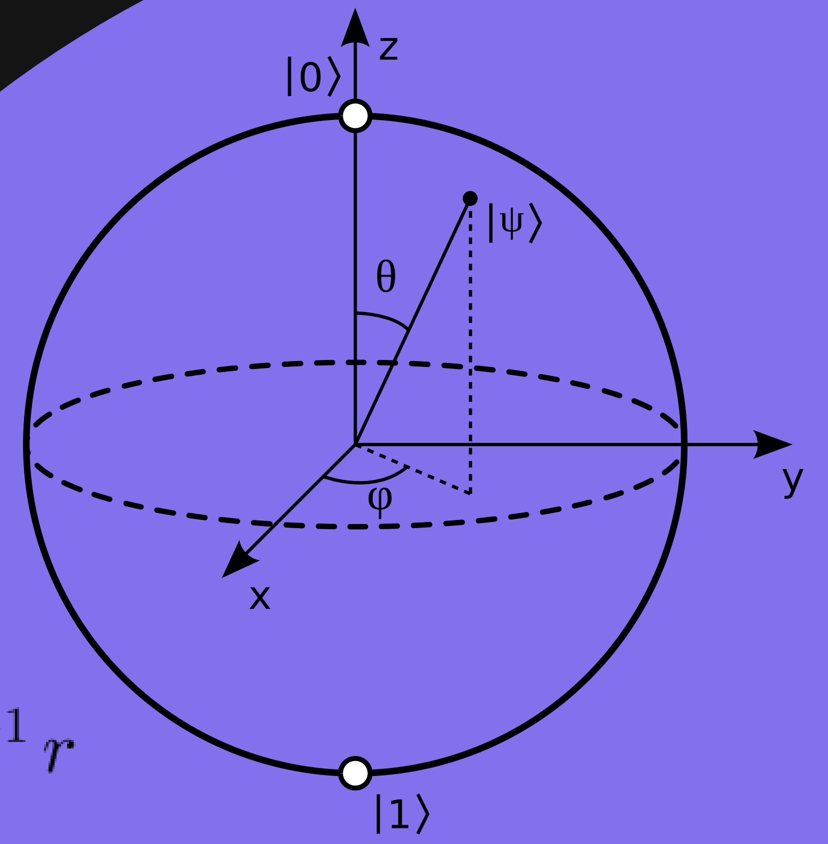
One ancilla qubit for entanglements

C-Ry gates encode r, Ry gates encode θ,Rz gates encode Φ

To prepare the qubits in the desired states parametrized by the Bloch's vectors, (r,θ,Φ) needs to be mapped to the corresponding rotation angles (α,β,γ)

