IQM

Introduction to Quantum Algorithms

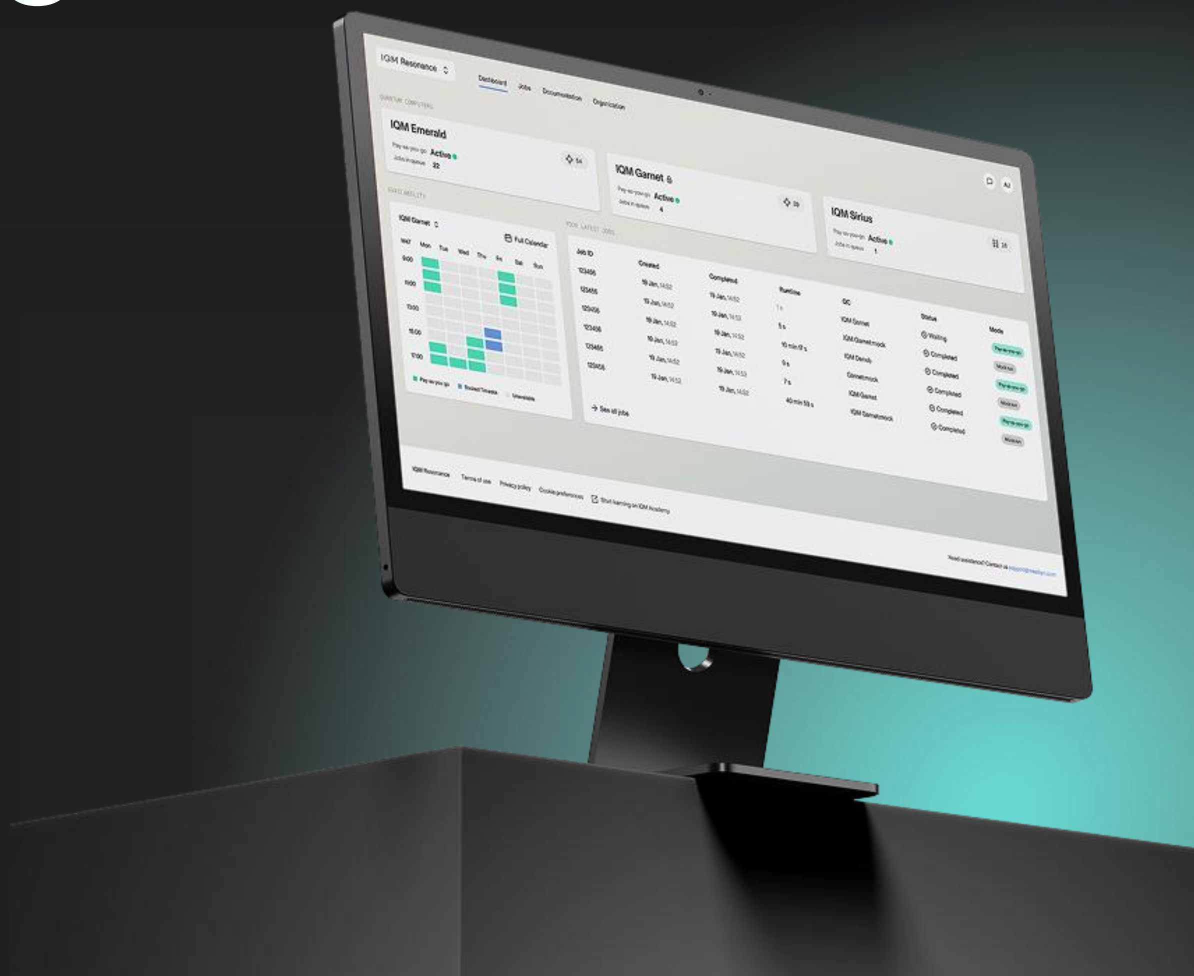
IQM Summer Quantum School 2026

August 26, 2026

Dr. Danny Bulmash, Team Lead, Education and Community

Dr. Nadia Milazzo, Senior Quantum Education Specialist

Niklas Steinmann, Community Manager



What's an algorithm?

Problem

Sort a list of n numbers.

Instance

Sort this list of numbers:

18	42	-6	25	3	-8

An algorithm is a sequence of steps to solve a problem. Ideally, an algorithm should always solve the problem correctly no matter what instance you put in.

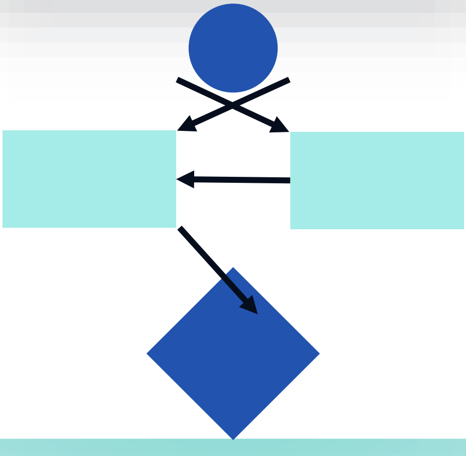
Classical algorithms

Prepare input

10110101001

Prepare a set of bits as input.

Apply logic



Perform a set of steps that transform the input into the problem solution.

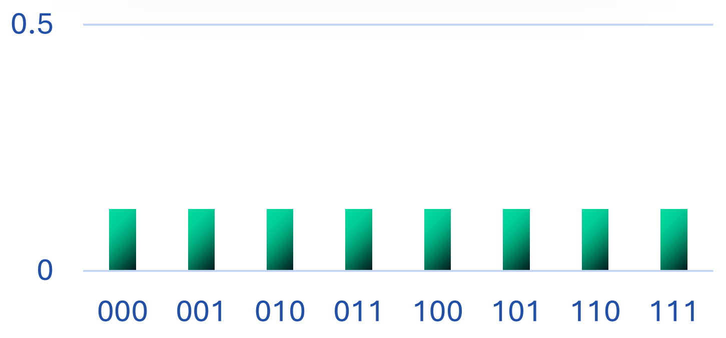
Measure outcome

00101100100

Read out the outcome.

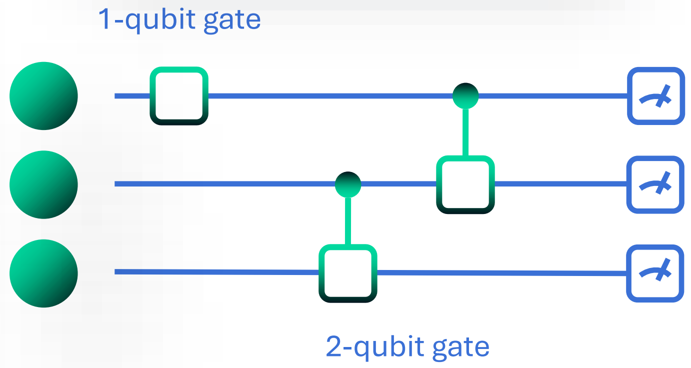
Quantum algorithms

Prepare state



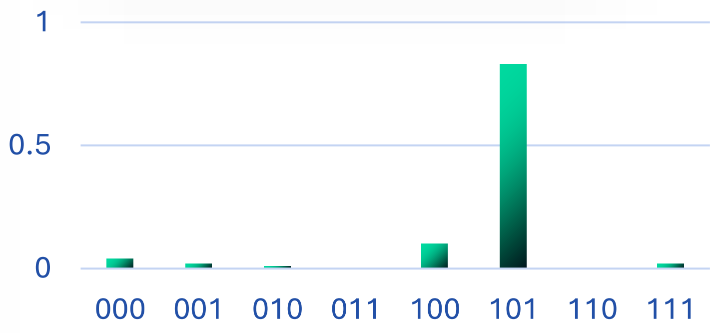
Use superposition/quantum parallelism to “process all possible inputs at once”

Apply circuit



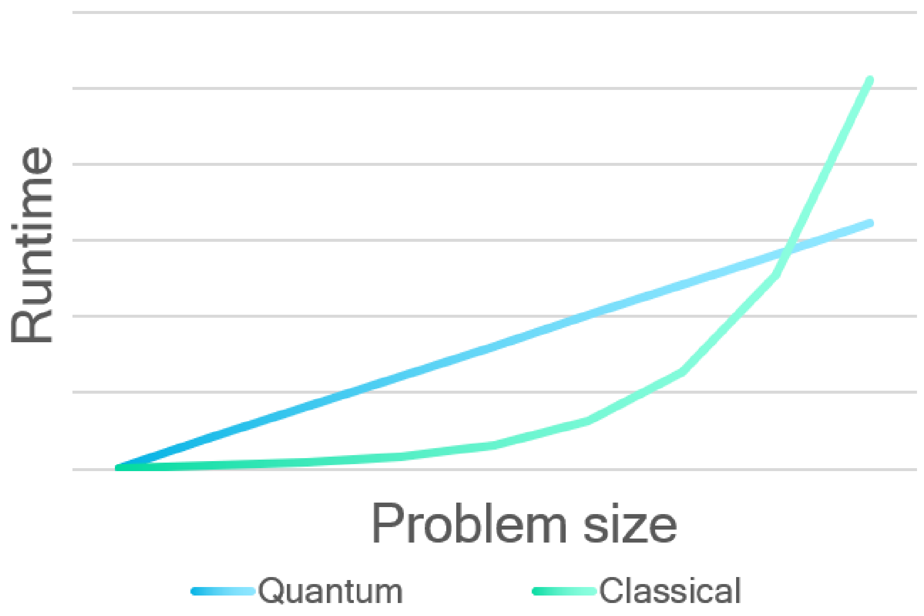
Use interference to make wanted outcomes more likely

Measure outcomes



Measure multiple times to find the most probable outcome

Quantum vs. classical



Quantum computers provide new algorithms that can be exponentially faster than classical computers at certain tasks, including simulation, optimization, or machine learning.

The landscape of quantum algorithms

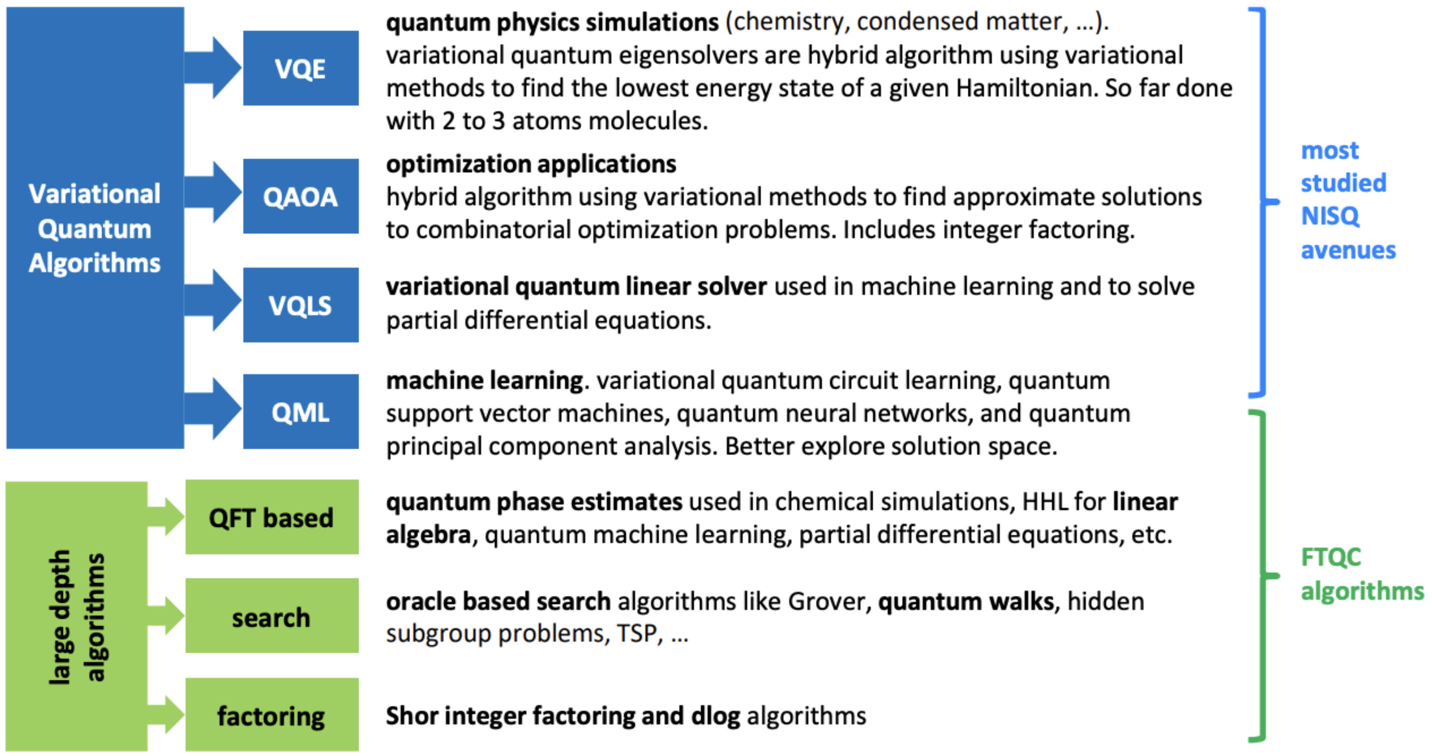


Figure 772 : the main NISQ algorithms classes as proposed by researchers and industry vendors, in blue, when those in green are specific to fault-tolerant quantum computers relying on quantum error correction. These NISQ variational algorithms should be theoretically resilient to noise and shallow but so far, they are not, particularly in a quantum advantage regime with over 50 qubits and with their current algorithm depth actual requirements. VQE algorithms scale in depth as N^6, N being their qubit numbers. Some proposed NISQ algorithms, not shown here, do not use a variational mechanism and may work better than variational algorithms. (cc) Olivier Ezratty, 2023.

