Chapter 4: Quantum Computing from First Principles

Classical optimization, as we saw in Chapters 1–3, is about choosing the best option from a search space. Quantum optimization keeps that goal, but changes the physical and mathematical language used to explore possibilities.

Before we can understand variational quantum algorithms, QAOA, VQE, or quantum annealing, we need to understand what information looks like inside a quantum computer.

This chapter builds that foundation carefully. We will begin with the smallest unit of quantum information, the qubit, and then move toward the ingredients needed for variational algorithms:

- quantum states,
- amplitudes,
- superposition,
- measurement,
- the Born rule,
- tensor products,
- entanglement,
- quantum gates,
- circuits.

The goal is not to cover all of quantum mechanics. Instead, the goal is to understand the part of quantum computing that variational quantum algorithms use every day.

Quantum computing is often described with dramatic phrases: “many worlds,” “parallel universes,” or “trying all answers at once.” These phrases are usually more confusing than helpful. A quantum computer is not simply a magical parallel classical computer. Its power comes from a precise combination of complex amplitudes, unitary transformations, interference, entanglement, and measurement. Standard quantum computing texts describe these ideas mathematically using finite-dimensional complex vector spaces and linear algebra, which is the approach we will use here (Nielsen and Chuang, 2010).

4.1 From Classical Bits to Quantum Bits

A classical computer stores information using bits.

A bit is a variable that can take one of two values:

$$0 \quad \text{or} \quad 1.$$

For example:

- A light switch may be off or on.
- A yes/no answer may be encoded as 0 or 1.
- A binary decision variable in optimization may represent whether an asset is selected, whether a task is assigned, or whether an edge is cut.

In classical optimization, binary variables are extremely important. A problem such as

$$x_i \in \{0,1\}$$

means that each decision variable x_i must be either 0 or 1.

A quantum computer also has a two-level information unit, but it behaves differently.

A qubit, short for quantum bit, is the basic unit of quantum information. When measured in the standard computational basis, a qubit produces either outcome 0 or outcome 1. But before measurement, its state can be a combination of the two basis states.

The two standard basis states are written as

$$|0\rangle$$

and

$$|1\rangle.$$

This notation is called ket notation or Dirac notation. It is widely used in quantum mechanics and quantum computing (Nielsen and Chuang, 2010).

You can think of $|0\rangle$ and $|1\rangle$ as the quantum analogues of classical 0 and 1. But a qubit is not restricted to being only one of them before measurement.

A general qubit state has the form

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle.$$

Here:

- $|\psi\rangle$ is the state of the qubit.
- α and β are complex numbers.
- α is the amplitude of $|0\rangle$.
- β is the amplitude of $|1\rangle$.

The amplitudes must satisfy

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

This condition ensures that total measurement probability is 1.

For example,

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

is a valid qubit state because

$$\left|\frac{1}{\sqrt{2}}\right|^2 + \left|\frac{1}{\sqrt{2}}\right|^2 = \frac{1}{2} + \frac{1}{2} = 1.$$

Another valid state is

$$|\varphi\rangle = (1) \square (2)|0\rangle + \text{frac}\sqrt{(3)}2|1\rangle,$$

because

$$\leq\text{ft}|(1) \square (2)|\square + \leq\text{ft}|\text{frac}\sqrt{(3)}2|\square = (1) \square (4) + (3) \square (4) = 1.$$

The key new idea is that the state of a qubit is described by amplitudes, not directly by ordinary probabilities.

4.2 Amplitudes Are Not Probabilities

A probability is a nonnegative real number between 0 and 1. If an event has probability 0.7, then it is expected to occur about 70% of the time over many repeated trials.

An amplitude is different. An amplitude can be positive, negative, or complex.

A complex number has the form

a + bi,

where a and b are real numbers and

$$i^2 = -1.$$

For example,

frac1√(2), -frac1√(2), fraci√(2)

can all appear as amplitudes.