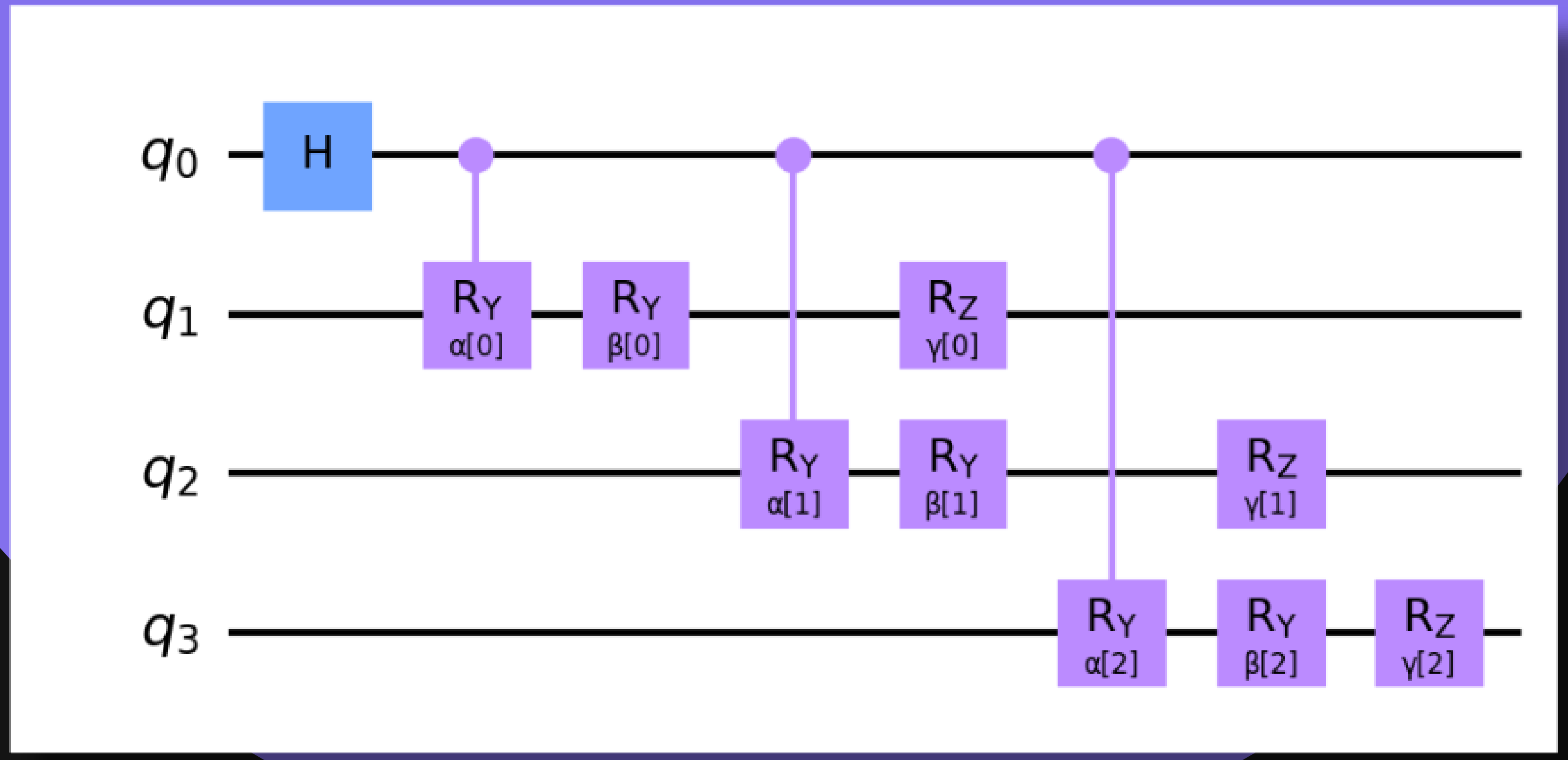
Read-out:

Single-qubit tomography and then map the measured state back to the varaibles



$$\theta \in [0, \pi]$$
$$\phi \in [0, 2\pi)$$
$$r \in [0, 1]$$

$$\alpha = 2\cos^{-1} r$$
$$\gamma = -\phi$$
$$\beta = \sin^{-1}(r \sin\theta\cos\phi + \sqrt{1 - r^2}\sqrt{1 - \sin^2\theta\cos^2\phi})$$



Selection with Quantum Annealing

Quantum annealing can help diversifying the solution set by solving the QUBO problem considering both the fitness values and the pairwise distance between individuals in the population.

Each individual in the population is labeled as x and is associated with a binary variable σ . If the individual is selected , then σ=1.

The quadratic term in the cost Hamiltonian is the pairwise distance between 2 individuals. The greater the distance, the higher the diversity.

The linear term in the Hamiltonian is the fitness value of the individual.

QUBO matrix:

