Grover’s Algorithm and its application

Presenter: Changyeol Lee

Department of Computer Science, Yonsei University

Combinatorial Optimization Lab

Problem

Given a function $f(\mathrm{x}): \{0,1\}^n \rightarrow \{0,1\}$, find a n-bit string x^* such that $f(\mathrm{x}^*) = 1$.

Let $N = 2^n$.

Requires $O(N)$ function calls in the classical model.

Grover's Algorithm (1996)

Requires $\Theta(\sqrt{N})$ function calls in the quantum model.

Recap

Recap – complex number

Complex number. $z = a + bi$ where a and b are real numbers.

- $a = Re(z)$ is the *real part* of z

- $b = Im(z)$ is the *imaginary part* of z

- $z^* \coloneqq a - bi$ is the *conjugate* of z.

- $|z| = \sqrt{Re(z)^2 + Im(z)^2} = \sqrt{a^2 + b^2}$ is the *magnitude* of z.

Observation. $|z|^2 = (a + bi)(a - bi) = z^* z$.

Recap – qubit

The *Qubit* (short for *quantum bit*). $|\phi\rangle = \alpha|0\rangle + \beta|1\rangle$

where α and β are complex numbers such that $|\alpha|^2 + |\beta|^2 = 1$.

Superposition. Measuring $|\phi\rangle$ will yield either zero w/ probability $|\alpha|^2$ or one w/ probability $|\beta|^2$.

The state of the qubit $|\phi\rangle$ is two-dimensional complex vector $\binom{\alpha}{\beta}$.

$\langle\phi| \coloneqq (\alpha \quad \beta)^* = (\alpha^* \quad \beta^*)$, i.e., the conjugate transpose of $\binom{\alpha}{\beta}$.

Recap – systems of qubit

Systems of Qubit. (Tensor product or Kronecker product)

$$|\phi_1\phi_2\rangle = |\phi_1\rangle \otimes |\phi_2\rangle = \begin{pmatrix} \alpha_1 \\ \beta_1 \end{pmatrix} \otimes \begin{pmatrix} \alpha_2 \\ \beta_2 \end{pmatrix} = \begin{pmatrix} \alpha_1\alpha_2 \\ \alpha_1\beta_2 \\ \beta_1\alpha_2 \\ \beta_1\beta_2 \end{pmatrix} = \alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \beta_1\alpha_2|10\rangle + \beta_1\beta_2|11\rangle$$

The state below is *entangled* i.e., not separable.

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

The state $|\phi'\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |11\rangle)$ is not entangled. Since

$$|\phi'\rangle = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 \\ 1 \\ 0 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |1\rangle$$

Recap – *qubits are just vectors*

Inner product. $\langle\phi_1|\phi_2\rangle = (\alpha_1 \quad \beta_1)^* \begin{pmatrix}\alpha_2 \\ \beta_2\end{pmatrix} = \alpha_1^*\alpha_2 + \beta_1^*\beta_2$ where $(\alpha_1 \quad \beta_1)^* = (\alpha_1^* \quad \beta_1^*)$.

Outer product. $|\phi_1\rangle\langle\phi_2| = \begin{pmatrix}\alpha_1 \\ \beta_1\end{pmatrix}(\alpha_2 \quad \beta_2)^* = \begin{pmatrix}\alpha_1\alpha_2^* & \alpha_1\beta_2^* \\ \beta_1\alpha_2^* & \beta_1\beta_2^*\end{pmatrix}$

$|\phi_1\rangle\langle\phi_2|$: a matrix with mapping $|\phi_1\rangle \to |\phi_2\rangle$

Exercise (expressing matrix).

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

  = Mapping $\{|0\rangle \to |0\rangle,\ |1\rangle \to -|1\rangle\}$

- $|00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10| = \begin{pmatrix}1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0\end{pmatrix}$

  = Mapping $\{|00\rangle \to |00\rangle,\ |01\rangle \to |01\rangle,\ |10\rangle \to |11\rangle,\ |11\rangle \to |10\rangle\}$

Recap – unitary matrix

Unitary Matrix. The matrix U is unitary if $UU^\dagger = U^\dagger U = I$ where $U^\dagger$ is the transposed conjugate of U.

- $U^\dagger \coloneqq {U^*}^T$ is sometimes called Hermitian conjugate matrix or adjoint matrix.

Unitary Transformation. Change of the state is done by a series of unitary transformations.

Basic unitary transformations are called *gates*.

Unitarity implies

1. #input qubits = #output qubits
2. Reversible

Recap – common gates (one-qubit gates)

Not. $\mathrm{NOT} = X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

- $\alpha|0\rangle + \beta|1\rangle \longrightarrow \boldsymbol{\beta}|0\rangle + \boldsymbol{\alpha}|1\rangle$

Hadamard. $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

Exercise.