Probabilities are obtained from amplitudes by taking the squared magnitude. For a qubit

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

the probability of measuring 0 is

$$P(0) = |\alpha|^2,$$

and the probability of measuring 1 is

$$P(1) = |\beta|^2.$$

This connection between amplitudes and probabilities is known as the Born rule, one of the basic rules of quantum theory (Nielsen and Chuang, 2010).

For example, suppose

$$|\psi\rangle = \text{frac1}\sqrt{(3)}|0\rangle + \sqrt{((2)\ \square\ (3))}|1\rangle.$$

Then

$$P(0) = \leq\text{ft}|\text{frac1}\sqrt{(3)}|\square = (1)\ \square\ (3),$$

and

$$P(1) = \leq\text{ft}|\sqrt{((2)\ \square\ (3))}|\square = (2)\ \square\ (3).$$

If we prepare this same state many times and measure it each time, we expect to see outcome 0 about one third of the time and outcome 1 about two thirds of the time.

This “many times” phrase is important. A single measurement gives one classical outcome. It does not reveal the full quantum state.

That is why variational quantum algorithms require repeated measurements. They prepare a quantum state, measure many times, estimate an average cost, and then send that estimate to a classical optimizer.

4.3 Superposition: A State with Multiple Basis Components

A qubit such as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

is said to be in a superposition of $|0\rangle$ and $|1\rangle$ when both amplitudes may be nonzero.

For example,

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

is a superposition. It is often called the plus state.

Another state,

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

is also a superposition.

Notice something subtle:

$$|+\rangle$$

and

$$|-\rangle$$

give the same measurement probabilities in the standard basis. For both states,

$$P(0) = \frac{1}{2}, \qquad P(1) = \frac{1}{2}.$$

So why are they different?

They differ in the relative sign between the amplitudes. This sign can affect what happens after later quantum gates. Quantum algorithms use such relative phases to create interference, where amplitudes combine constructively or destructively.

This is one reason quantum computing is not just probability theory. Two quantum states can have the same measurement probabilities in one basis but behave differently under later operations.

4.4 Measurement and the Born Rule

A measurement extracts classical information from a quantum system.

For a single qubit measured in the computational basis, the possible outcomes are 0 and 1. If

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

then the Born rule says

$$P(0)=|\alpha|\square,$$

and

$$P(1)=|\beta|\square.$$

After the measurement, the state is updated to match the outcome. If the outcome is 0, the post-measurement state is $|0\rangle$. If the outcome is 1, the post-measurement state is $|1\rangle$. This measurement update rule is part of the standard circuit model of quantum computation (Nielsen and Chuang, 2010).

For example, suppose

$$|\psi\rangle = \frac{\sqrt{3}}{2}|0\rangle + \frac{1}{2}|1\rangle.$$

Then

$$P(0)=(3)\ \square\ (4),$$

and

$$P(1)=(1)\ \square\ (4).$$

If we measure once, we might get 0 or 1. If we get 0, the state becomes

$$|0\rangle.$$

If we immediately measure again in the same basis, we will get 0 with probability 1.

This is different from merely “looking at a hidden classical value.” In the standard quantum model, measurement changes the state.

For variational quantum algorithms, this has a practical consequence:

> We cannot generally prepare one quantum state, measure it once, and learn everything we need.

Instead, we repeatedly prepare the same parameterized circuit and measure many times. Each repetition is called a shot. If we use 1,000 shots, we run the same circuit 1,000 times and collect 1,000 measurement outcomes.

For example, if a circuit ideally produces

$$P(0)=0.25,\ P(1)=0.75,$$

then 1,000 shots might produce about 250 zeros and 750 ones, though random fluctuations are expected.

This randomness is called sampling noise or shot noise. Later, in Chapter 8, we will see how shot noise affects the estimation of cost functions.

4.5 Vector Representation of a Qubit

In Chapter 2, we introduced vectors as ordered lists of numbers. A qubit state can be represented as a vector.