Source: Olivier Ezratty (2023, Understanding Quantum Technologies)

Outline for today

1. The Bernstein-Vazirani algorithm: a quantum algorithm that is provably better than any classical one
    - See how quantum parallelism and interference work

2. Lab, part 1: Implement Bernstein-Vazirani

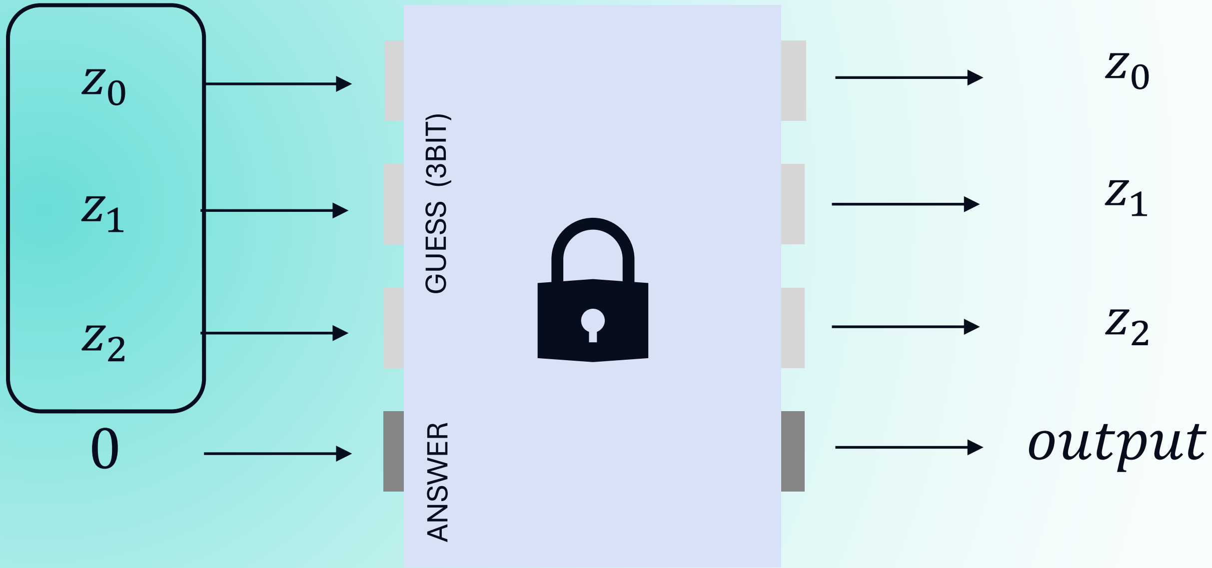
3. Variational quantum algorithms
    - QAOA: A classical-quantum hybrid algorithm that may be useful in the short term

4. Lab, part 2: Try using QAOA

A simple quantum algorithm – Bernstein-Vazirani

The problem

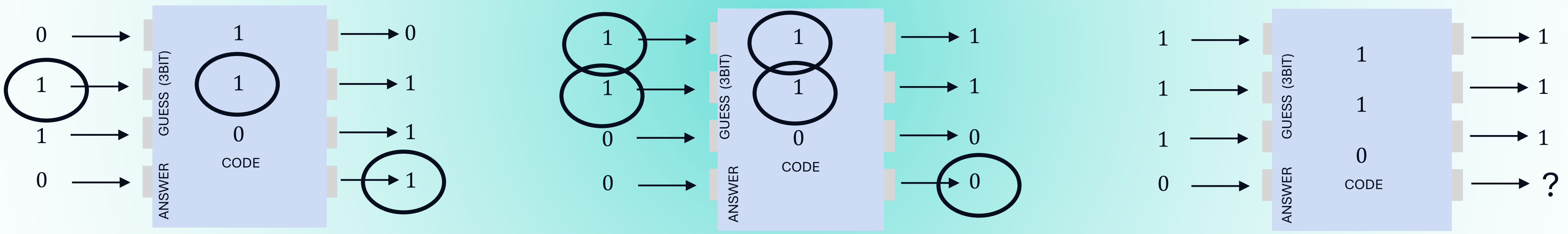
- Given: a secret (but unknown) code of n bits
- Given: a black box or "oracle"
  - Input: n "guess" qubits and one "answer" qubit
  - Oracle compares the input and the secret code:
    - Every time a 1 in the guess matches a 1 in the code, it flips the answer qubit (0 -> 1, 1 -> 0)
  - Outputs the answer qubit, doesn't change the input qubits.
- **Goal**: identify the secret code by experimenting with the oracle.



(Imagine a friend of ours made the oracle but didn't tell us the code.)

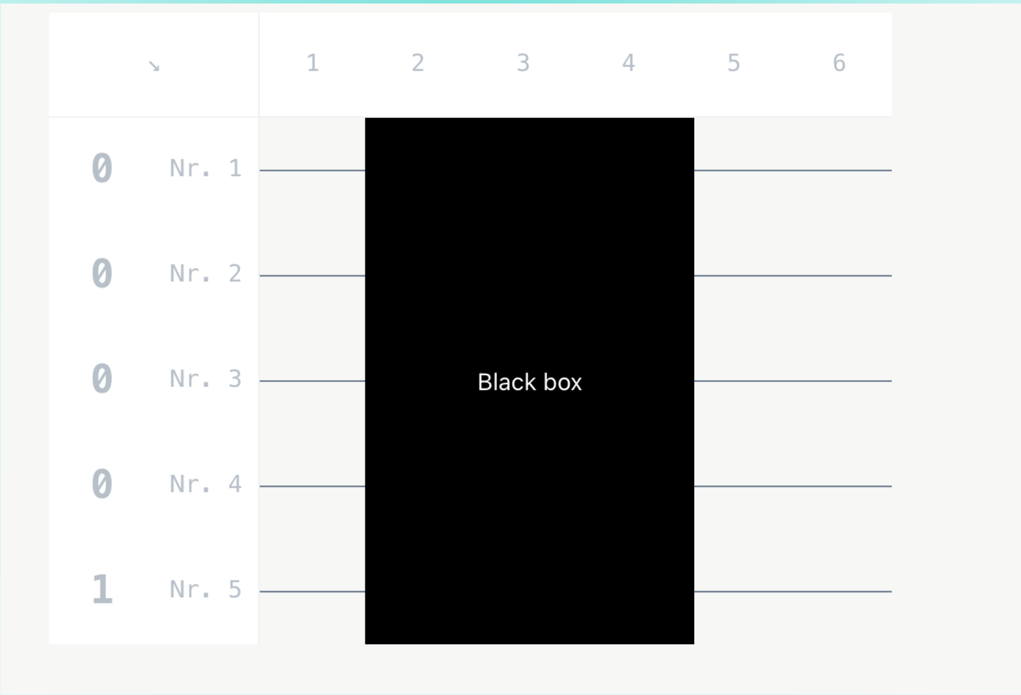
Example

Imagine the secret code is 110.



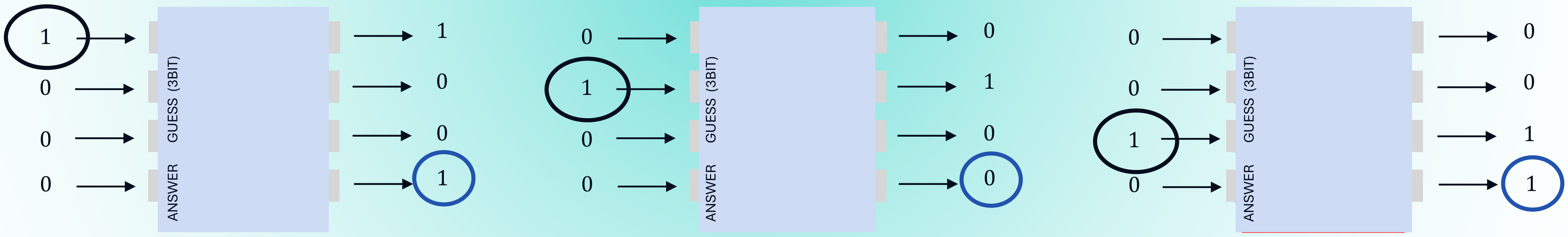
Experiment time!

Use the applet here https://tinyurl.com/iqm-secret-code or the QR code to develop a strategy for finding the secret code. How many attempts do you need to find the code? Describe your approach.



One solution

- Set one input to 1 and the rest to 0.
- The last qubit indicates if there is a 0 or 1 on this position in the secret code
- Repeat for every digit in the secret code
- With this algorithm, we need to use the oracle n times to find the code c.
  - One can **prove** that no classical algorithm will use the oracle fewer than n times.

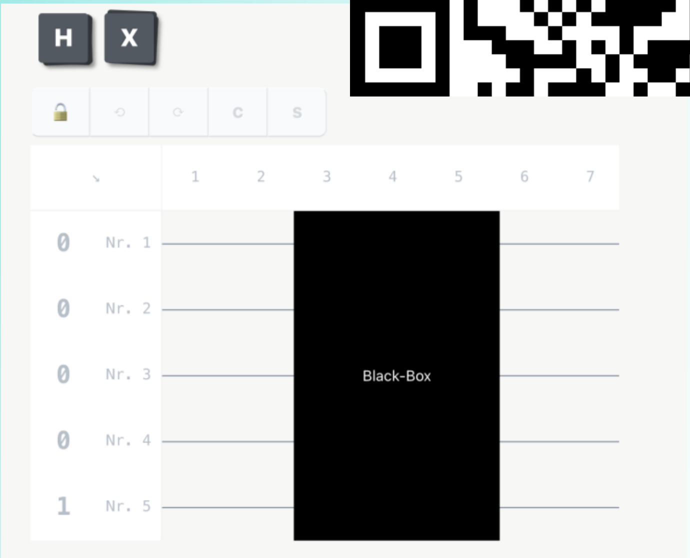


Secret code is: 101 !

Experiment time!

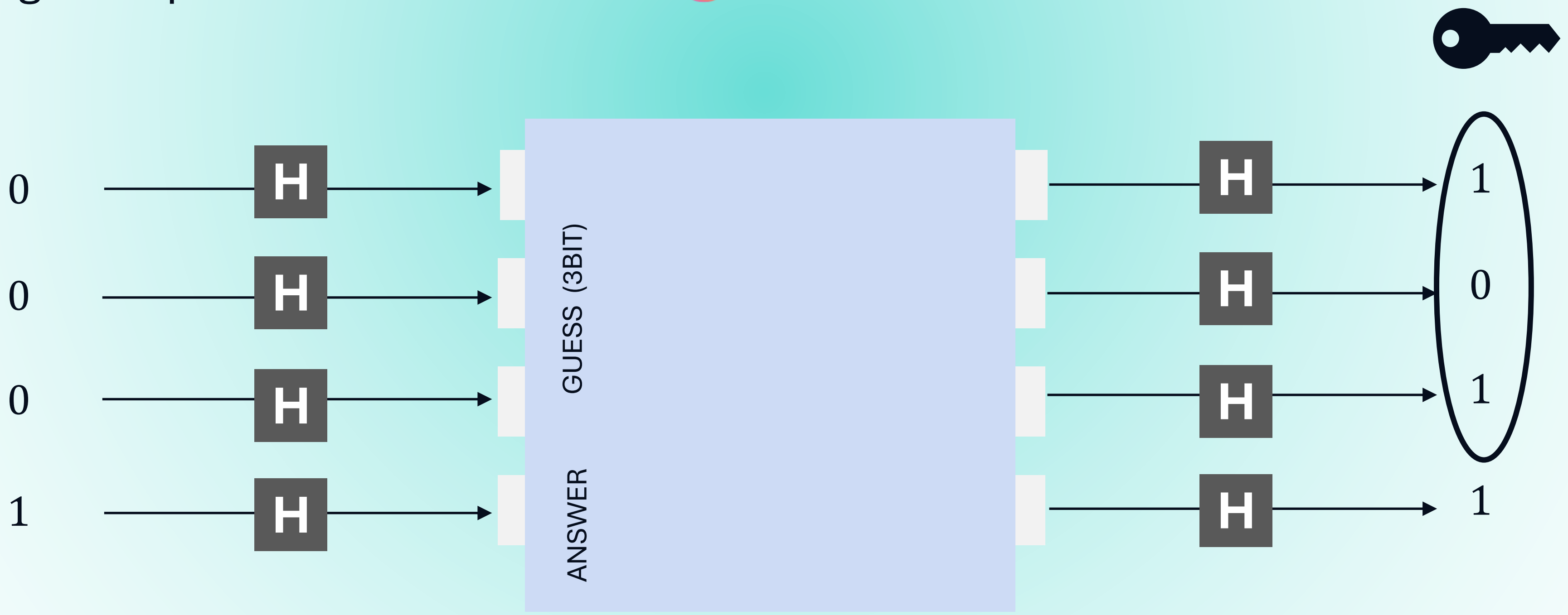
Explore the quantum solution with the applet https://tinyurl.com/iqm-solver ! To find out the secret 4-digit code of this black box, prepare an equal superposition of all 5 qubits. Also apply a set of **H** after the black box.

Afterward, open the black box and check how you can retrieve the secret code from the measurement outcomes.



