4. Quantum Algorithms: Grover Search and Applications

2024/05/10

Atsushi Matsuo

IBM Research – Tokyo

Lecture 4: Quantum Algorithms:
Grover search and Applications

Agenda

- Introduction
- Grover search
- Quantum circuit for Grover search
- Qiskit Implementation

<u>Break</u>

- Geometric view of Grover Iteration
- Optimality of Grover search
- Summary
- Homework

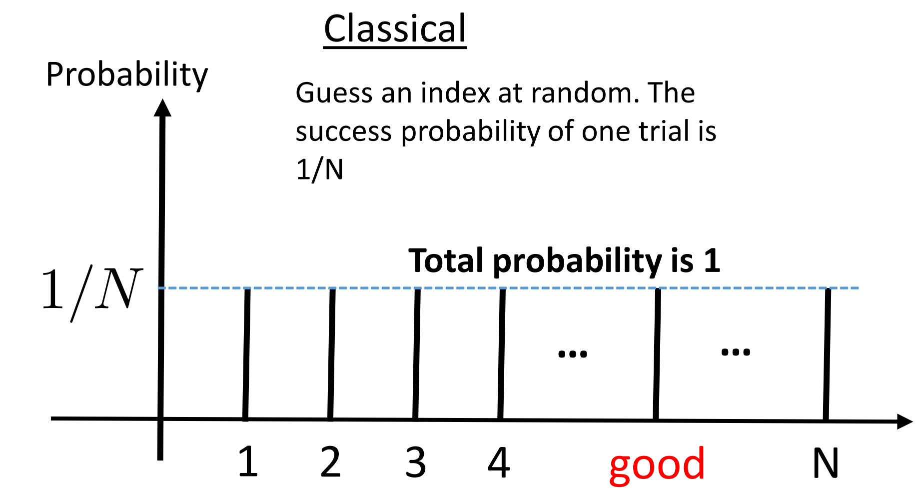
Introduction

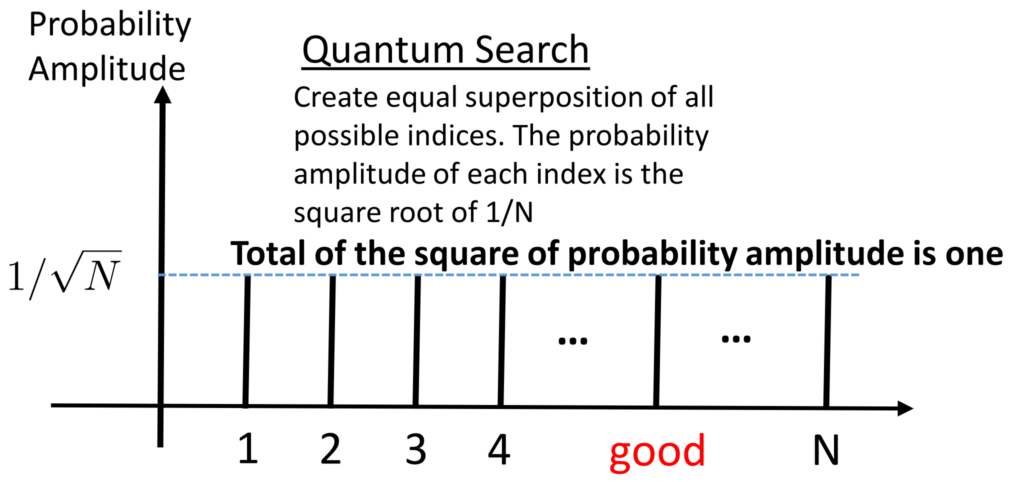
- The Grover search* is a quantum search algorithm.
  - Searching an unsorted database is often used as an example.
  - Also, it can be used to speed up many classical algorithms that use search algorithms
- Searching problem: Find ω from a list *L*
  - *L* is a list of size *N*, and ω is called the answer (or the "good" index).
- *How can we find ω from the list L?*
  - In classical computation, check each element of *L* until we find the answer.
    - In the worst case, Need $O(N)$ times.
  - In quantum computation, use Grover search!
    - Need $O(\sqrt{N})$ times.
- Quadratic speed up, <u>not exponential.</u>

* Grover, Lov K. "A fast quantum mechanical algorithm for database search." *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. 1996.

Classical and Quantum Search Algorithm

- **Searching Problem**
  - Find a good index *i* from N possible indices
  - Suppose we are given a black box that can answer whether index *i* is "good" or not. The black box is often called an "oracle".
- **Classical Search Algorithm**
  - Pick an index from 1～N at random. Ask the oracle with the index
  - The success probability is 1/N (if there is exactly one good index)
- **Quantum Search Algorithm**
  - Ask a quantum oracle with the superposition of all indices
  - With a query to the quantum oracle, the success probability is still 1/N, but before the measurement, the probability amplitude is 1/√N





Probability and Probability Amplitudes

- Success probabilities of classical algorithms
  - If one trial has success probability 1/N, k trials have success probability ~ k/N
  - Need to repeat k times up to the same order N
- Probability amplitude of quantum algorithms
  - Create a quantum superposition of all possible indices

$$\frac{1}{\sqrt{N}}|0\rangle + \frac{1}{\sqrt{N}}|1\rangle + \dots + \frac{1}{\sqrt{N}}|N-1\rangle$$

  - Query the oracle to mark the bad/good indices and store the result in the second register

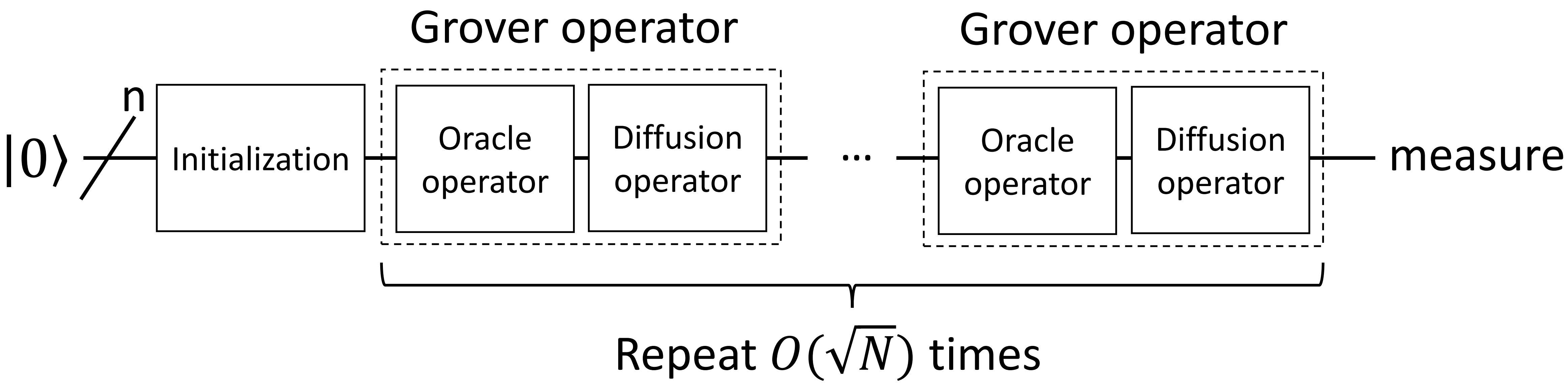
$$\frac{1}{\sqrt{N}}|0\rangle|\mathrm{bad}\rangle + \frac{1}{\sqrt{N}}|1\rangle|\mathrm{bad}\rangle + \dots + \frac{1}{\sqrt{N}}|i\rangle|\mathrm{good}\rangle + \dots + \frac{1}{\sqrt{N}}|N-1\rangle|\mathrm{bad}\rangle$$