The basis states are written as

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Then a general qubit state

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

corresponds to the vector

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

For example,

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

is represented by

$$|+\rangle = \text{beginbmatrix } 1/\sqrt{(2)} \text{ ; } 1/\sqrt{(2)} \text{ endbmatrix.}$$

The state vector contains amplitudes, not direct measurement outcomes.

The mathematical space of possible quantum states is a complex vector space with an inner product. In quantum mechanics, such a space is called a Hilbert space. For the finite-dimensional systems used in basic quantum computing, you can think of a Hilbert space as a vector space where lengths and angles are defined in a way that works with complex numbers (Nielsen and Chuang, 2010).

The normalization rule

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

means that the state vector has length 1.

4.6 The Bloch Sphere: A Picture of One Qubit

Although a qubit state is described by complex amplitudes, it can be helpful to picture a single qubit using the Bloch sphere.

The Bloch sphere is a geometric representation of pure one-qubit states. Each point on the surface of the sphere corresponds to a possible pure qubit state, up to an overall global phase that does not affect measurement probabilities (Nielsen and Chuang, 2010).

A general one-qubit pure state can be written as

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle,$$

where:

- θ controls how much weight is on |0⟩ versus |1⟩,
- φ controls the relative phase,
- $e^{i\varphi}$ is a complex number lying on the unit circle.

Some important points are:

$$|0\rangle$$

at the north pole,

$$|1\rangle$$

at the south pole,

$$|+\rangle = \text{frac}|0\rangle+|1\rangle\sqrt{(2)}$$

on the positive x-axis, and

$$|-\rangle = \text{frac}|0\rangle-|1\rangle\sqrt{(2)}$$

on the negative x-axis.

The Bloch sphere is useful for building intuition about single-qubit gates. However, it does not scale well to many qubits. A two-qubit or many-qubit system cannot generally be visualized as a simple collection of independent Bloch spheres, because entanglement can appear.

4.7 Multiple Qubits and Tensor Products

Optimization problems rarely involve just one binary variable. A useful quantum optimization algorithm may need many qubits.

If one qubit has basis states

$$|0\rangle, |1\rangle,$$

then two qubits have four computational basis states:

$$|00\rangle, |01\rangle, |10\rangle, |11\rangle.$$

These correspond to the four possible classical bit strings of length 2.

A general two-qubit state has the form

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle,$$

where

$$|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1.$$

For n qubits, there are

$$2^n$$

computational basis states. A general n-qubit state can be written as

$$|\psi\rangle = \sum_{x \in \{0,1\}^n} \alpha_x |x\rangle,$$

where:

- x is a binary string such as 001011,
- $|x\rangle$ is the basis state corresponding to that bit string,
- α_x is the amplitude of that basis state,
- the amplitudes satisfy

$$\sum_{x \in \{0,1\}^n} |\alpha_x|^2 = 1.$$

This exponential number of amplitudes is one reason quantum systems can be difficult to simulate classically. Feynman famously argued that quantum systems appear naturally suited to simulation by quantum mechanical devices rather than ordinary classical computers (Feynman, 1982). However, the existence of 2^n amplitudes does not automatically mean that a quantum computer gives an exponential speedup for every problem. Measurement only returns classical samples, and useful algorithms must arrange amplitudes so that good answers are likely to be observed.

To combine qubits mathematically, we use the tensor product.

The tensor product is an operation that combines vector spaces. For two qubits,

$$|a\rangle \otimes |b\rangle$$

means “the joint state of qubit a and qubit b.”