Quantum solution – Bernstein-Vazirani algorithm

- Initialize guess qubits with 0, answer qubit with 1
- Apply H gates before and after the black box
- Output of guess qubits is the secret code 🤩



Attendance

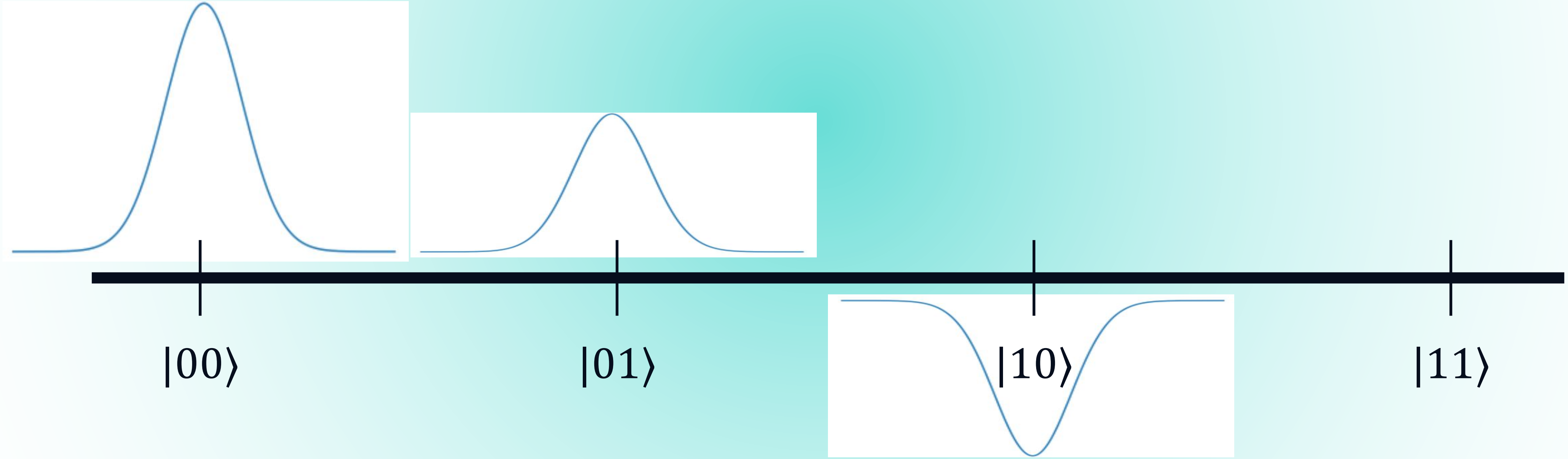
https://tinyurl.com/34f7x2v2

How does Bernstein-Vazirani work?

Pause: a visual representation of a quantum state

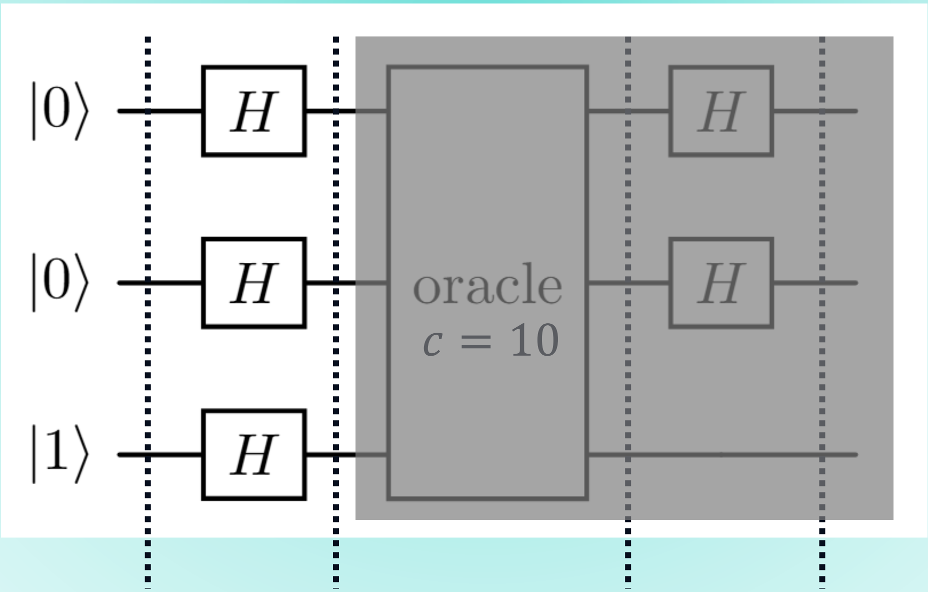
Example: represent 2-qubit state $\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{2}|01\rangle - \frac{1}{2}|10\rangle$

Draw peaks, height is just the number in front of the state.

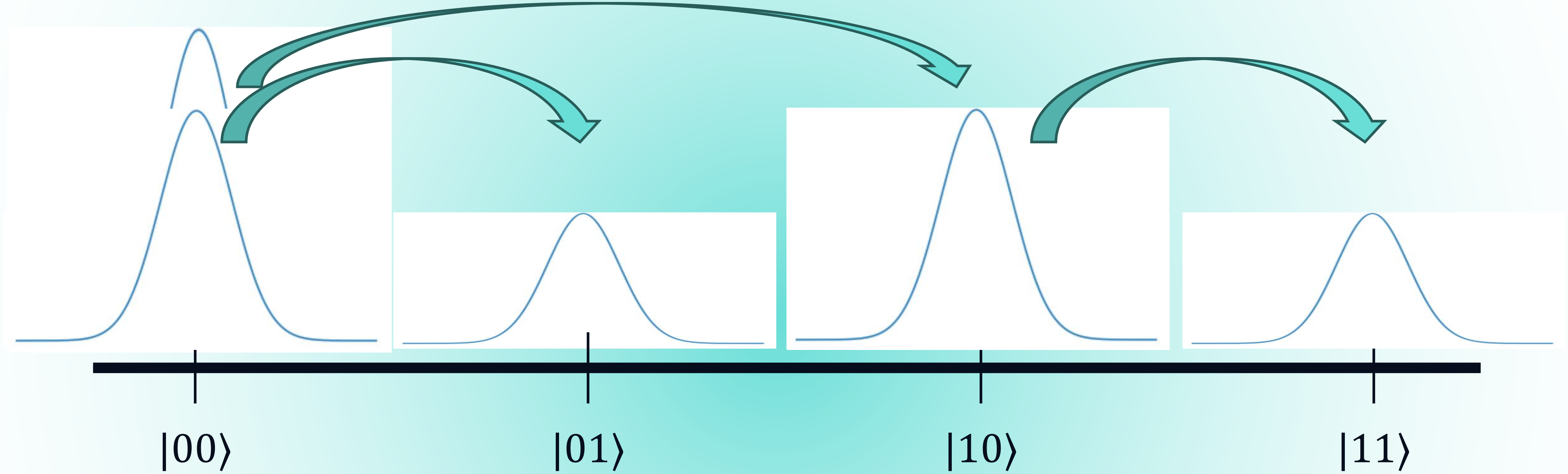


Let's unpack...

- The Bernstein-Vazirani algorithm finds c with a single query of f using interference and quantum parallelism.
- Let's see how this works for a 2-qubit example, where $c = 10$.



Step 1 – quantum parallelism

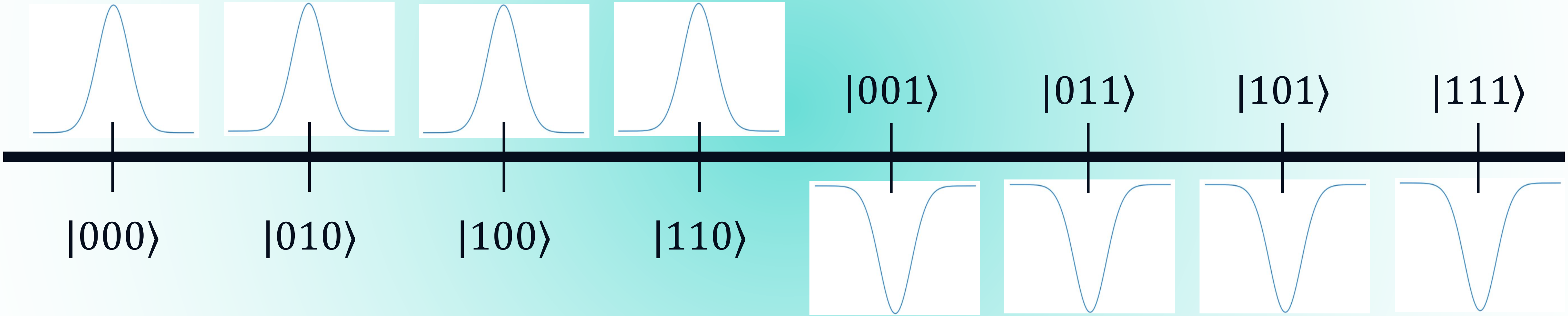


H on the first qubit

H on the second qubit

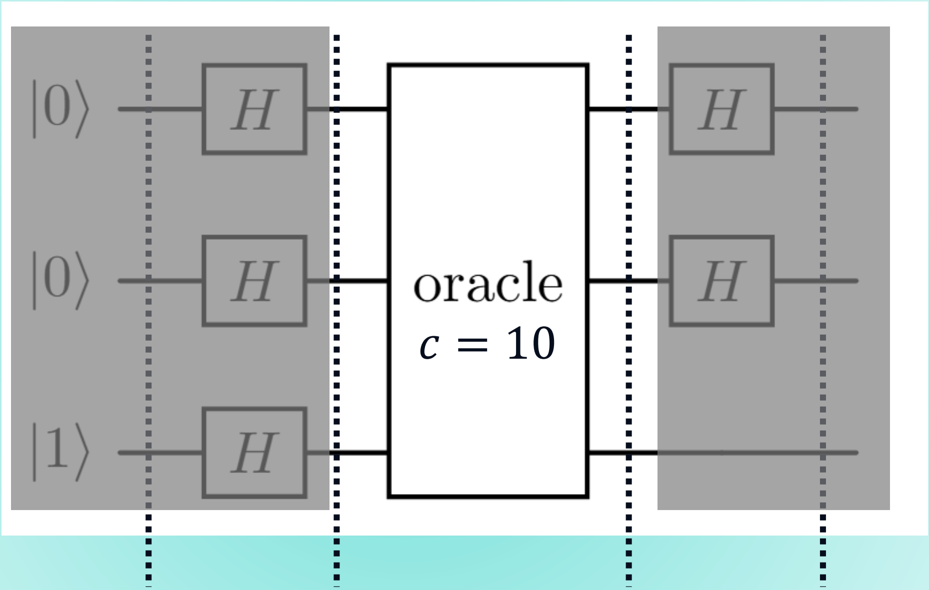
Ancilla is just $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$

Step 1 – quantum parallelism



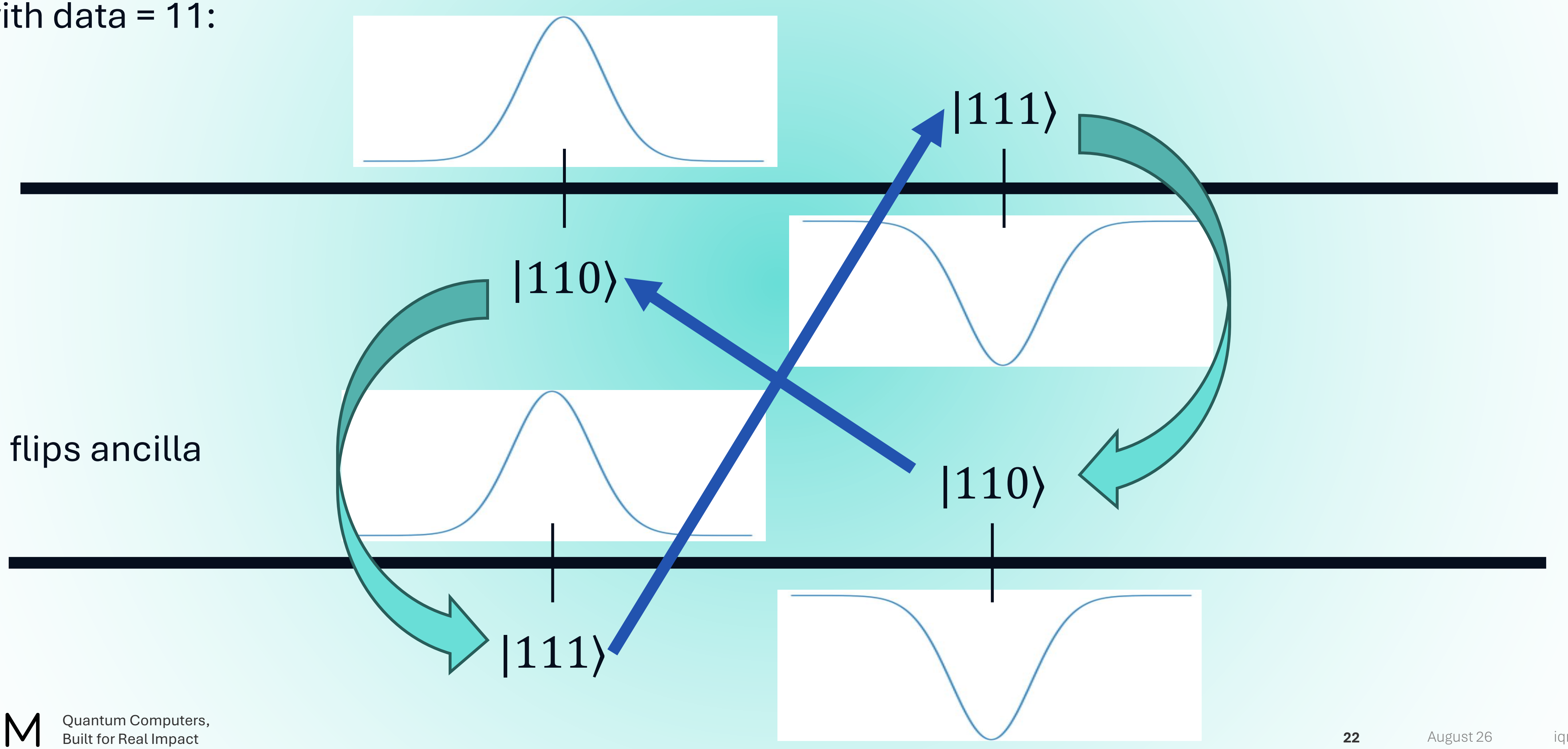
Let’s include the ancilla as the third qubit.