$$Q_{ii} = -\alpha f(x_i)$$

$$Q_{ij} = -\beta |dist(x_i, x_j)|$$

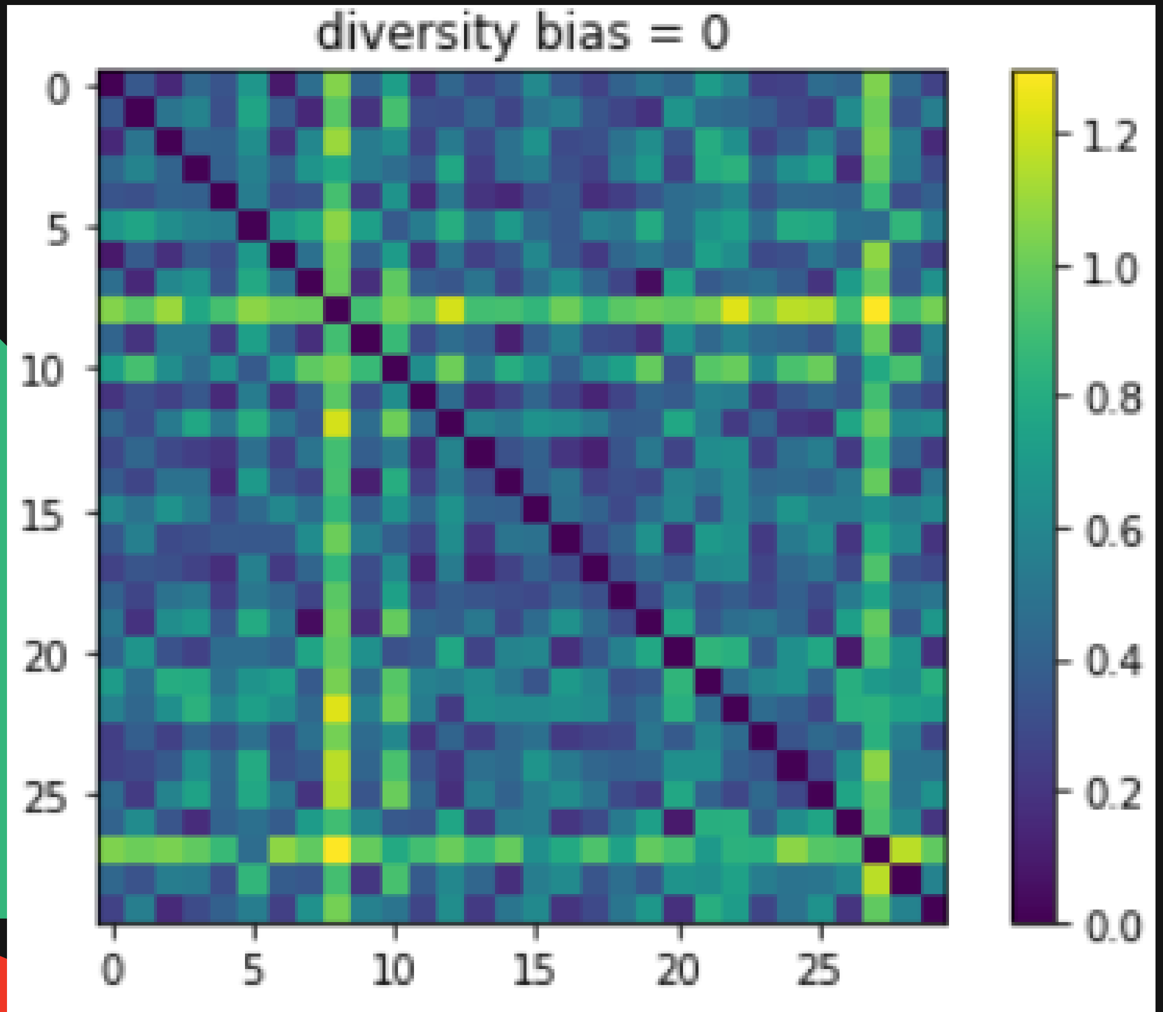
Cost Hamiltonian:

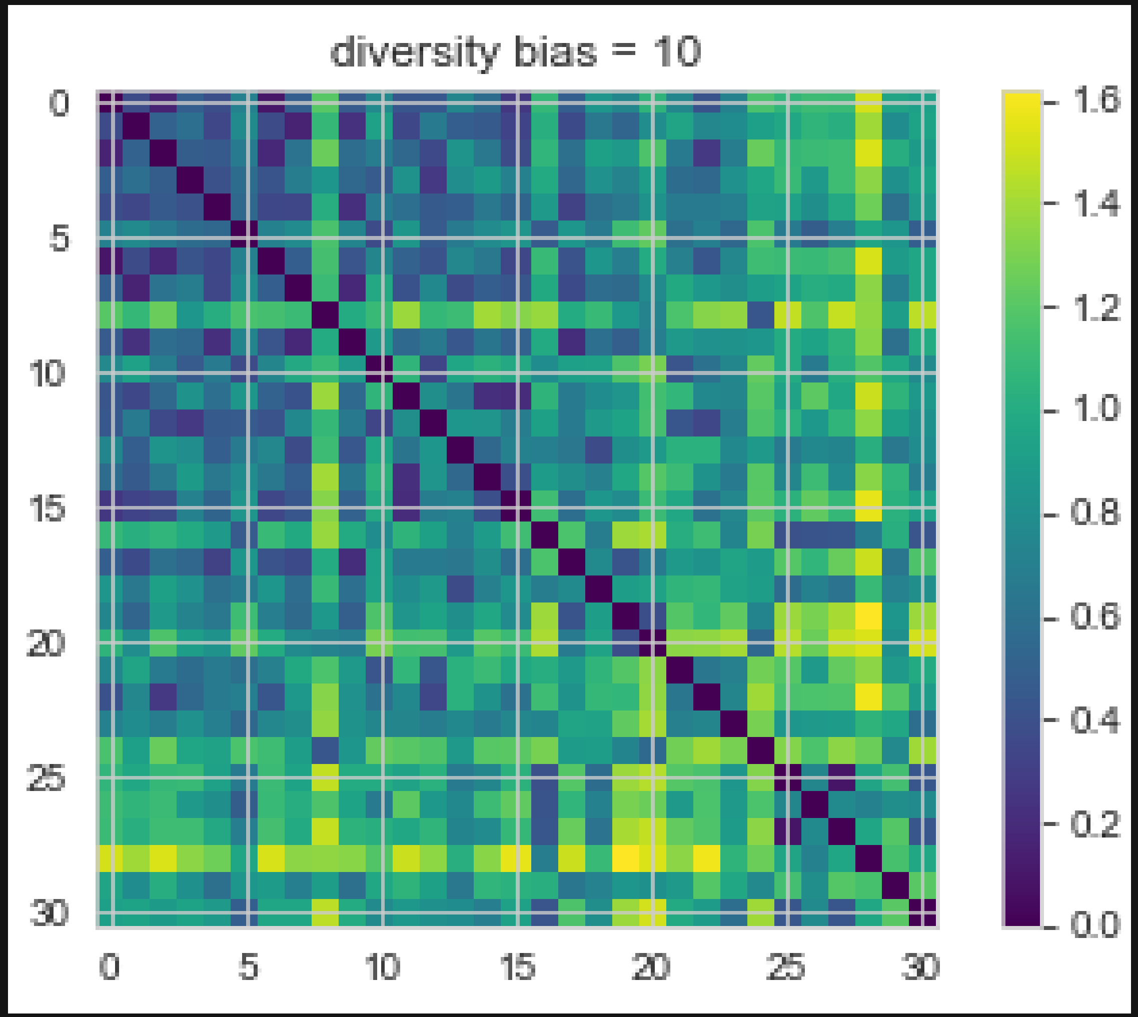
$$H_c = \sum_{i<j} Q_{ij}\sigma_i\sigma_j + (\sum_i \sigma_i - \mu)^2$$