Usually, we write this more compactly as

$$|a\rangle|b\rangle$$

or simply

$$|ab\rangle.$$

For example,

$$|0\rangle \otimes |1\rangle = |01\rangle.$$

Using vector form,

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

Then

$$|0\rangle \otimes |1\rangle = \text{beginbmatrix } 1 \text{ ; } 0 \text{ endbmatrix} \otimes \text{beginbmatrix } 0 \text{ ; } 1 \text{ endbmatrix} = \text{beginbmatrix } 0 \text{ ; } 1 \text{ ; } 0 \text{ ; } 0 \text{ endbmatrix},$$

which represents $|01\rangle$ in the ordered basis

$$|00\rangle, |01\rangle, |10\rangle, |11\rangle.$$

Tensor products are central because every multi-qubit circuit acts on a tensor-product state space.

4.8 Product States

A product state is a multi-qubit state that can be written as a tensor product of individual qubit states.

For example,

$$|0\rangle \otimes |1\rangle = |01\rangle$$

is a product state.

Another example is

$$|+\rangle \otimes |+\rangle.$$

Since

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

we get

$$|+\rangle \otimes |+\rangle = \leq ft(frac|0\rangle + |1\rangle \sqrt{(2)}) \otimes \leq ft(frac|0\rangle + |1\rangle \sqrt{(2)}).$$

Expanding gives

$$|+\rangle \otimes |+\rangle = (1) \square (2) \leq ft(|00\rangle + |01\rangle + |10\rangle + |11\rangle).$$

This state gives equal probability to all four bit strings:

$$P(00) = P(01) = P(10) = P(11) = \frac{1}{4}.$$

For n qubits, the state

$$|+\rangle^{\otimes n}$$

is the equal superposition over all 2^n bit strings:

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

This state is important in quantum optimization. In QAOA, for example, a common starting state is the equal superposition over all candidate bit strings. The algorithm then applies alternating operations designed to increase the probability of measuring high-quality solutions.

But again, the equal superposition alone does not solve the problem. If you simply measure

$$|+\rangle\square\otimes n\square,$$

you get a uniformly random bit string. The algorithm must transform amplitudes so that good bit strings become more likely.

4.9 Entanglement: Correlations That Are Not Classical Product States

Some multi-qubit states cannot be written as product states. These are called entangled states.

Entanglement is one of the central features distinguishing quantum information from classical information. It plays a major role in quantum computation and quantum communication (Nielsen and Chuang, 2010).

A famous example is the two-qubit Bell state

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}.$$

If we measure both qubits in the computational basis, we get:

00

with probability

(1) □ (2),

and

11

with probability

(1) □ (2).

We never get 01 or 10.

The measurement outcomes are perfectly correlated. If the first qubit is measured as 0, the second is also 0. If the first is measured as 1, the second is also 1.

Could this just be an ordinary probabilistic mixture? For this particular measurement basis, the outcomes resemble classical correlation. But the full quantum state has phase relations that cannot be described as merely “either 00 or 11 with hidden classical uncertainty.” Entangled states can produce correlations that violate inequalities satisfied by local hidden-variable theories, as shown by Bell’s theorem and later experiments (Bell, 1964; Aspect, Dalibard, and Roger, 1982).

For our purposes, the most important point is simpler:

> Entanglement means the whole quantum state cannot be described as independent states of the parts.

In variational quantum algorithms, entangling gates are often used so that the circuit can represent correlations between decision variables. For example, in a scheduling problem, the best assignment of one task may depend on the assignment of another. In a portfolio problem, choosing one asset may affect the desirability of choosing another. Quantum circuits use entangling operations to create joint states whose amplitudes depend on combinations of variables.

However, entanglement is not automatically useful. Too little entanglement may make a circuit too weak to represent good solutions. Too much unstructured circuit depth can make training difficult and can worsen noise effects on real hardware. Later chapters will discuss this tradeoff when we study ansatz design and barren plateaus.

4.10 Quantum Gates: Operations on Qubits

A quantum gate is an operation that changes a quantum state.

In the circuit model, ideal quantum gates are represented by unitary matrices. A matrix U is unitary if

$$U^\dagger U = I,$$

where:

- U^† is the conjugate transpose of U,
- I is the identity matrix.