Step 2: Phase kickback



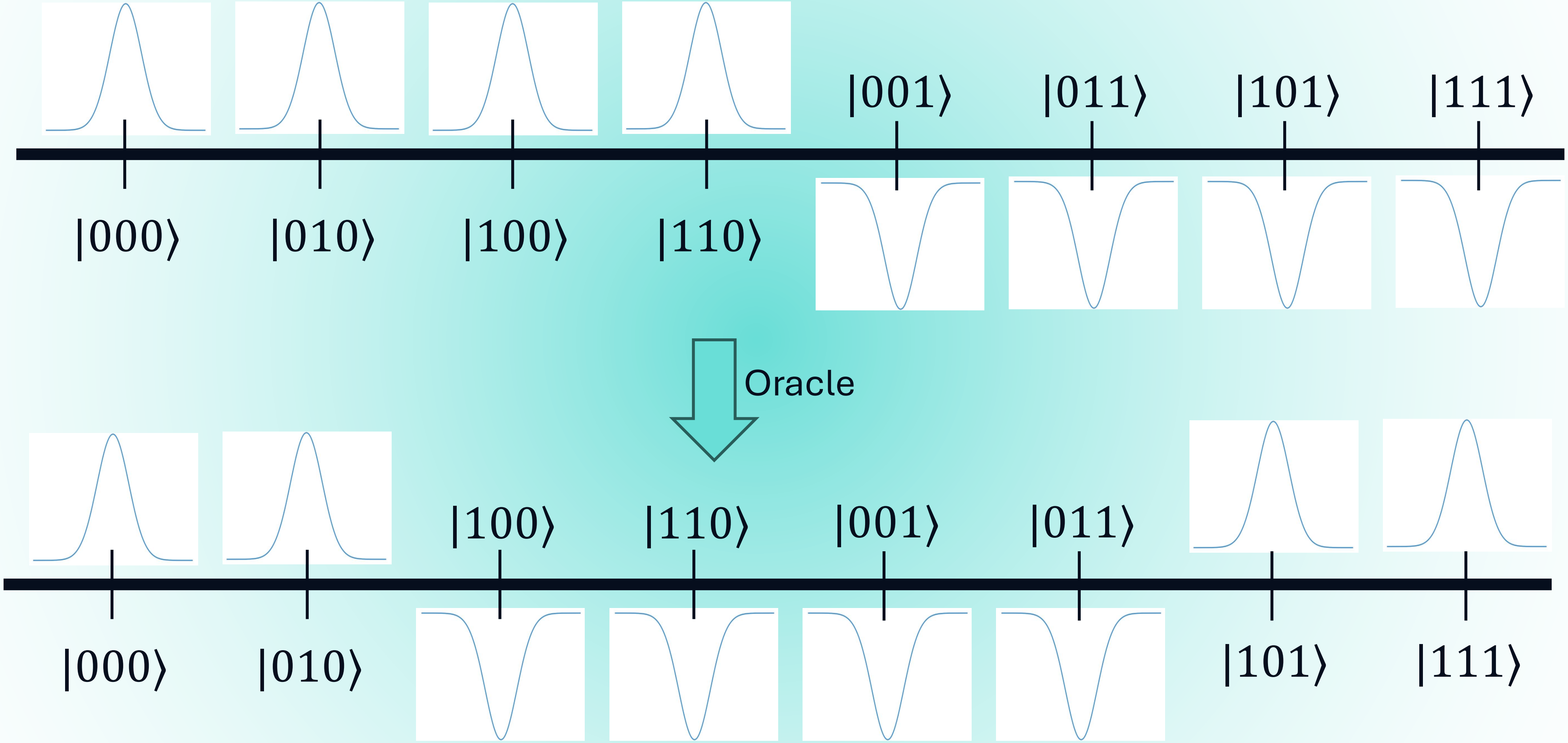
Focusing in on specific data qubits:

Part with data = 11:

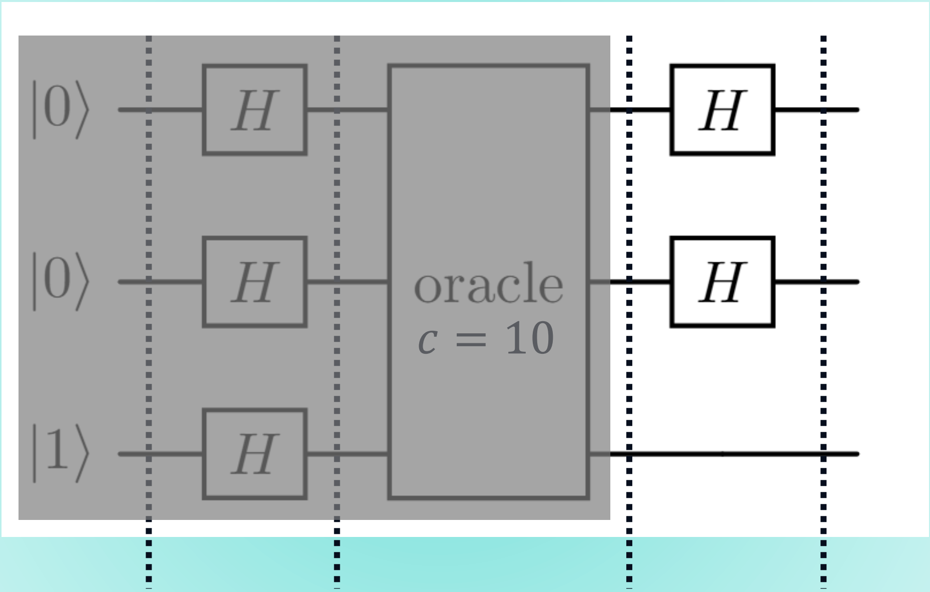


Oracle flips ancilla

Upshot: oracle flips state’s **sign** when it fires

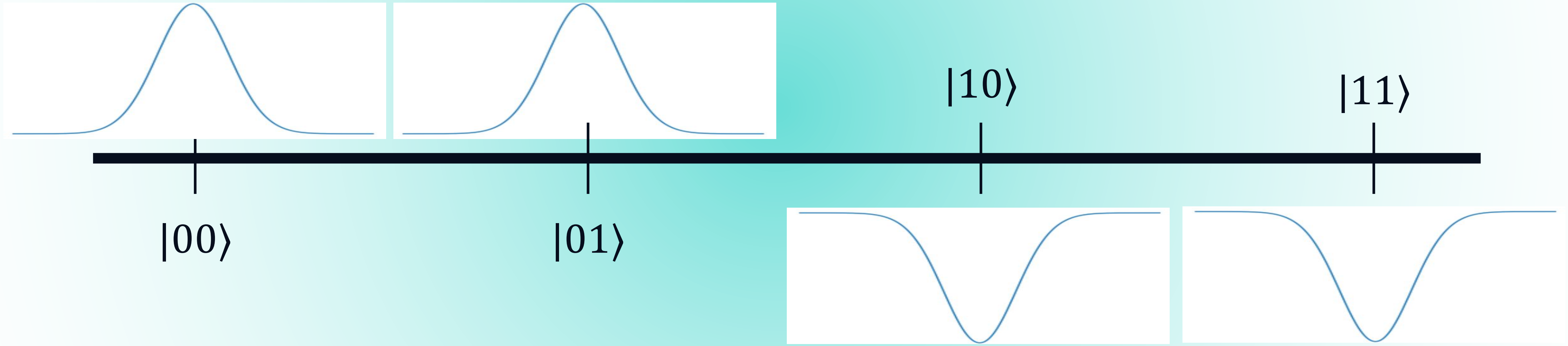


Step 3: Interference

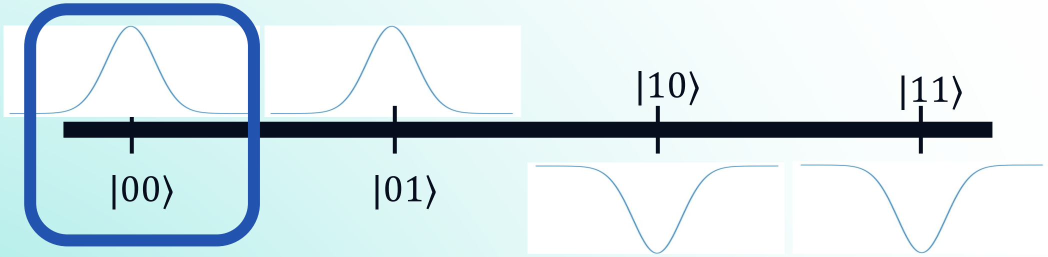


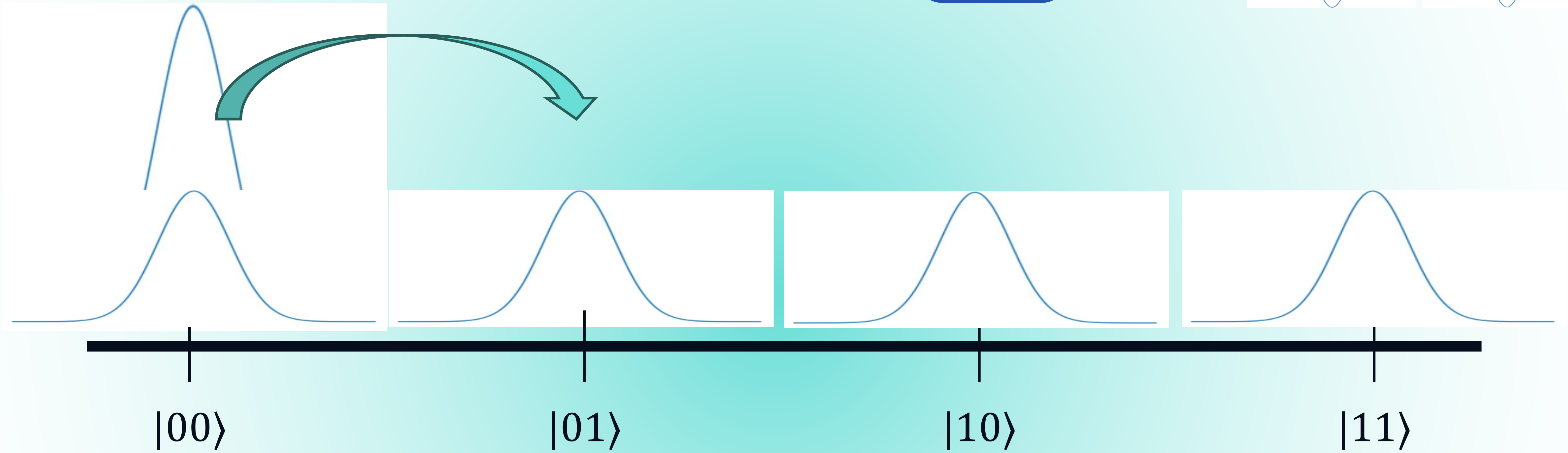
Step 3: interference

We can ignore the ancilla now.