  - If we measure right after that, the result is similar to the classical algorithm
  - But, if we can add/gather the probability amplitudes, we may be able to amplify the good states quadratically faster
    - k repetitions of probability amplitudes resulting in $\frac{k}{\sqrt{N}}$ with success probability $\frac{k^2}{N}$
    - Only need to repeat up to the same order of the square root of N

Grover Search

Overview

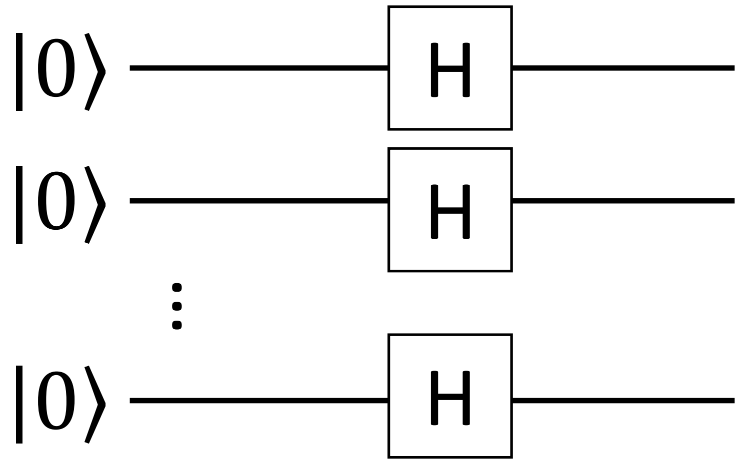
- The Grover search consists of three parts.
  1. Initialization
  2. Apply an Oracle operator
  3. Apply a Diffusion operator
- Repeat above 2 and 3 $O(\sqrt{N})$ times after initialization.



Initialization

- Create the superposition of all possible states $|00 \ldots 0\rangle \ldots |11 \ldots .1\rangle$ with equal amplitudes

- Apply Hadamard (H) gates to each qubit.



- The state will change to $|s\rangle = \sum_{x\in\{0,1\}^n}^{n} \frac{1}{\sqrt{2^n}} |x\rangle$ from $|00 \ldots 0\rangle$

Oracle operator

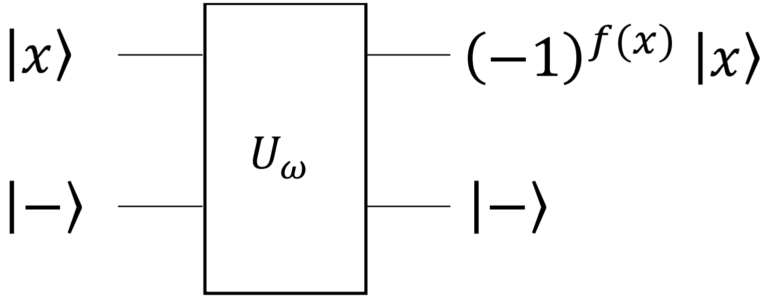
- Use the oracle in the oracle operator.
- Oracle: It's a black box function $f(x)$ as follows

$$\begin{cases} f(x) = 1 \ \text{ for } x = \omega, \\ f(x) = 0 \ \text{ for } x \neq \omega. \end{cases}$$

- Oracle operator: It's a black box operator U_ω as follows

$$U_\omega|x\rangle = (-1)^{f(x)}|x\rangle \qquad \begin{cases} U_\omega|x\rangle = -|x\rangle \ \text{ for } x = \omega, \\ U_\omega|x\rangle = \quad |x\rangle \ \text{ for } x \neq \omega. \end{cases}$$

- It changes the phase of $|x\rangle$ if $x = \omega$ by using a phase kickback.
    - I will explain in more detail later.



But… How can we make it? When we do not know the answer?

- I will explain in more detail later.

Diffusion operator

the superposition of all possible states with equal amplitudes. ($2^n = N$)

- Diffusion operator: $U_s = 2|s\rangle\langle s| - I$
- An Operator for the Inversion about the mean.
  - What does it mean?

$$|s\rangle = N^{-1/2} \sum_{i\in\{0,1\}^n}^{n} |i\rangle$$

$$\langle s| = N^{-1/2} \sum_{j\in\{0,1\}^n}^{n} \langle j|$$

$$\langle j|i\rangle = \delta_{ij}$$

$$\langle \alpha \rangle = N^{-1} \sum_k \alpha_k$$

$$(2|s\rangle\langle s| - I)\sum_k \alpha_k |k\rangle$$

$$= 2N^{-1} \sum_{i,j,k} \alpha_k |i\rangle\langle j|k\rangle - \sum_k \alpha_k |k\rangle$$

$$= 2N^{-1} \sum_{i,k} \alpha_k |i\rangle - \sum_k \alpha_k |k\rangle$$

$$= \sum_k (2\langle\alpha\rangle - \alpha_k)|k\rangle$$

Arrange i and k since $2\langle\alpha\rangle$ is just a scalar

0 $\quad \alpha_k \quad \langle\alpha\rangle \quad 2\langle\alpha\rangle - \alpha_k$

Example of 2-qubit Grover search

- Suppose ω is 2
- Initializing: Obtain the super position of all the possible states with equal amplitudes

$$|s\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|\omega\rangle + \frac{1}{2}|11\rangle$$