Unitary operations preserve the total probability of the quantum state. In other words, if a state is normalized before the gate, it remains normalized after the gate. This is a basic requirement for valid closed-system quantum evolution in the standard circuit model (Nielsen and Chuang, 2010).

If the input state is

$$|\psi\rangle,$$

then after applying gate U, the state becomes

$$U|\psi\rangle.$$

Let us examine several gates that appear frequently in variational quantum algorithms.

4.11 The Pauli-X Gate: A Quantum NOT

The Pauli-X gate is represented by the matrix

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

It acts as

$$X|0\rangle = |1\rangle,$$

and

$$X|1\rangle = |0\rangle.$$

So, on computational basis states, it behaves like a classical NOT gate.

For example,

$$X \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

But because quantum gates are linear, X also acts on superpositions. If

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle,$$

then

$$X|\psi\rangle = \alpha|1\rangle + \beta|0\rangle.$$

Equivalently,

$$X|\psi\rangle = \beta|0\rangle + \alpha|1\rangle.$$

So X swaps the amplitudes of $|0\rangle$ and $|1\rangle$.

4.12 The Pauli-Z Gate: A Phase Flip

The Pauli-Z gate is

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

It acts as

$$Z|0\rangle = |0\rangle,$$

and

$$Z|1\rangle = -|1\rangle.$$

The Z gate does not change the measurement probabilities in the computational basis by itself. It changes the sign, or phase, of the |1⟩ amplitude.

For example,

$$Z|+\rangle = Z\leq ft(frac|0\rangle+|1\rangle\sqrt{(2)}) = frac|0\rangle-|1\rangle\sqrt{(2)} = |-\rangle.$$

If we measure |+⟩ or |-⟩ immediately in the computational basis, both give 0 or 1 with equal probability. But after additional gates, the difference can matter.

This is the beginning of quantum interference.

4.13 The Hadamard Gate: Creating and Reversing Superposition

The Hadamard gate, usually written H, is

$$H = frac1\sqrt{(2)}\ beginbmatrix\ 1\ 1\ ;\ 1\ -1\ endbmatrix.$$

It acts as

$$H|0\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} = |+\rangle,$$

and

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

The Hadamard gate is often used to create superposition.

For example, applying H to each qubit in $|000\rangle$ creates

$$H^{\otimes 3}|000\rangle = |+\rangle^{\otimes 3}.$$

This equals

$$\frac{1}{\sqrt{8}}\left(|000\rangle + |001\rangle + |010\rangle + |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle \right).$$

This state gives equal probability to all 8 bit strings.

The Hadamard also demonstrates interference. Since

$$H|+\rangle = |0\rangle,$$

and

$$H|-\rangle = |1\rangle,$$

the relative sign in $|+\rangle$ versus $|-\rangle$ becomes visible after another Hadamard.

Let us calculate one case:

$$H|+\rangle = H\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right) = \frac{H|0\rangle+H|1\rangle}{\sqrt{2}}.$$

Substitute:

$$= \frac{1}{\sqrt{2}}\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}} + \frac{|0\rangle-|1\rangle}{\sqrt{2}} \right).$$

The $|1\rangle$ terms cancel:

$$= (1) \square (2) \leq ft(2|0\rangle) = |0\rangle.$$

This cancellation is destructive interference. The reinforcement of the $|0\rangle$ term is constructive interference.

Quantum algorithms are designed so that amplitudes of good answers interfere constructively while amplitudes of bad answers interfere destructively. Achieving this is difficult, which is why quantum algorithm design is subtle.

4.14 Rotation Gates: Adjustable Operations

Variational quantum algorithms need adjustable circuits. The adjustable numbers in a circuit are called parameters.

A common family of parameterized gates is the rotation gates:

$$R\square X\square(\theta),\ R\square Y\square(\theta),\ R\square Z\square(\theta).$$

Here, θ is a real-valued parameter, often interpreted as an angle.