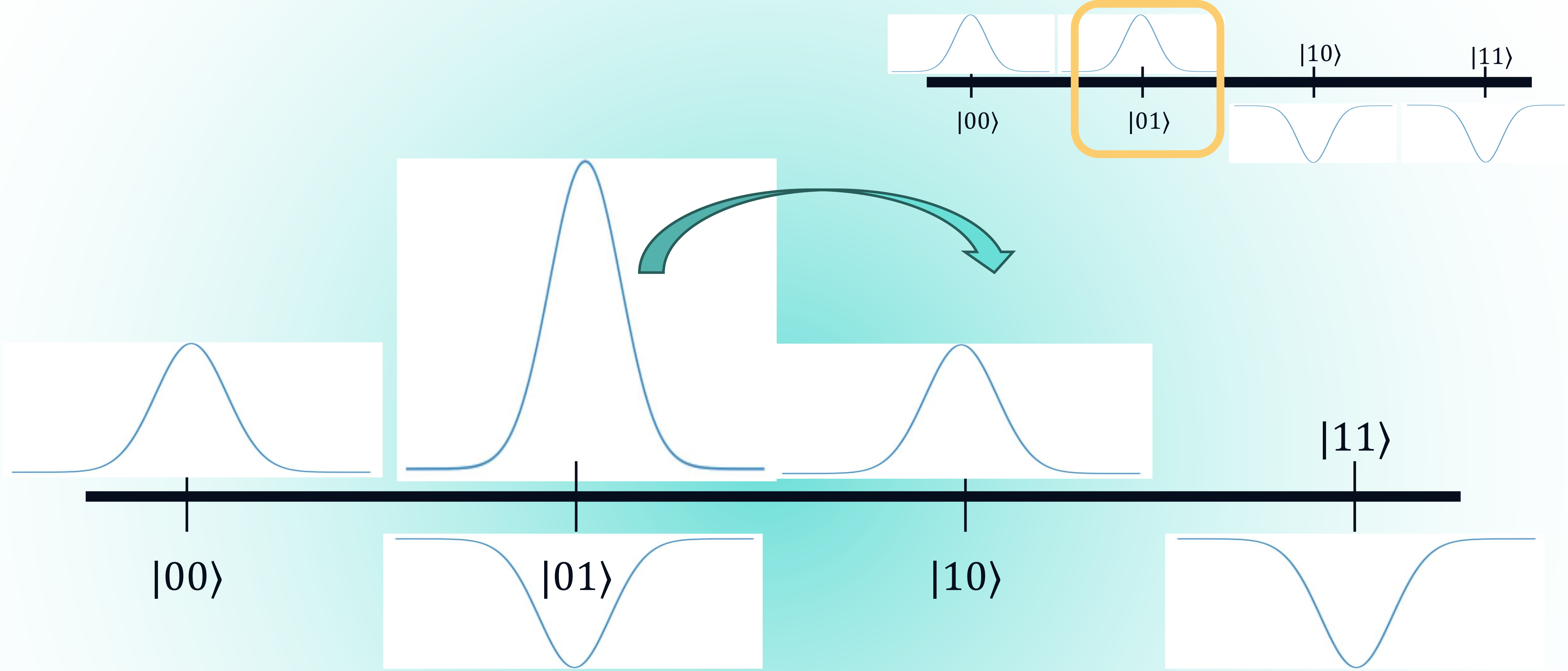


How do our two Hadamards affect each of these pieces?

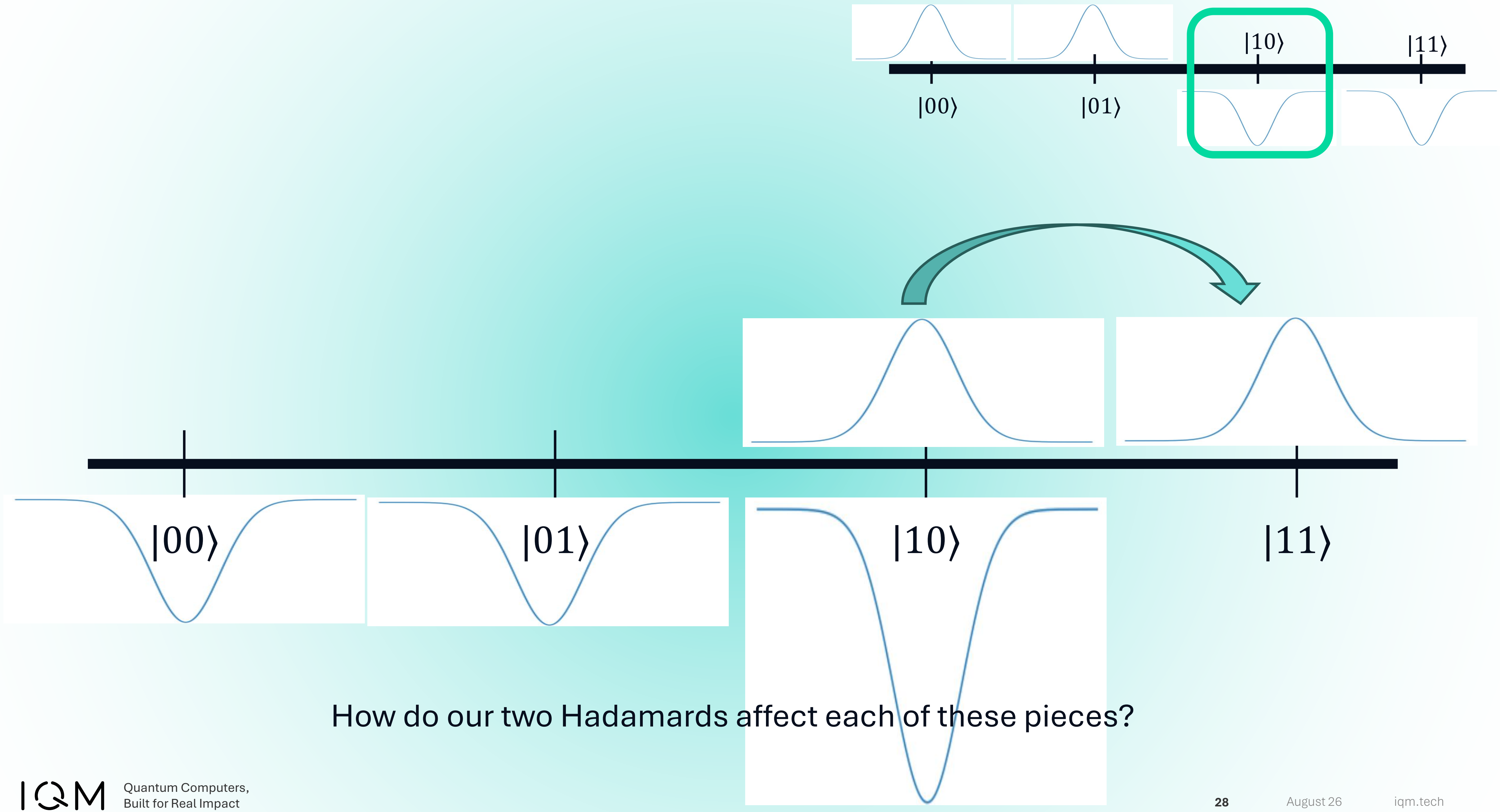




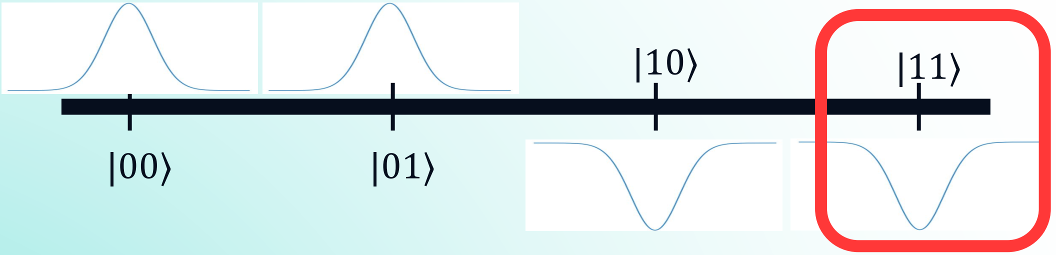
How do our two Hadamards affect each of these pieces?

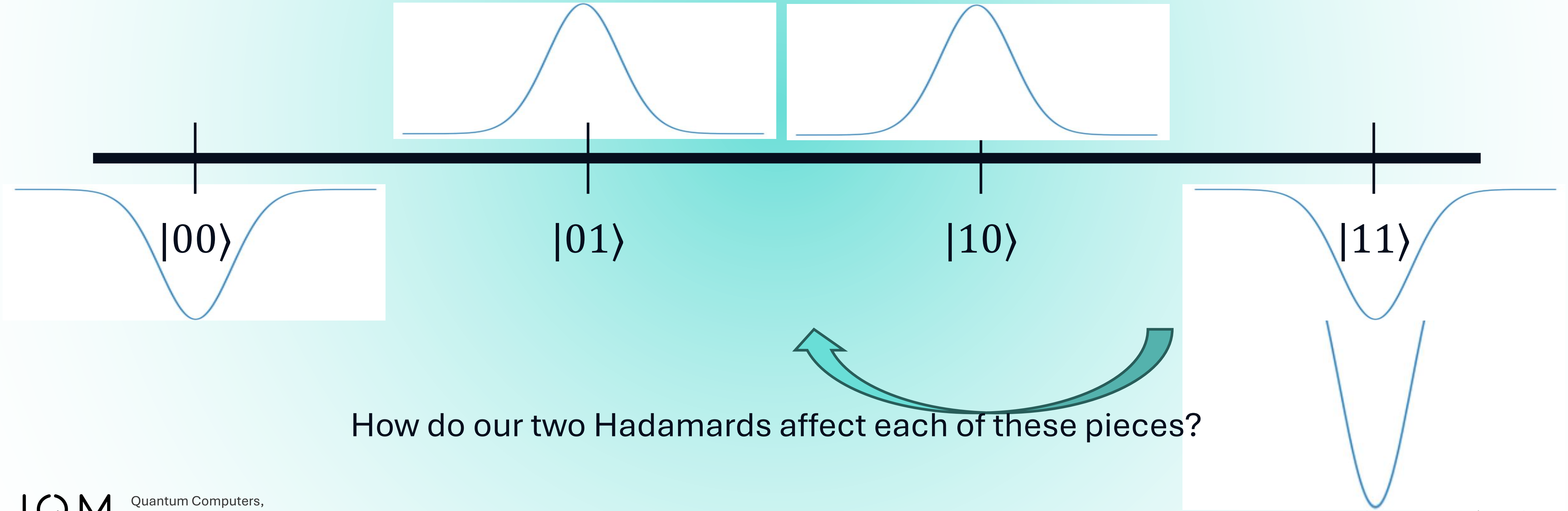


How do our two Hadamards affect each of these pieces?

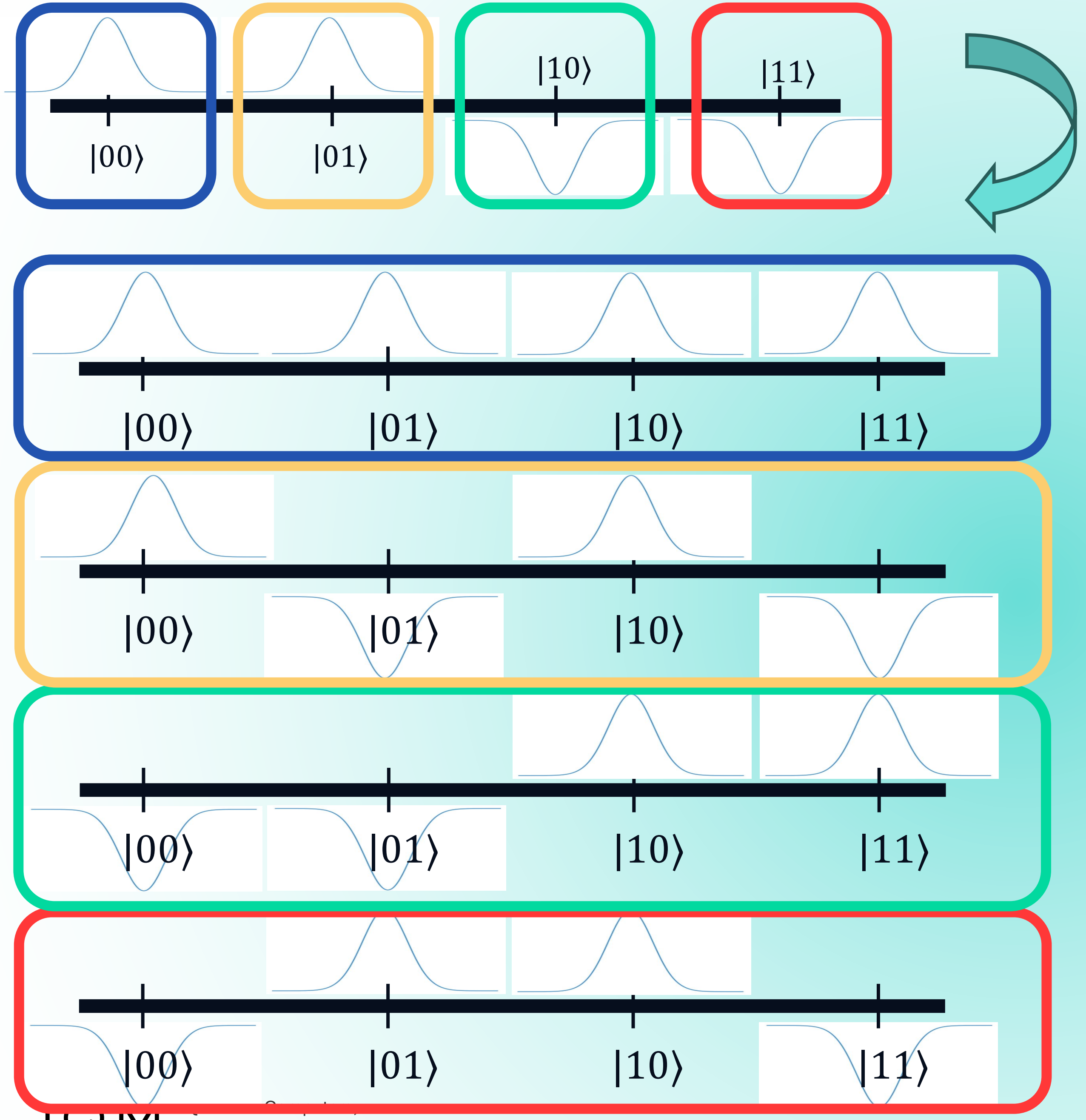


How do our two Hadamards affect each of these pieces?