- Apply an Oracle operator: Changes the phase of ω

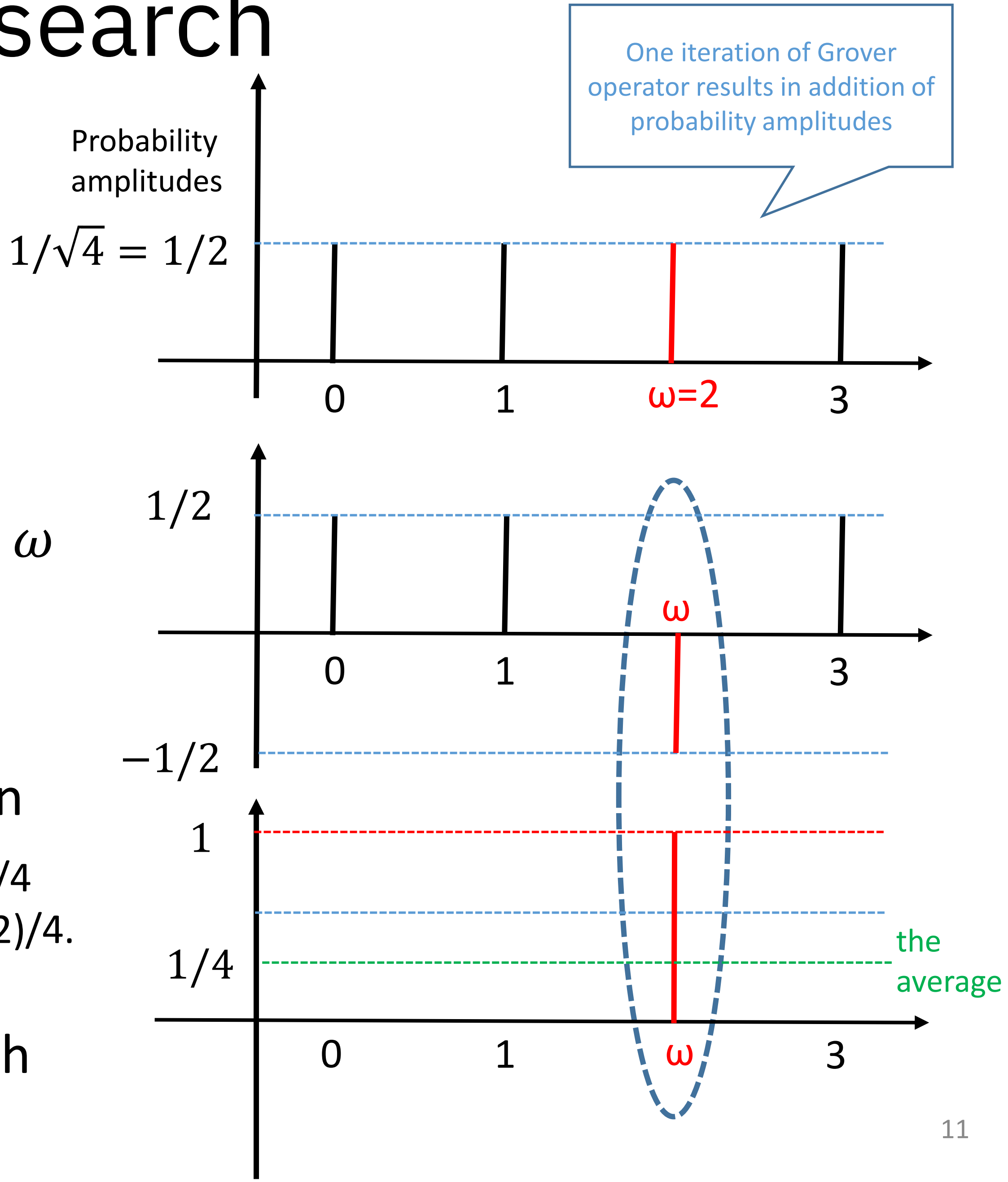
$$U_\omega|s\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle - \frac{1}{2}|\omega\rangle + \frac{1}{2}|11\rangle$$

- Apply a Diffusion operator: Inversion about mean

$$U_s U_\omega|s\rangle = 0|00\rangle + 0|01\rangle + 1|\omega\rangle + 0|11\rangle$$

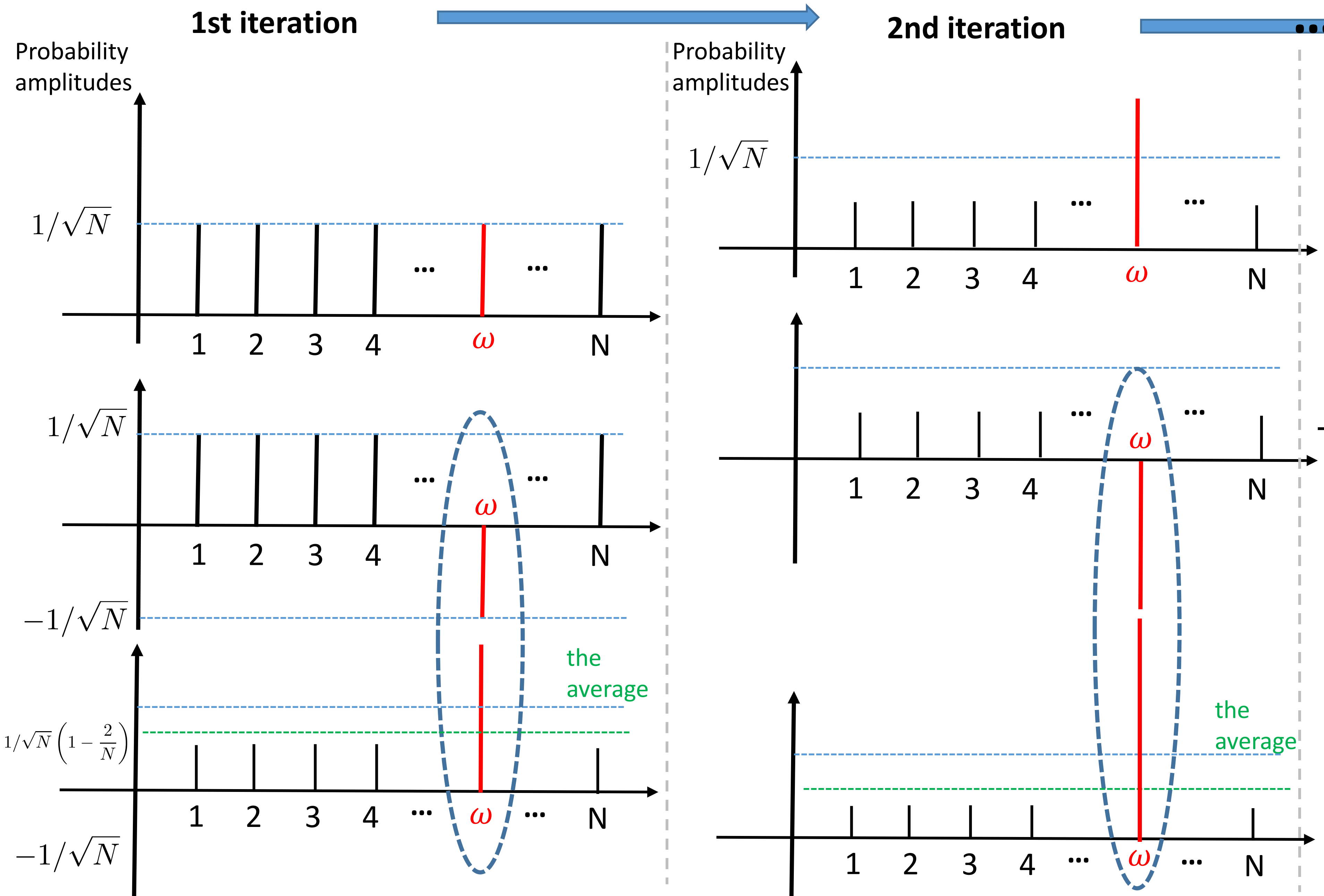
The average is 1/4 since (3*1/2 -1/2)/4.