For example, the Y-rotation gate is

$$R\square Y\square(\theta) = beginbmatrix\ cos(\theta/2)\ -sin(\theta/2)\ ;\ sin(\theta/2)\ cos(\theta/2)\ endbmatrix.$$

Applied to $|0\rangle$, it gives

$$R_Y(\theta)|0\rangle = \cos(\theta/2)|0\rangle + \sin(\theta/2)|1\rangle.$$

If $\theta = 0$, then

$$R_Y(0)|0\rangle = |0\rangle.$$

If $\theta = \pi$, then

$$R_Y(\pi)|0\rangle = |1\rangle.$$

If $\theta = (\pi)/(2)$, then

$$R□Y□≤ft((\pi) □ (2))|0\rangle = frac1√(2)|0\rangle + frac1√(2)|1\rangle.$$

This is exactly the kind of adjustable behavior used in variational circuits.

A variational algorithm may prepare a state such as

$$|\psi(\theta)\rangle = R□Y□(\theta)|0\rangle,$$

measure it, compute a cost estimate, and then use a classical optimizer to update θ.

For many qubits, we use many parameters:

$$|\psi(boldsymbol\theta)\rangle,$$

where

$$boldsymbol\theta = (\theta□,\theta□,□,\theta□)$$

is a vector of circuit parameters.

This notation will appear often in later chapters.

4.15 Two-Qubit Gates and CNOT

Single-qubit gates can rotate individual qubits, but they cannot create entanglement by themselves. To create entanglement, we need gates that act jointly on two or more qubits.

The most common example is the controlled-NOT gate, or CNOT.

CNOT acts on two qubits:

- the first qubit is the control,
- the second qubit is the target.

The rule is:

> If the control qubit is 1, flip the target qubit. If the control qubit is 0, leave the target unchanged.

On basis states,

$$\mathrm{CNOT}|00\rangle = |00\rangle,$$

$$\mathrm{CNOT}|01\rangle = |01\rangle,$$

$$\mathrm{CNOT}|10\rangle = |11\rangle,$$

$$\mathrm{CNOT}|11\rangle = |10\rangle.$$

Now let us see how CNOT creates entanglement.

Start with

$$|00\rangle.$$

Apply a Hadamard to the first qubit:

$$(H \otimes I)|00\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}}.$$

Now apply CNOT with the first qubit as control and the second as target:

$$\mathrm{CNOT}\left(\frac{|00\rangle + |10\rangle}{\sqrt{2}}\right) = \frac{|00\rangle + |11\rangle}{\sqrt{2}}.$$

This is the Bell state

$$|\Phi^+\rangle.$$

The CNOT gate is therefore an entangling gate. Variational quantum circuits typically alternate layers of single-qubit rotations with entangling gates such as CNOT, CZ, or hardware-native two-qubit gates.

The exact choice matters because real quantum devices have limited connectivity and noisy gates. This hardware reality will become important in Chapter 16.

4.16 Quantum Circuits

A quantum circuit is a sequence of quantum gates and measurements applied to qubits.

A simple circuit might look conceptually like this:

1. Start with qubits in $|0\rangle$ states.
2. Apply Hadamard gates to create superposition.
3. Apply parameterized rotation gates.
4. Apply entangling gates.
5. Measure the qubits.
6. Record a classical bit string.

Mathematically, if a circuit applies gates

$$U_1, U_2, \ldots, U_L,$$

then the total unitary operation is

$$U = U_L U_{L-1} \cdots U_2 U_1.$$

The order looks reversed because the first gate acts on the state first.

If the initial state is

$$|0\rangle^{\otimes n},$$

then the final state before measurement is

$$|\psi\rangle = U|0\rangle^{\otimes n}.$$

For a parameterized quantum circuit, the unitary depends on parameters:

$$U(\boldsymbol{\theta}).$$

The prepared state is

$$|\psi(\boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|0\rangle^{\otimes n}.$$

This is one of the most important formulas in variational quantum algorithms.