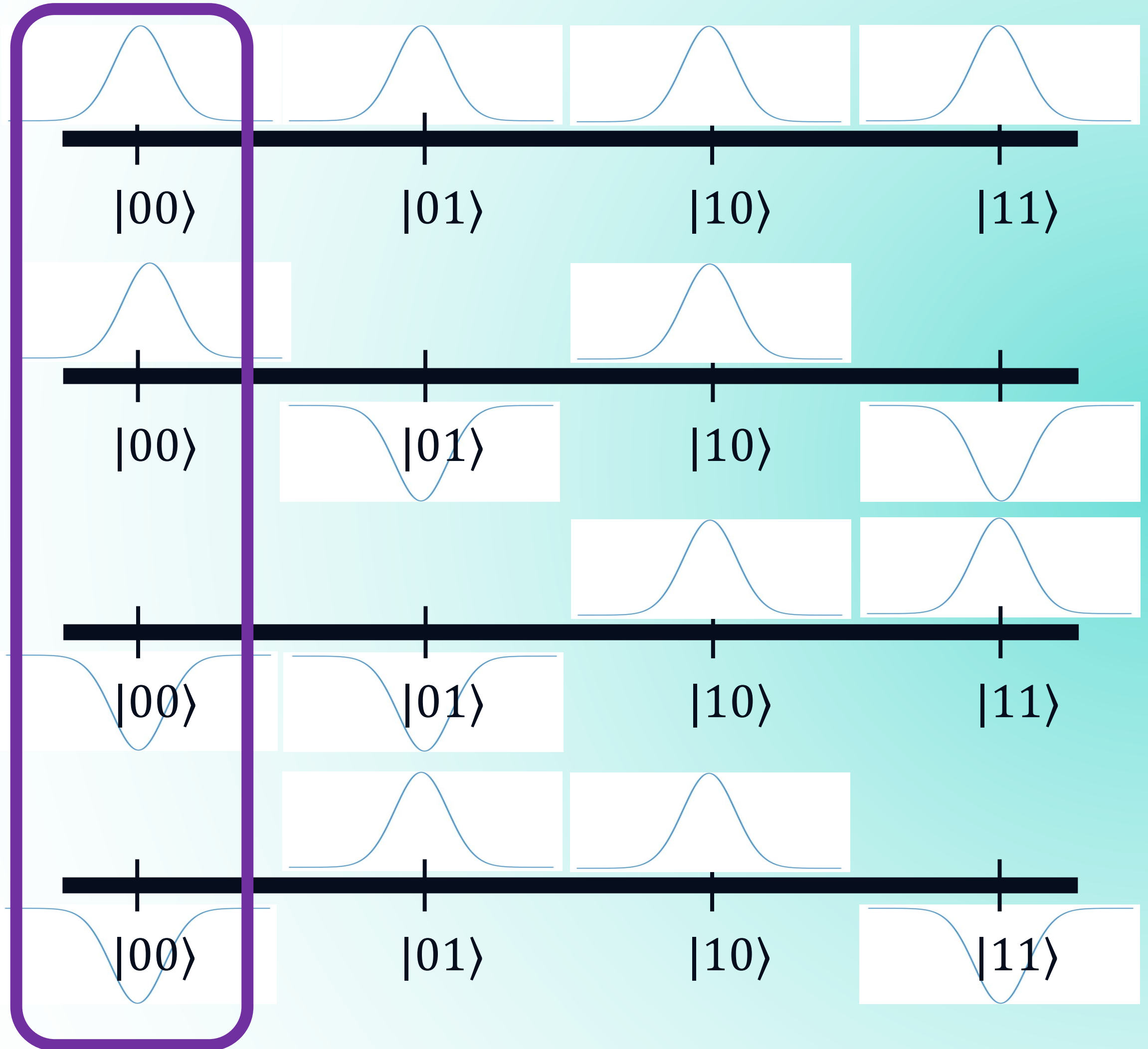
How do our two Hadamards affect each of these pieces?

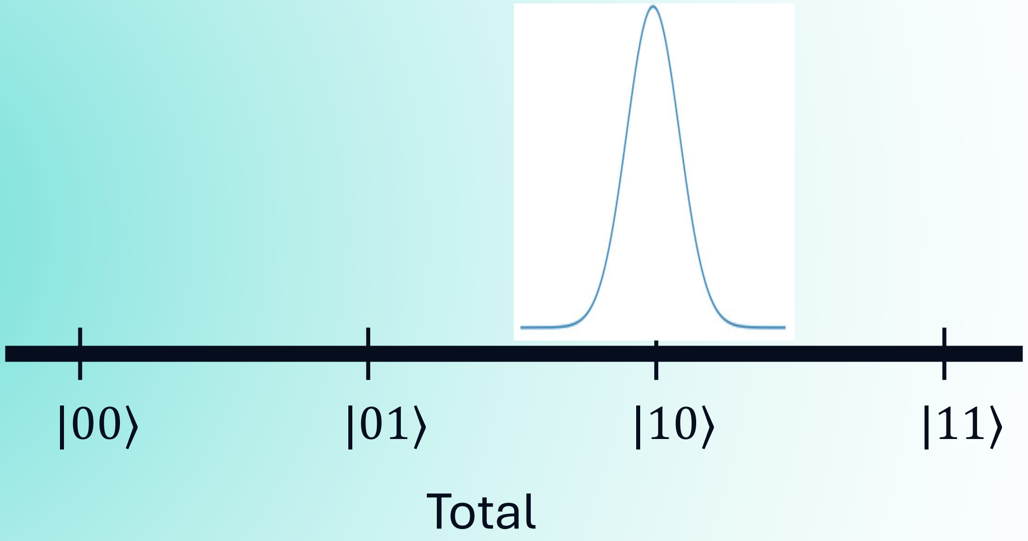


Two Hadamards

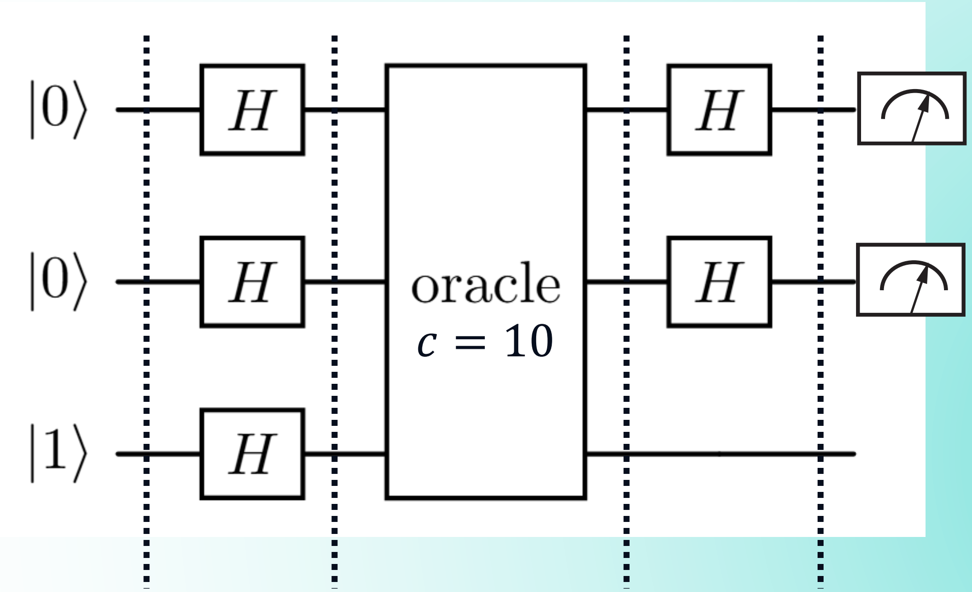
Now let’s combine the waves!

Waves interfere

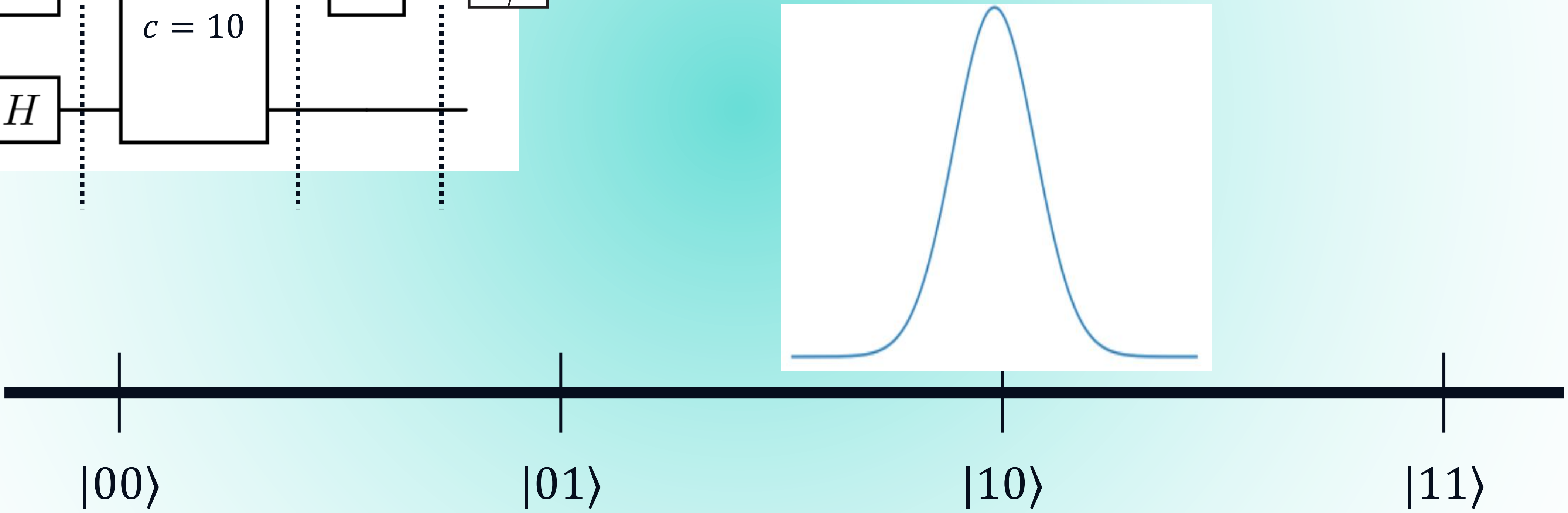




Bernstein-Vazirani algorithm - result



A single measurement of the n qubits reveals the code c!



What have we learned?

Qubits leverage superposition to represent and process multiple states concurrently.

This superposition, combined with entanglement, allows quantum algorithms to explore many possible solutions or paths in parallel within a single computation step.

Quantum interference patterns are exploited to amplify correct solutions and suppress incorrect ones.

These effects enable speedups compared to classical solutions.

Day 2 – Lab, part 1

Go to the lab notebook here:

https://tinyurl.com/QS-Algorithms-PARTICIPANT

Variational Quantum Algorithms

The variational principle

Many problems we might want to solve are expressed as a minimization (or maximization) problem:

- Find the lowest-energy configuration of electrons in a molecule
- Find a driving route that takes the least time
- Minimize the risk in a stock portfolio

The quantity we want to minimize is called a **cost function**

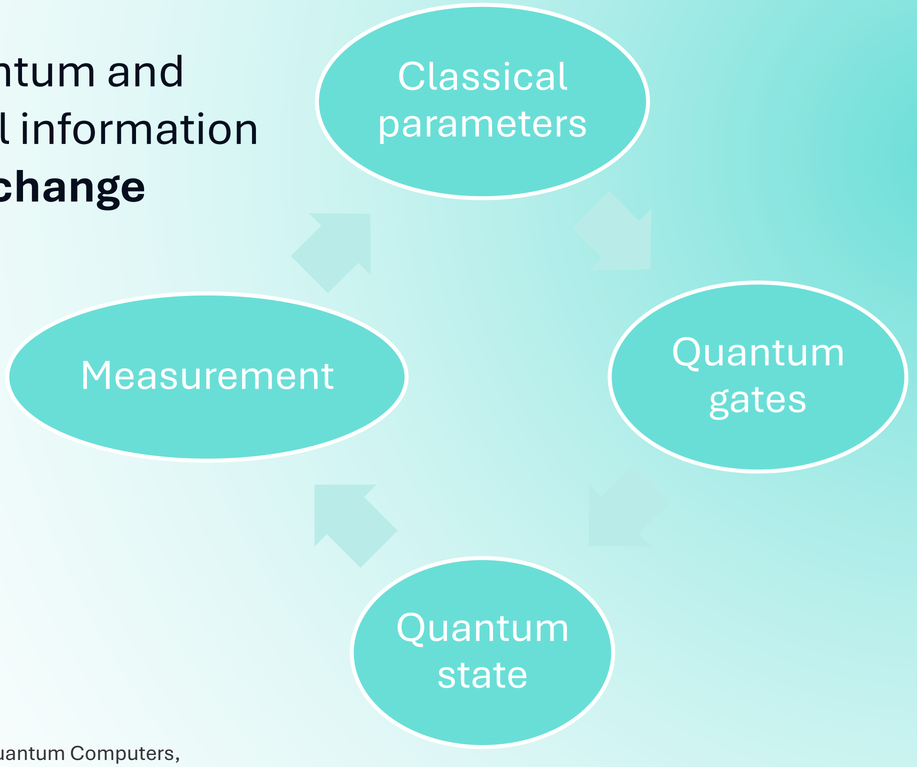
The variational principle says:

1. Make a (educated) guess for the solution
2. Calculate the cost function
3. The minimum of the cost function is a *good approximation* of the true solution

Variational algorithms

Use variational principle to find the best possible approximation of the true solution of our problem