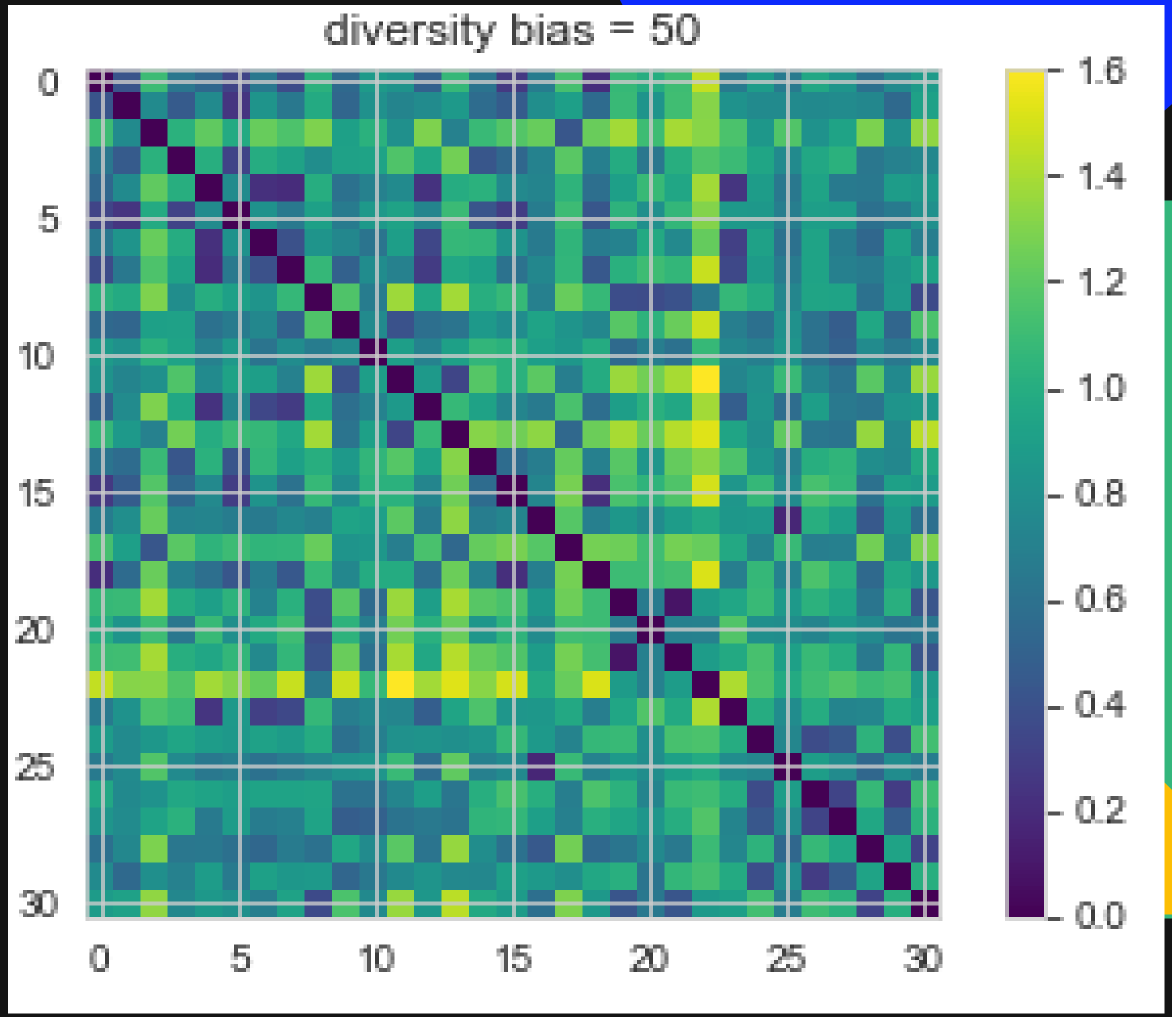
Optimization Result :

pairwise distance= 0.5726

pairwise distance= 0.839







[1.31268783e-04, 2.25336851e-04, 2.66535780e-09, 4.89312111e+05] f(x) = -0.003

[1.01695929e-04, 1.34853641e-04, 3.09654886e-09, 5.44735499e+05] f(x) = -0.012

[2.15860006e-05, 1.42802437e-04, 8.81499275e-09, 2.18828068e+05] f(x) = -0.023

L_1 ∈[2.e-6, 200.e-6]
L_2 ∈[20.e-6, 500.e-6]
C_1 ∈[1.e-9, 10.e-9]
f ∈[100_000, 600_000]