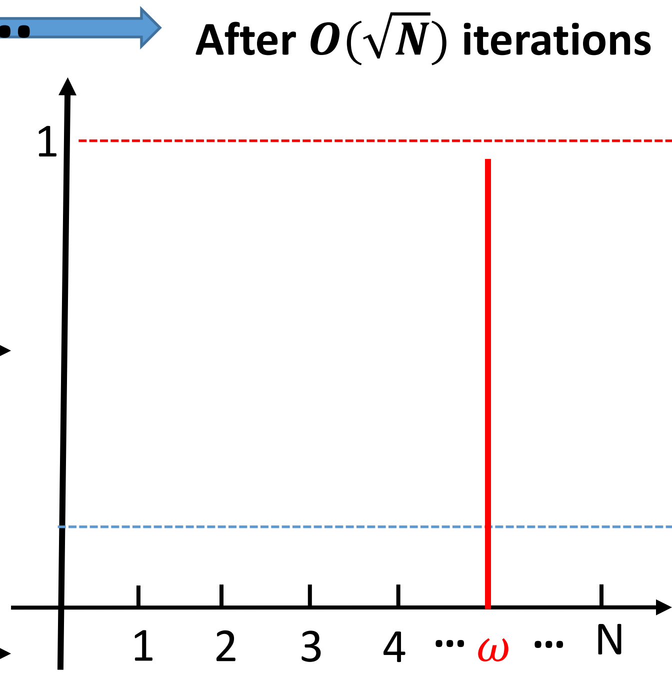
<u>Only 1 iteration</u> is needed for 2-qubit Grover search



n-qubit Grover search

Good stateのインデックスをiからオメガに変更





The amplitudes of all the “bad” states have been lowered, and just the amplitude of $|\omega\rangle$ becomes ~ 1

Don’t iterate “too many” times!

The amplitude of $|\omega\rangle$ will start decreasing.

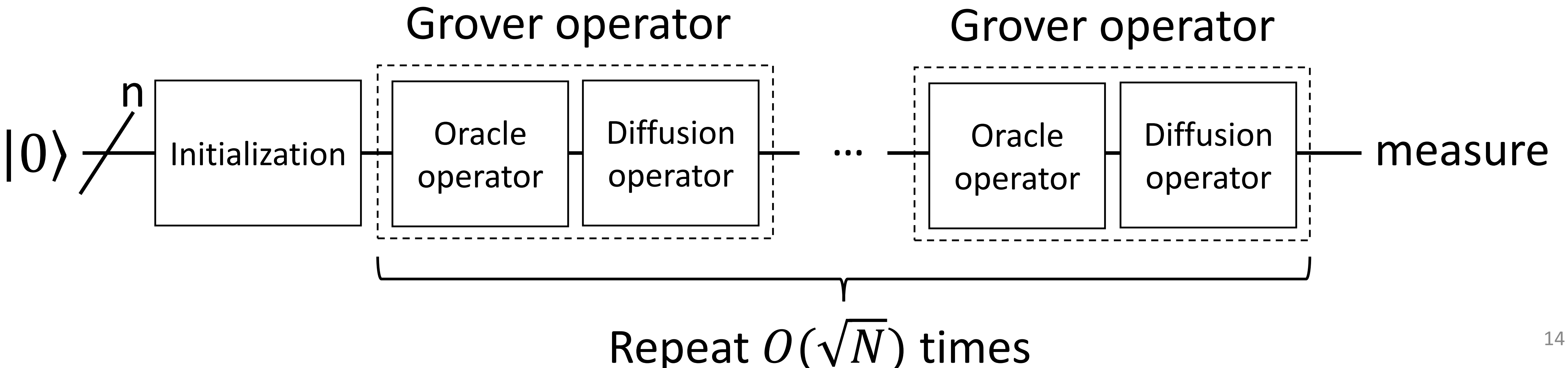
Quantum circuit for Grover search

How to create Quantum circuit for Grover search?

- Initialization: Apply Hadamard (H) gates to each qubit (easy)
- Diffusion operator: create $U_s = 2|s\rangle\langle s| - I$ (sounds possible)
- Oracle operator: create $U_\omega|x\rangle = (-1)^{f(x)}|x\rangle$
  ~~How can we [illegible] do it? When we don't know the answer? (impossible?)~~
  ~~If we can create the oracle, that means we know the answer, right?~~

Wrong!

There is a clear distinction between knowing the answer and being able to recognize the answer



How to create an oracle?

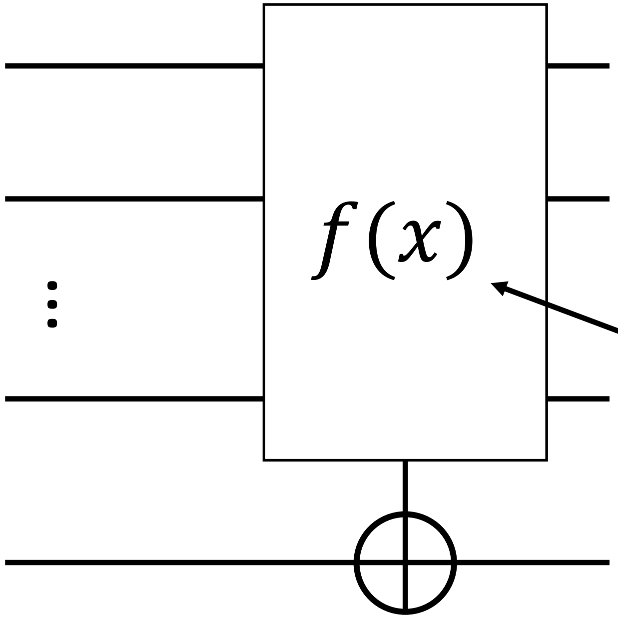
- An oracle is black box function $f(x)$ as follows

$$\begin{cases} f(x) = 1 \ \text{ for } x = \omega, \\ f(x) = 0 \ \text{ for } x \neq \omega. \end{cases}$$

- How to create it?
  - → Just implement $f(x)$ to the quantum circuit!