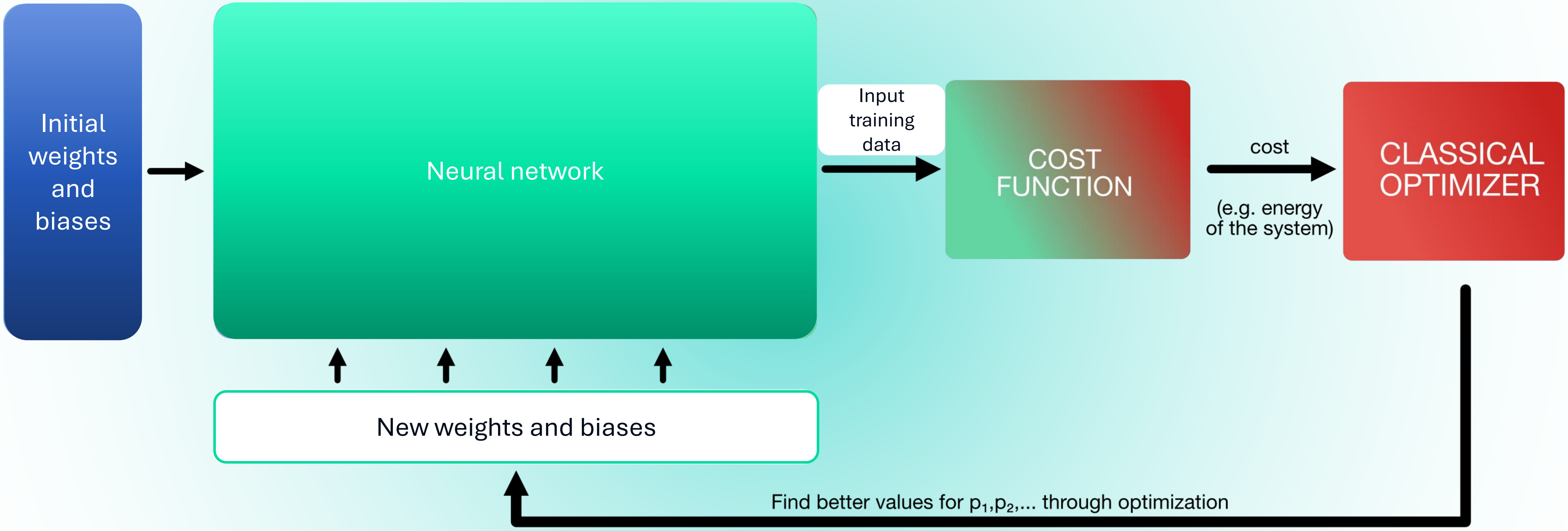
Quantum and classical information **exchange**



Why is this interesting for **NISQ**?

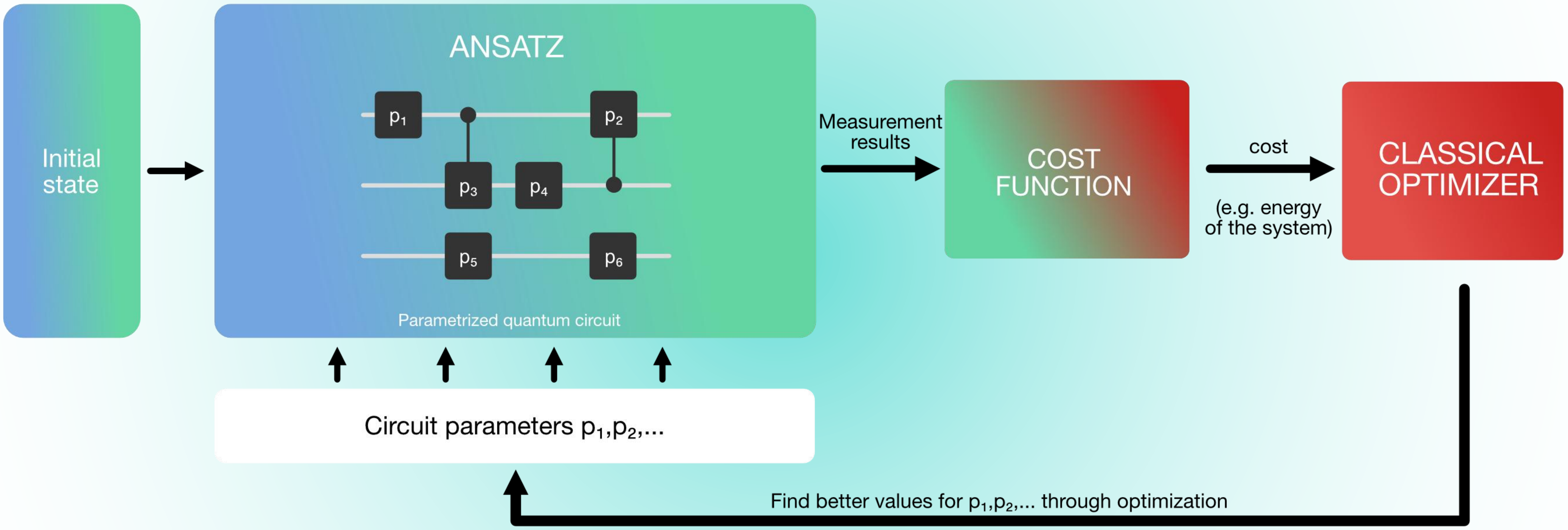
- Hybrid computing resources
- Noise-resilient

Modular structure: an analogy



Classical machine learning

Modular structure: an analogy

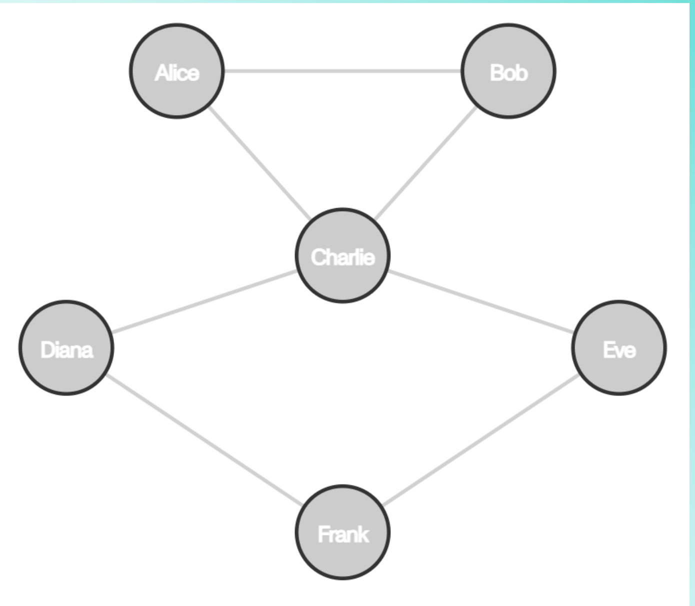


Variational quantum algorithm

The cost function

Max-Cut problem

- You are proctoring an exam and need to divide the students into two rooms.
- Students who are friends with each other and in the same room are more likely to cheat (Assume you know which students are friends)
- **Goal**: Assign each student a room to minimize the chance of cheating, that is, split up as many friendships as possible.



- Each dot (node) is a student.
- Each line (edge) is a friendship.
- NP-hard problem in general!

Activity 1

Use the applet here
https://tinyurl.com/iqm-max-cut
to play with the Max-Cut problem.

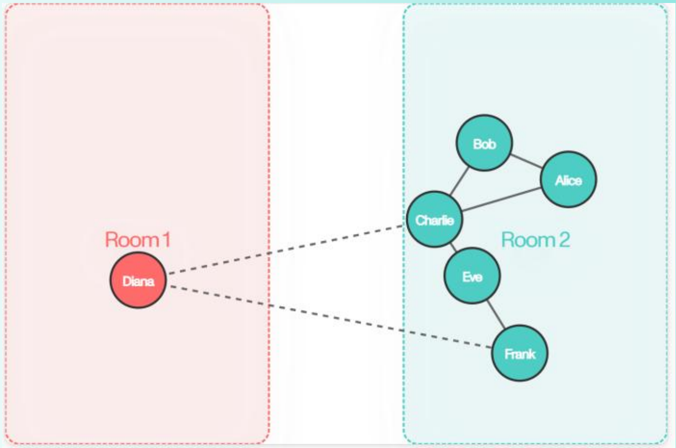
Remember, the cost function takes as input:

- The list of friendships
- A choice of room for each student

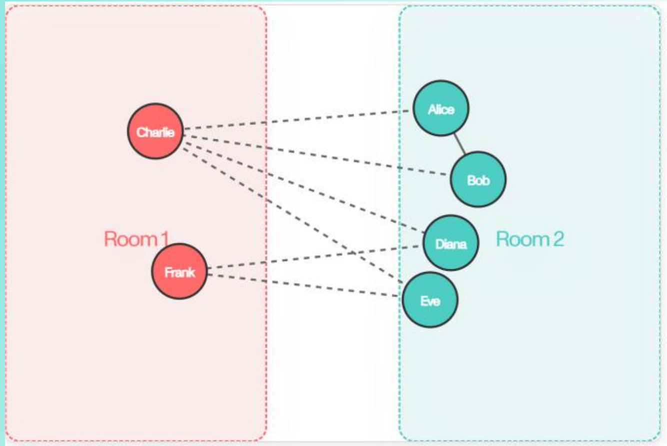
Can you figure out a way to encode the problem into a cost function?

Activity 1

Can you figure out a way to encode the problem into a cost function?



Cost = (5 unbroken) - (2 broken)
= **+3**



Cost = (1 unbroken) - (6 broken)
= **-5**

A cost function for Max-Cut

- Assign each student a qubit. $|0\rangle$ means that student is in room 0 and $|1\rangle$ means the student is in room 1.

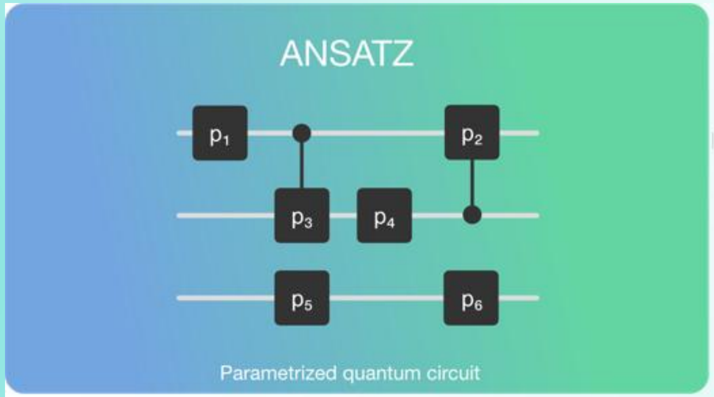
- Define a cost function: $H(Z_i) = \sum_{i,j} \begin{cases} 0 \text{ if } i,j \text{ are not friends} \\ Z_i Z_j \text{ if } i,j \text{ are friends} \end{cases}$

  $Z_i = 1$ if qubit i is in state $|1\rangle$
  $Z_i = -1$ if qubit i is in state $|0\rangle$

  - Friends in the same room contribute +1 to the cost function
  - Friends in different room contribute -1 to the cost function
  - $H(Z_i)$ is minimized when the number of split-up friendships is maximal.

- Turns out: this cost function is easy to calculate with a quantum computer!

The ansatz



Goal of a variational quantum algorithm

Find a **quantum circuit** that transforms some given initial state into the (approximate) solution of the minimization problem

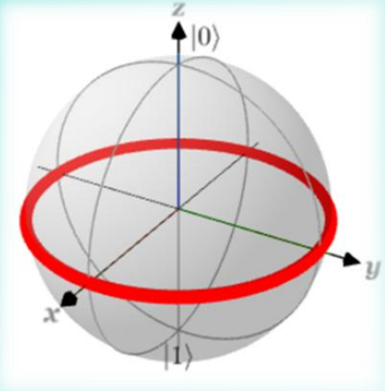
Infinitely many quantum circuits (and initial states) are possible! **Can’t try them all…**

An ansatz is a collection of guesses for the solution to a problem.

- For us, an Ansatz is a collection of quantum states (i.e. a collection of quantum circuits applied to some initial state)
- It can be a random guess or an informed guess

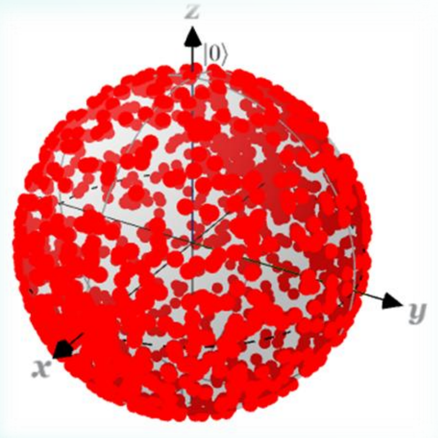
Example

θ_2 is a classical parameter – can be any angle.