The algorithm does not usually know all amplitudes of

$$|\psi(\boldsymbol{\theta})\rangle.$$

Instead, it estimates useful quantities by measurement.

4.17 From Circuits to Samples

When we measure an n-qubit state in the computational basis, we obtain one n-bit string.

For example, measuring 4 qubits may produce

$$0101.$$

If the state is

$$|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x |x\rangle,$$

then the probability of observing bit string x is

$$P(x) = |\alpha_x|^2.$$

Suppose a two-qubit state is

$$|\psi\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{\sqrt{2}}|11\rangle.$$

There is no $|10\rangle$ term, so its amplitude is 0.

The measurement probabilities are:

$$P(00) = \frac{1}{4},$$

$$P(01) = \frac{1}{4},$$

$$P(10) = 0,$$

$$P(11) = \frac{1}{2}.$$

If this state represents candidate solutions to an optimization problem, then the circuit is sampling candidate bit strings according to these probabilities.

A variational quantum optimization algorithm tries to adjust the circuit parameters so that better bit strings are sampled more often.

This is the basic sampling view of quantum optimization.

4.18 Observables and Expectation Values

In optimization, we usually want a number that tells us how good a solution is. In quantum mechanics, measurable quantities are represented by mathematical objects called observables.

An observable is represented by a Hermitian matrix. A matrix A is Hermitian if

$$A^\dagger = A.$$

Hermitian matrices have real eigenvalues, which is necessary because measurement outcomes of physical quantities are real numbers (Nielsen and Chuang, 2010).

For variational algorithms, we often care about the expectation value of an observable H in a quantum state |ψ⟩. It is written as

$$\langle\psi|H|\psi\rangle.$$

The symbol

$$\langle\psi|$$

is called a bra. It is the conjugate transpose of the ket

$$|\psi\rangle.$$

The expression

$$\langle\psi|H|\psi\rangle$$

means the average value we would estimate by repeatedly preparing |ψ⟩, measuring the observable H, and averaging the results.

In variational quantum algorithms, the cost function often has the form

$$C(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle.$$

Here:

- boldsymbolθ are circuit parameters,
- |ψ(boldsymbolθ)⟩ is the quantum state prepared by the circuit,
- H is an observable encoding the problem,
- C(boldsymbolθ) is the cost that the classical optimizer tries to minimize or maximize.

This formula is the bridge between quantum circuits and optimization.

In VQE, H may represent a molecular Hamiltonian, and the goal is to minimize the expected energy. In QAOA, H may represent a combinatorial objective function such as MaxCut. These algorithms will be developed later in the book.

4.19 A First Example: One-Qubit Variational Cost

Let us build a tiny example to connect the ideas.

Suppose we prepare

$$|\psi(\theta)\rangle = R_Y(\theta)|0\rangle.$$

From earlier,

$$|\psi(\theta)\rangle = \cos(\theta/2)|0\rangle + \sin(\theta/2)|1\rangle.$$

Now suppose our observable is

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

The expectation value is

$$C(\theta) = \langle\psi(\theta)|Z|\psi(\theta)\rangle.$$

Since measuring Z gives value +1 for outcome |0⟩ and -1 for outcome |1⟩, we can compute

$$C(\theta) = (+1)P(0) + (-1)P(1).$$

Here,

$$P(0) = \cos^2(\theta/2),$$

and

$$P(1) = \sin^2(\theta/2).$$

Therefore,

\[C(θ) = cos□(θ/2) - sin

Document information

Chapter 4: Quantum Computing from First Principles

Project	Variational Quantum Algorithms for Optimization
Document	Document 1.8
Author	phone
Verifier	Not verified
Downloaded	July 04, 2026 18:10 KST
Status	Working
Document link	https://www.theorytrace.com/projects/variational-quantum-algorithms-for-optimization/-documents/chapter-4-quantum-computing-from-first-principles/