Input qubits corresponds to x

ancillary bit $|0\rangle$



$|f(x)\rangle$

We can implement any logical function $f(x)$ with quantum gates.

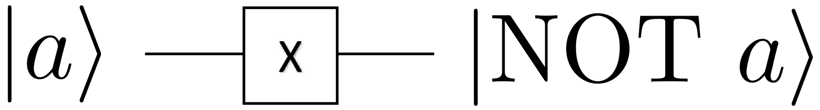
* We may use additional qubits as registers.

$$f(x) = (x_1 \vee \overline{x_2} \vee x_2) \wedge \ldots$$

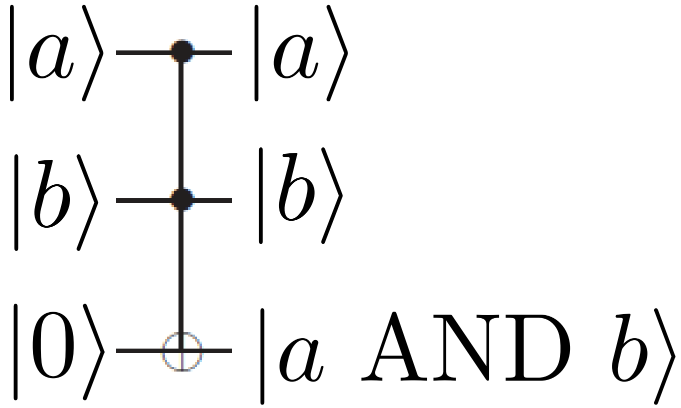
Reversible Boolean gates

For *a, b* in *{0, 1}* (i.e., binaries), we can see compute the following operations with reversible gates.

NOT gate

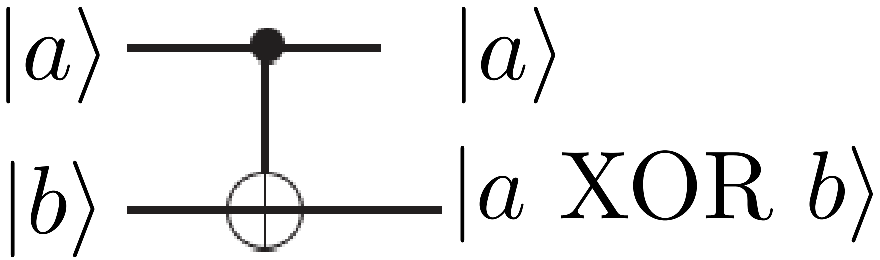


AND gate



CCNOT gate (Toffoli gate) is a conditional gate that performs an X-gate on target bit (q2), if the two control qubits (q1, q0) are |11>.

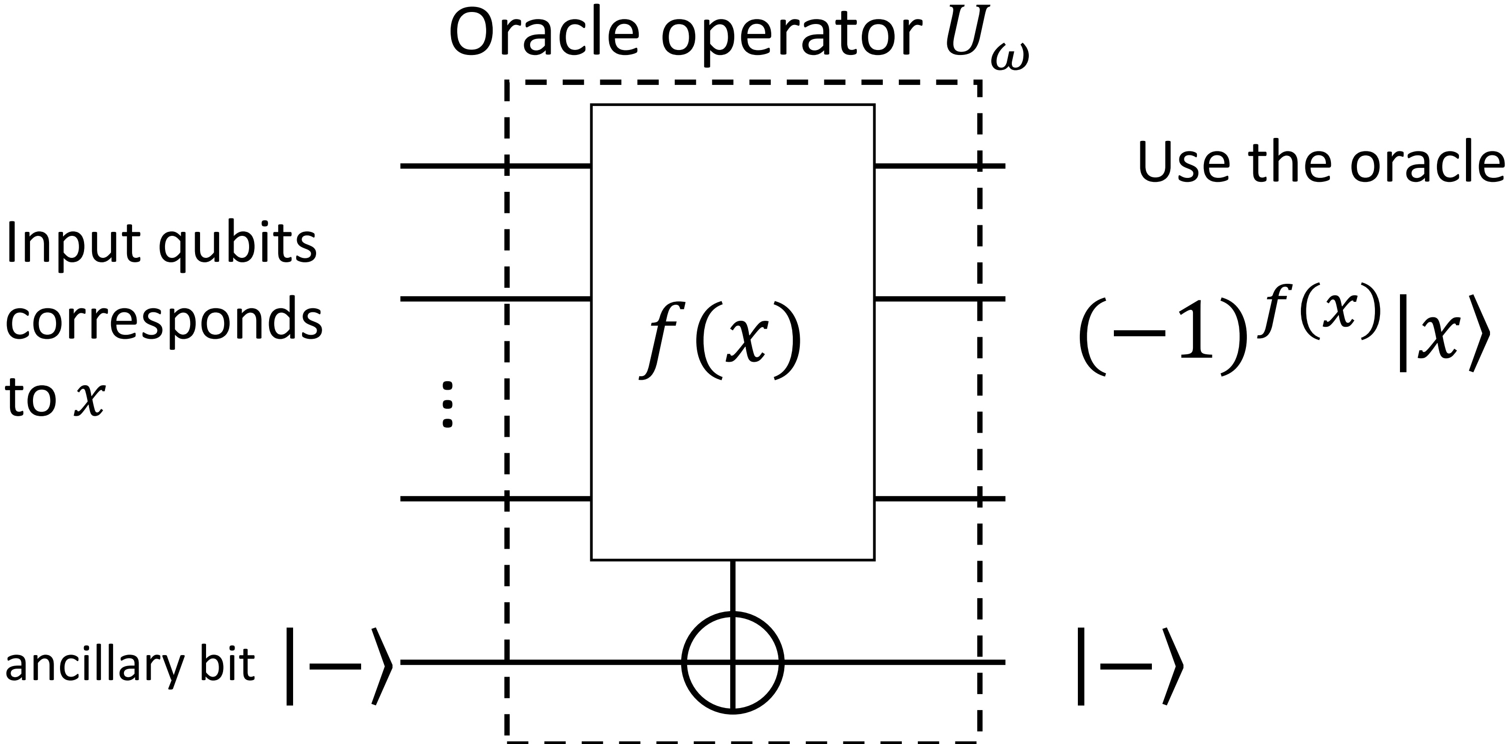
XOR gate



How to create Oracle operator

- Oracle operator: a black box operator U_ω as follows.
  It changes the <u>phase</u> of $|x\rangle$ if $x = \omega$ by using a phase kick back.