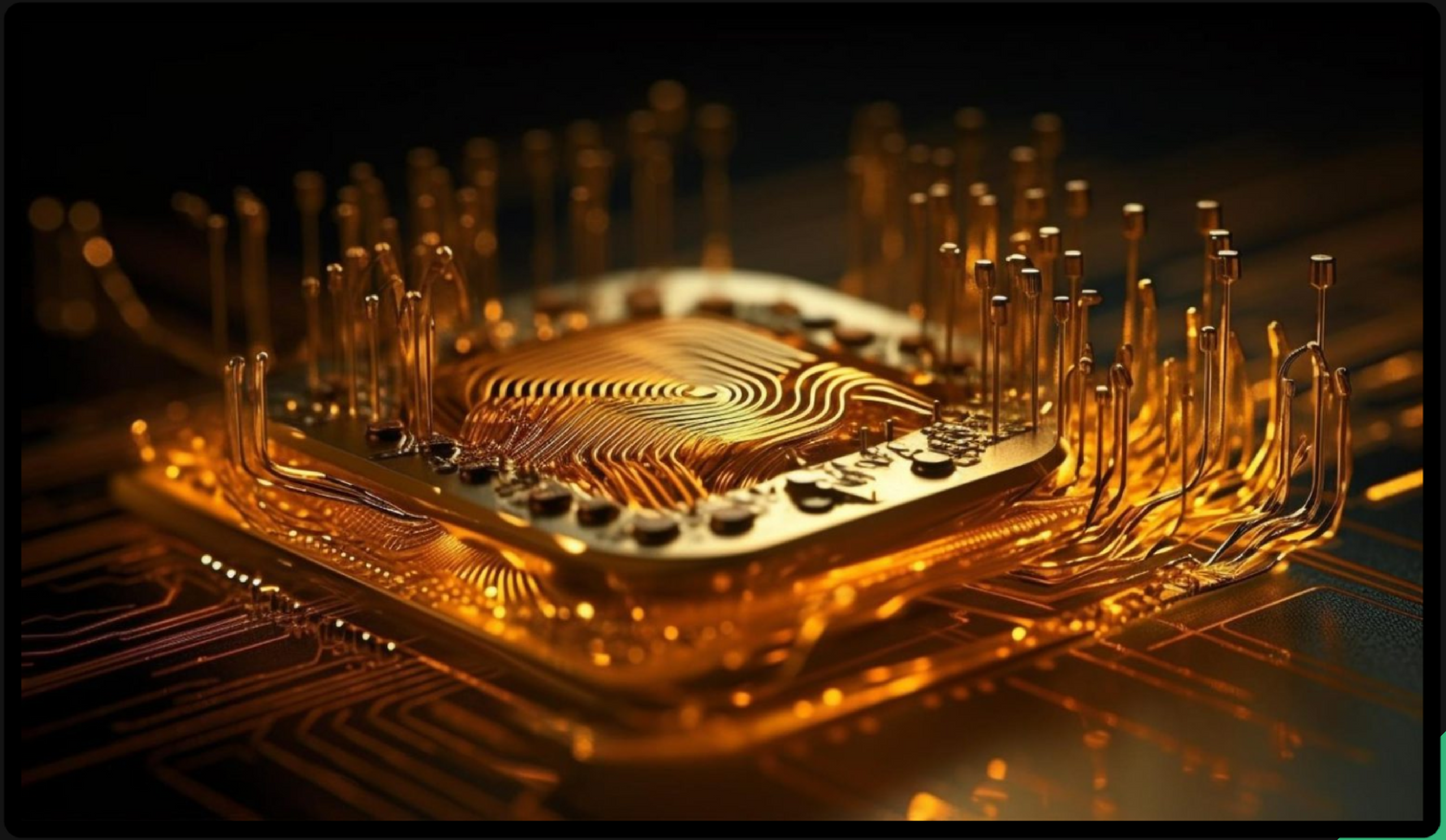
What we can currently achieve with quantum optimization

Adiabatic Quantum Computing/ Quantum Annealing
quadratic combinatorial optimization

Grover's Search Algorithm
small-size discrete optimization

Variational Quantum Algorithm
classification, QUBO, regression

Hybrid Algorithms
continuous-variable optimization, black-box function optimization



THANK YOU FOR YOUR PATIENCE

May quantum be with you!

References:

1. https://www.dwavesys.com/solutions-and-products/systems/
2. https://docs.dwavesys.com/docs/latest/c_qpu_annealing.html
3. https://qiskit.org/documentation/stable/0.35/tutorials/algorithms/06_grover.html
4. https://www.monolithicpower.com/understanding-llc-operation-part-i-power-switches-and-resonant-tank
5. Bermejo, P., & Orus, R. (2022, October 6). Variational quantum continuous optimization: A cornerstone of quantum mathematical analysis. arXiv.org. https://arxiv.org/abs/2210.03136
6. Bittel, L., & Kliesch, M. (2021). Training variational quantum algorithms is NP-hard. Physical Review Letters, 127(12). https://doi.org/10.1103/physrevlett.127.120502
7. Von Dollen, D., Yarkoni, S., Weimer, D., Neukart, F., & Bäck, T. (2022). Quantum-enhanced selection operators for evolutionary algorithms. Proceedings of the Genetic and Evolutionary Computation Conference Companion. https://doi.org/10.1145/3520304.3528915