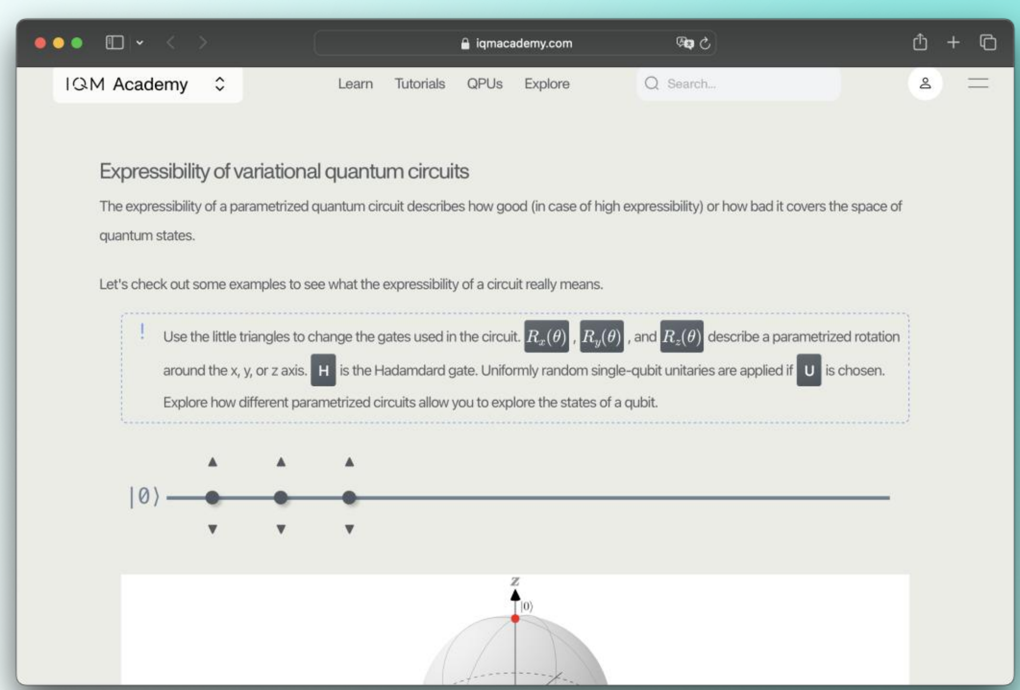


What if the answer to our problem lies outside this circle?

Activity 2: Expressibility



Expressibility is a measure of the power of an Ansatz to generate a uniform set of states in the space of quantum states.



Try it yourself at
https://tinyurl.com/iqm-expressibility

The optimizer

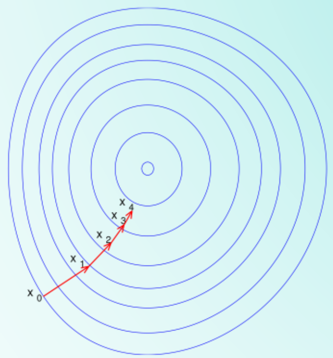
What is the role of the optimizer?

It determines how to adjust Ansatz parameters to decrease the cost function

Gradient-based methods

$$\boldsymbol{\theta}_{i+1} = \boldsymbol{\theta}_i - \eta \boxed{\nabla C(\boldsymbol{\theta}_i)}$$

The gradient tells us how to change the parameters to make C decrease as fast as possible. Think "*go straight down the mountain.*"



Gradient-free methods

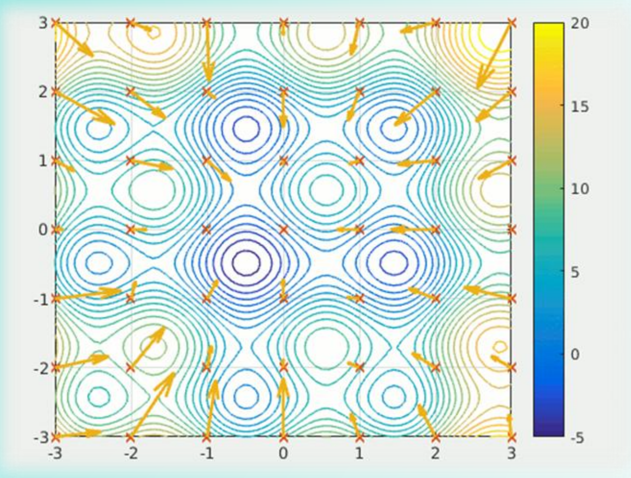
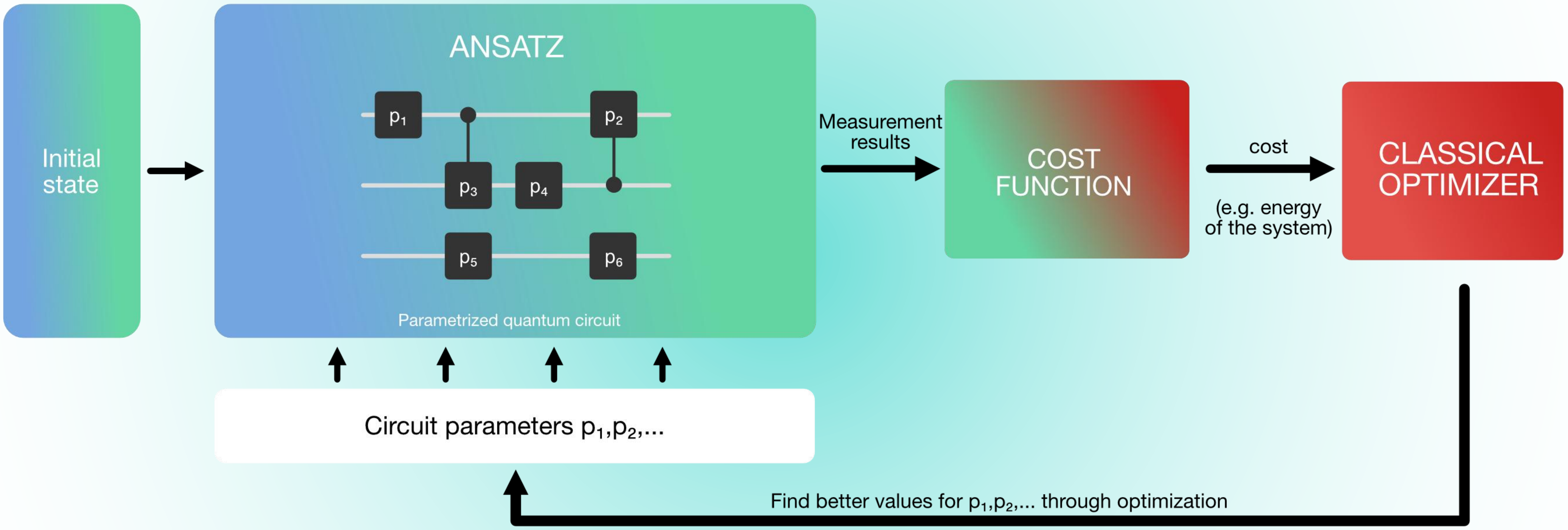


Image sources: Wikipedia

Example: particle swarm optimization

Let's put all the pieces together!



Variational quantum algorithm

What have we learned?

- Variational quantum algorithms are hybrid quantum-classical algorithms based on the variational principle

- Their key components are:

- Quantum Circuit (Ansatz): Encodes the problem into a quantum state.
- Cost Function: Represents the problem's objective, evaluated on a quantum computer.
- Classical Optimizer: Updates parameters to minimize (or maximize) the cost function.

- We can run variational quantum algorithms already today on NISQ quantum hardware for different applications.

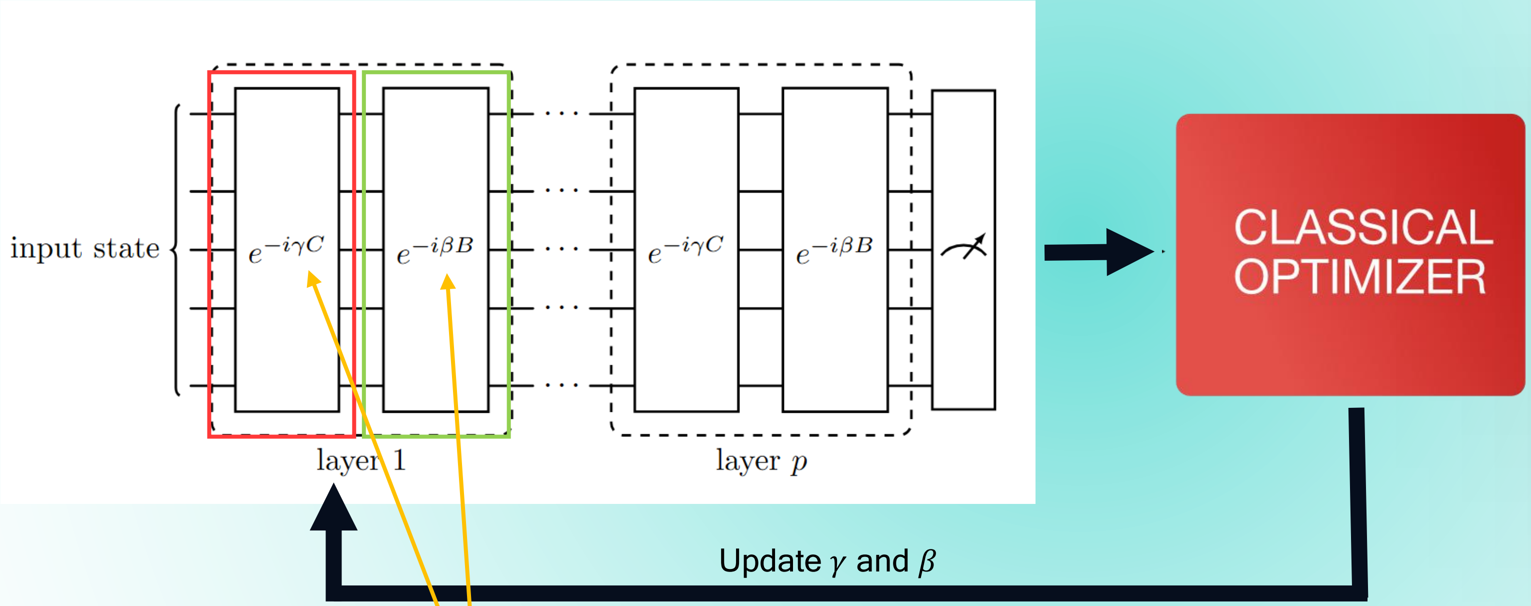
Day 2 - Lab

Go to the lab notebook here:

https://tinyurl.com/QS-Algorithms-PARTICIPANT

Quantum Approximate Optimization Algorithm (QAOA)

Solve combinatorial optimization problems. How does it work?



- Fix the number of layers p
- γ and β are the ansatz parameters
- C is a circuit implementing the cost function
- B is a "mixer" operator that adds "quantumness"

Example applications

Max-Cut

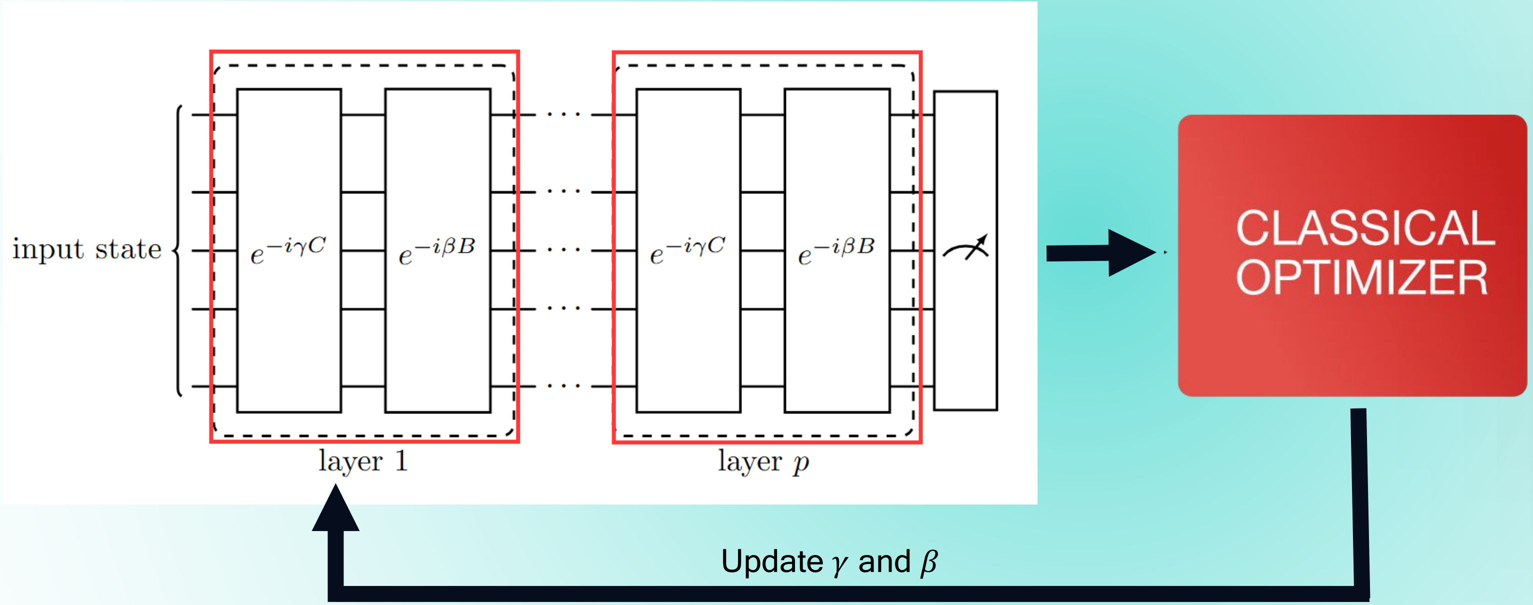
Portfolio optimization

Scheduling

Learn more in lab!

Quantum Approximate Optimization Algorithm (QAOA)

Solve combinatorial optimization problems. How does it work?



- **Fix the number of layers p**
- **γ and β are the ansatz parameters**
- **C is a circuit implementing the cost function**
- **B is a “mixer” operator that adds “quantumness”**

Example applications

Max-Cut

Portfolio optimization

Scheduling

Learn more in lab!