$$U_\omega|x\rangle = (-1)^{f(x)}|x\rangle \qquad \begin{cases} U_\omega|x\rangle = -|x\rangle \text{ for } x = \omega, \\ U_\omega|x\rangle = \quad |x\rangle \text{ for } x \neq \omega. \end{cases}$$

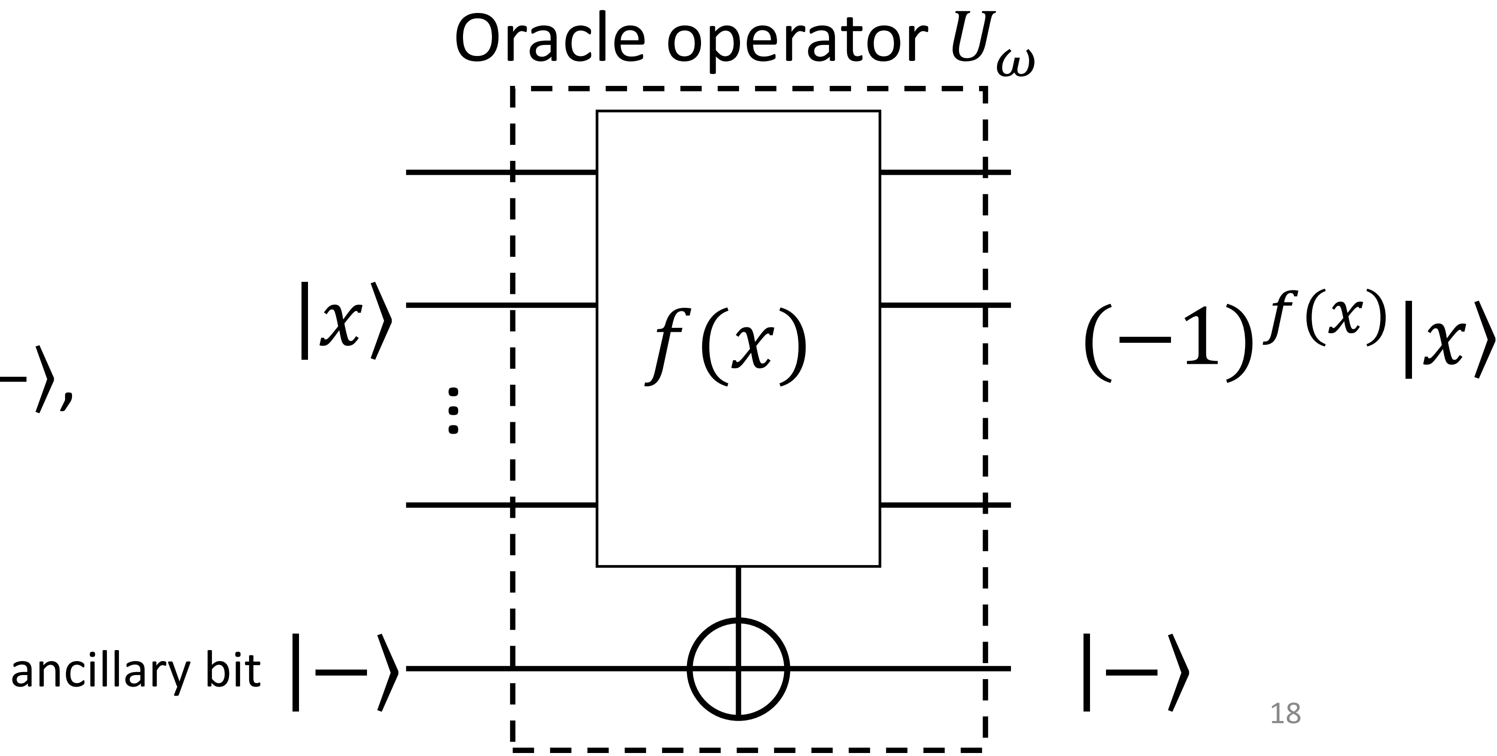


Use the oracle and a phase kickback!

Phase kickback

- $|-\rangle$ is an eigenvector of the matrix representing an X gate, with an eigenvalue of -1.
  - $|-\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix}$
  - A matrix of an X gate is $\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
  - $X|-\rangle = -1 * |-\rangle$

- When we apply an X gate to $|-\rangle$, the state remains unchanged but we obtain a phase of -1.



How to create diffusion operator?

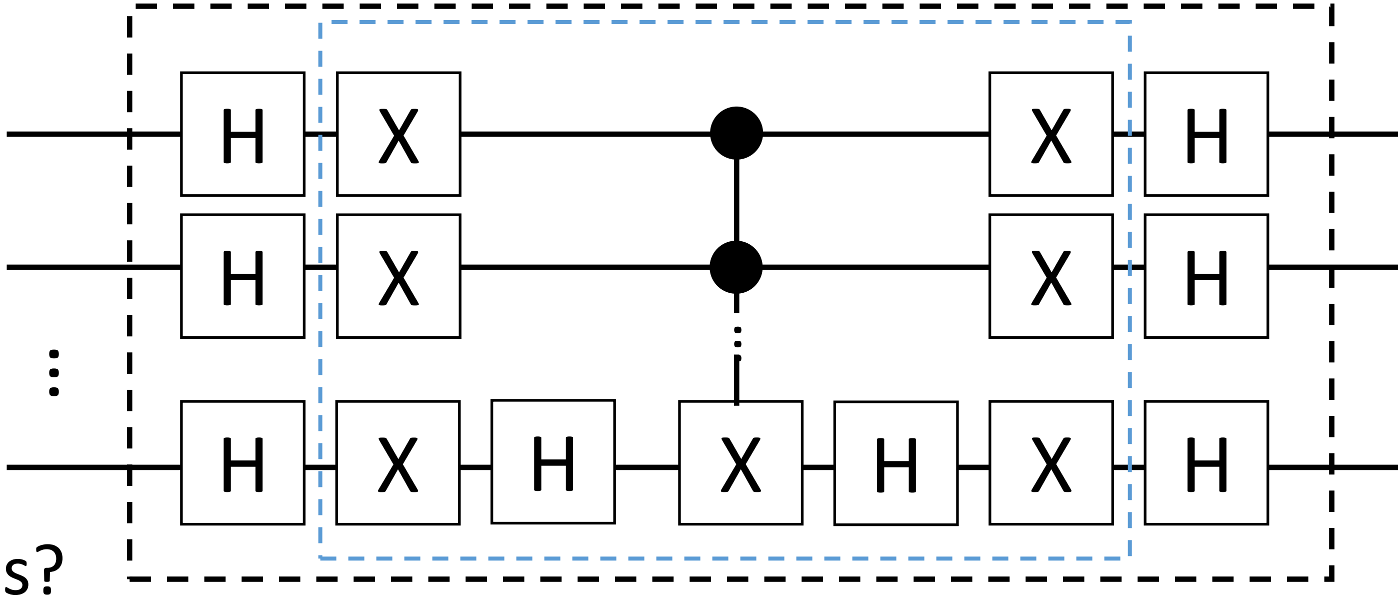
$$|s\rangle = N^{-1/2}\sum_{i\in\{0,1\}^n}^{n}|i\rangle$$

- Diffusion operator: $U_s = 2|s\rangle\langle s| - I$
- $2|s\rangle\langle s| - I = H^{\otimes n}(2|0\rangle\langle 0| - I)H^{\otimes n}$
- We consider the following operator
  - Equal to $(2|0\rangle\langle 0| - I)$ *up to global phase*

HH=I
Since an H gate is self-inverse

$$-(2|0\rangle\langle 0| - I) = \begin{bmatrix} -1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix}$$

Diffusion operator U_s

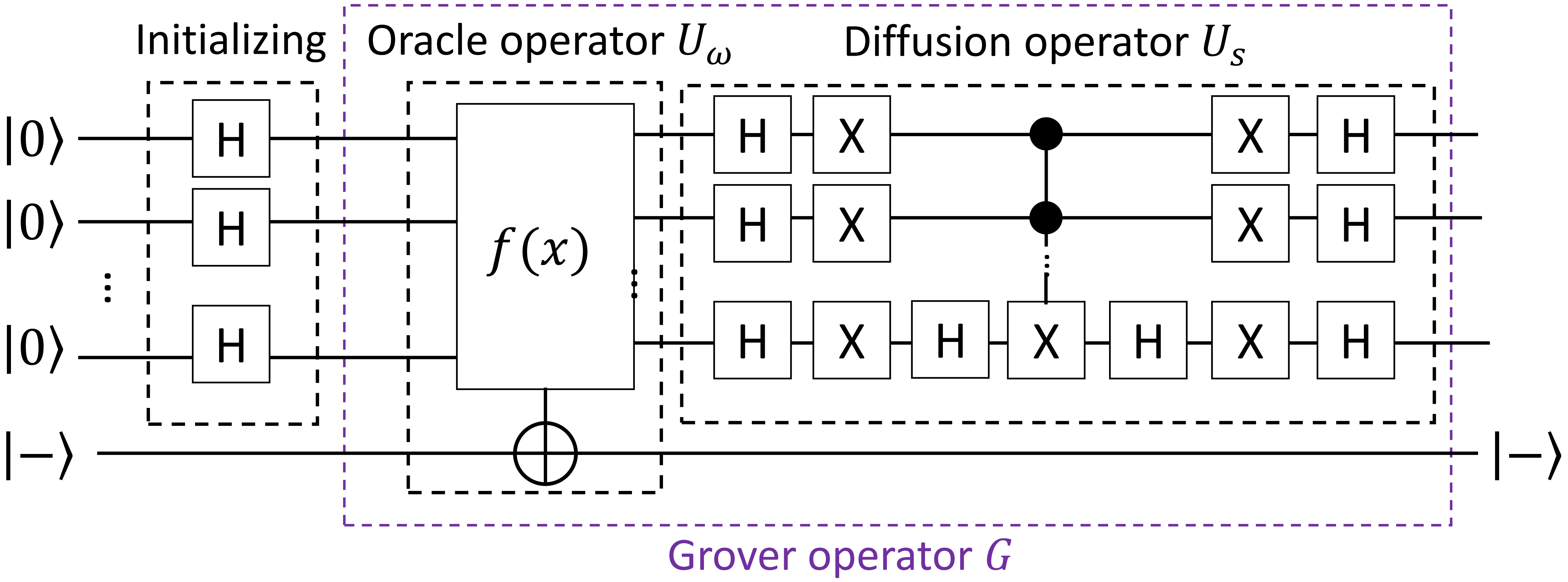


$-(2|0\rangle\langle 0| - I)$

- Looks similar to Controlled-Z gates?
  - Changes a phase of $|00 \dots 0\rangle$

Quantum circuit for Grover search

- By combining those circuit, we obtain a quantum circuit as follows.



Repeat $O(\sqrt{N})$ times

Break

We have a hands-on session next.

Please make sure to prepare your laptop.

Qiskit implementation

Let's implement!

Geometric view of Grover Iteration

“Good” vector and “bad” vector

- A quantum state is represented as a vector. We can always represent it as a weighted sum of other orthogonal vectors.

$$|\psi\rangle = \alpha\,|B\rangle + \beta\,|G\rangle = \gamma\,|\phi\rangle + \delta\,\left|\phi^{\perp}\right\rangle$$

- We can explain the behavior of Grover iterations as rotations. The vector is moved towards the space of the “good” vector.
  - “good” vector: $|\omega\rangle$ (“good” state)
  - ”bad” vector: spans perpendicular to $|\omega\rangle$, which is obtained from $|s\rangle$ by removing $|\omega\rangle$ and rescaling them.

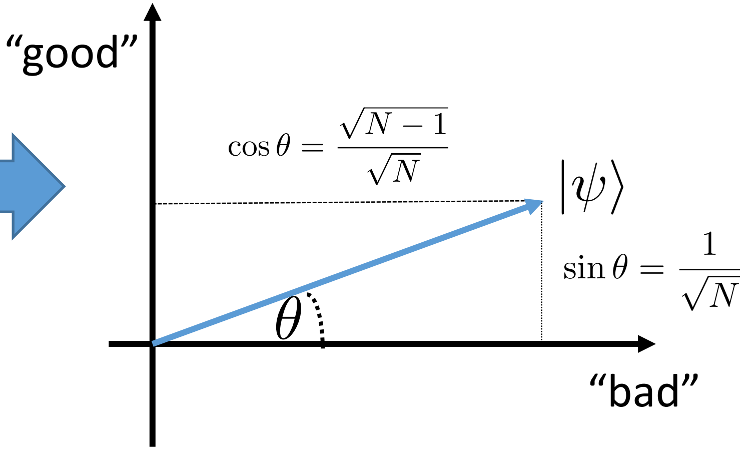
$$|\psi\rangle = \cos\theta|bad\rangle + \sin\theta\,|good\rangle,$$
$$\cos^2\theta + \sin^2\theta = 1$$

Grover iteration explained with rotations of vectors

- **The initial state**

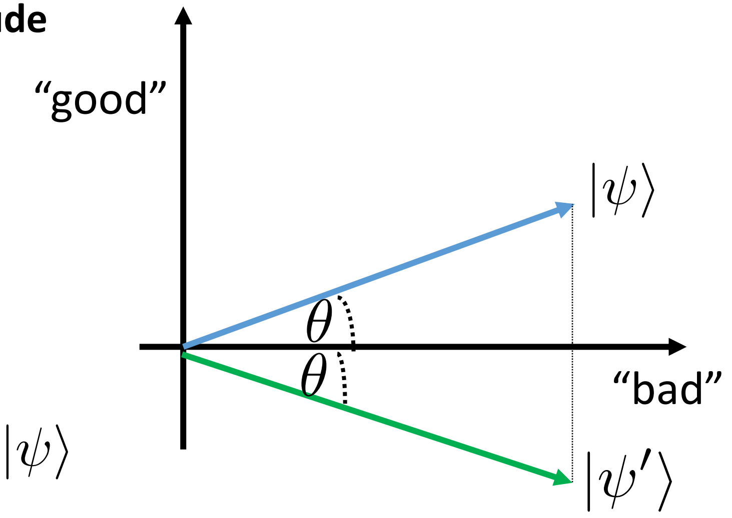
Note that $\sin\theta = \frac{1}{\sqrt{N}}$, $\cos\theta = \frac{\sqrt{N-1}}{\sqrt{N}}$

$$|\psi\rangle = \frac{1}{\sqrt{N}}|0\rangle + \frac{1}{\sqrt{N}}|1\rangle + \ldots + \frac{1}{\sqrt{N}}|N\rangle = \cos\theta\,|\mathrm{bad}\rangle + \sin\theta\,|\mathrm{good}\rangle$$

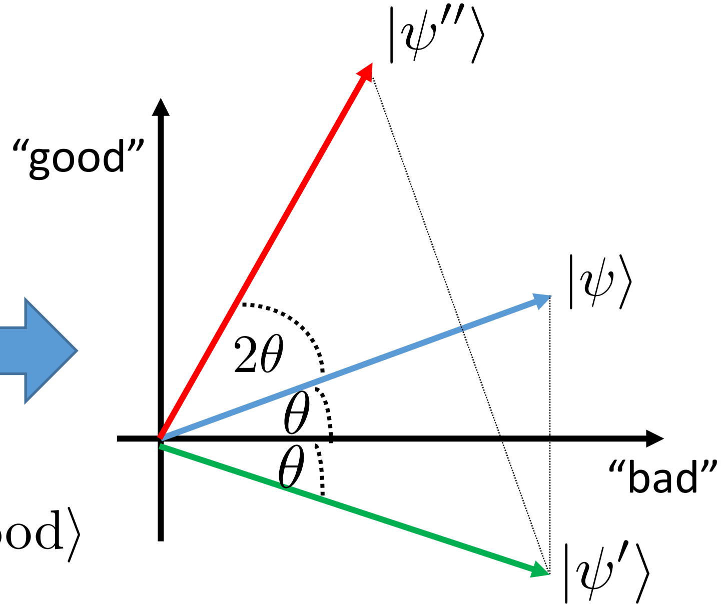


- **The oracle operator flips the probability amplitude of the good vector and leaves the bad one.**

$$U_\omega|\psi\rangle = |\psi'\rangle = \cos\theta\,|\mathrm{bad}\rangle - \sin\theta\,|\mathrm{good}\rangle$$



- **The diffusion operator flips the probability amplitude over the initial vector** $|\psi\rangle$



- **We have rotated the initial state by the angle ⌈2θ⌋ towards the “good” vector space.**

$$|\psi''\rangle = \cos(3\theta)\,|\mathrm{bad}\rangle + \sin(3\theta)\,|\mathrm{good}\rangle$$

The number of the optimal iteration

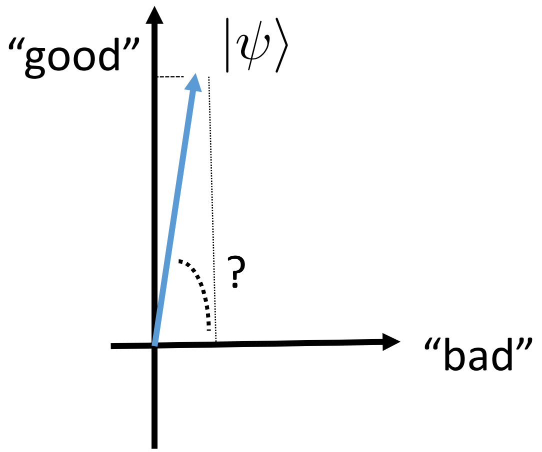
- After k iterations, the state will be $(U_s U_\omega)^k |s\rangle = \cos(2k+1)\theta\, |bad\rangle + \sin(2k+1)\theta\, |good\rangle$
- When k is $\mathrm{R} = \mathrm{ClosestInteger}\left(\frac{\pi}{4\theta} - \frac{1}{2}\right)$, $(U_s U_\omega)^k |s\rangle$ will be the closest to $|good\rangle$
  - ClosestInteger(x) means the closest integer to x
  - when (2k + 1)θ is π/2, the amplitude of ”good” will be 1
- Estimate the upper bound of R

using $\theta \geq \sin\theta = \frac{1}{\sqrt{N}}$

$$R \leq \left(\frac{\pi}{4\theta} - \frac{1}{2}\right) + 1 = \frac{\pi}{4\theta} + \frac{1}{2} \leq \frac{\pi}{4}\sqrt{N} + \frac{1}{2}$$

R is at most $O(\sqrt{N})$.

How many times do you need to repeat?



Summary of Geometric view of Grover iteration

- The success probability before applying the Grover operator is

$$\|\sin(\theta)\|^2 = \frac{1}{N}$$

- One step of Grover iterations rotates the vector by the angle 2θ towards the good space, and k steps of the iterations result in the success probability

$$\|\sin(\theta + 2k\theta)\|^2$$

- We can choose the number of iterations *k* approximately $\pi/(4\theta) \approx \sqrt{N}$ to get ”good” answers with a high success probability.

Optimality of Grover search

- Grover search can search List L of size N by calling the oracle $O(\sqrt{N})$ times

- It is proven that no quantum algorithm can perform this task by calling the oracle fewer times than $O(\sqrt{N})$.

- If you are interested in the proof,
  see ”Nielsen & Chuang Quantum Computation and Quantum Information”
  Section 6.6 Optimality of the search algorithm

Summary

- Grover search is a quantum search algorithm.
  - Call the oracle only $O(\sqrt{N})$ times while classical computers need to call $O(N)$.
  - Quadratic speed up, not exponential.
- Structure of Grover search and the details of each operator.
  1. Initialization
  2. Oracle operator
  3. Diffusion operator
     Repeat 2 and 3 $O(\sqrt{N})$ times
- How to create quantum circuits for each operator.
  - Circuits for initialization, the oracle operator, and the diffusion operator.
  - Qiskit implementation
- How Grover search works
  - Addition of probability amplitudes
  - Rotations of vectors from the geometric view.
- Optimality of Grover search

Thank you