- $|0\rangle \longrightarrow \quad \frac{|0\rangle+|1\rangle}{\sqrt{2}}$
- $|1\rangle \longrightarrow \quad \frac{|0\rangle-|1\rangle}{\sqrt{2}}$

Recap – common gates (multi-qubit gates)

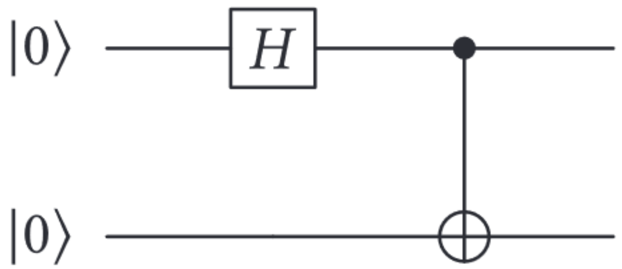
Controlled-NOT. $\mathrm{CNOT} = \begin{pmatrix} \boxed{\begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix}} & \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} \\ \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} & \boxed{\begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix}} \end{pmatrix}$

- $\alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \beta_1\alpha_2|10\rangle + \beta_1\beta_2|11\rangle \longrightarrow \alpha_1\alpha_2|00\rangle + \alpha_1\beta_2|01\rangle + \beta_1\beta_2|10\rangle + \beta_1\alpha_2|11\rangle$

Observe. $\mathrm{CNOT} = |00\rangle\langle 00| + |01\rangle\langle 01| + |10\rangle\langle 11| + |11\rangle\langle 10| = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

Recap – applying transformations

$$\mathrm{CNOT}(H \otimes I)$$



1. $(H \otimes I)(|0\rangle \otimes |0\rangle) = \dfrac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle = \dfrac{|00\rangle + |10\rangle}{\sqrt{2}}$

2. $\mathrm{CNOT}\,\frac{|00\rangle+|10\rangle}{\sqrt{2}} = \dfrac{1}{\sqrt{2}}(\mathrm{CNOT}|00\rangle + \mathrm{CNOT}|10\rangle) = \dfrac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \dfrac{|00\rangle + |11\rangle}{\sqrt{2}}$

Bell state $|\beta_{00}\rangle = \frac{|00\rangle+|11\rangle}{\sqrt{2}}$. (Note that its state is entangled.)

Quantum Oracle

Presented by Changyeol Lee

Problem

Given a function $f(\mathrm{x}): \{0,1\}^n \rightarrow \{0,1\}$, find a n-bit string x^* such that $f(\mathrm{x}^*) = 1$.

Let $N = 2^n$.

Requires $O(N)$ function calls in the classical model.

Grover's Algorithm

Requires $\Theta(\sqrt{N})$ **function calls** in the quantum model.

"Function" should be something like a quantum gate... and note that it must be **unitary**...

Quantum Oracle for Unary Function

Suppose we are given a black box unary function $f(\mathrm{x})\colon \{0,1\}^n \to \{0,1\}$.

Using this black box, one can build a new (unitary) gate that

- takes $(n+1)$-bits in and $(n+1)$-bits out,
- computes f when proper input is given, and
- has the same computational complexity.

How?

$$|\mathrm{x}\rangle|y\rangle \xrightarrow{U_f} |\mathrm{x}\rangle|f(\mathrm{x}) \oplus y\rangle$$

where $\oplus$ is integer mod-2.

x	y	$x \oplus y$
0	0	0
0	1	1
1	0	1
1	1	0

Quantum Oracle for Unary Function

Suppose we are given a constant function $f(\mathrm{x}): \{0,1\} \rightarrow 1$.

$$|\mathrm{x}\rangle|y\rangle \xrightarrow{U_f} |\mathrm{x}\rangle|1 \oplus y\rangle$$

$\|x\rangle\|y\rangle$	$U_f(\|x\rangle\|y\rangle)$
$\|0\rangle\|0\rangle$	$\|0\rangle\|1\rangle$
$\|0\rangle\|1\rangle$	$\|0\rangle\|0\rangle$
$\|1\rangle\|0\rangle$	$\|1\rangle\|1\rangle$
$\|1\rangle\|1\rangle$	$\|1\rangle\|0\rangle$

How does U_f look?

$$|00\rangle\langle 01| + |01\rangle\langle 00| + |10\rangle\langle 11| + |11\rangle\langle 10| = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

σ_x: Pauli matrix

Unitary?

- A square matrix is unitary if it can be broken down into smaller unitary matrices along its diagonal.

Quantum Oracle for Unary Function

Suppose we are given a unary function $f(\mathrm{x})\colon \{0,1\}^n \rightarrow \{0,1\}$.

Consider when $\mathrm{x} = 00\cdots 0$.

$$|\mathrm{x}\rangle|y\rangle \xrightarrow{U_f} |\mathrm{x}\rangle|f(x) \oplus y\rangle$$

$\lvert\mathrm{x}\rangle\lvert y\rangle$	$f(\mathrm{x})$	$U_f(\lvert\mathrm{x}\rangle\lvert y\rangle)$
$\lvert 00\cdots 0\rangle\lvert 0\rangle$	0	$\lvert 00\cdots 0\rangle\lvert 0\rangle$
$\lvert 00\cdots 0\rangle\lvert 1\rangle$	0	$\lvert 00\cdots 0\rangle\lvert 1\rangle$
$\lvert 00\cdots 0\rangle\lvert 0\rangle$	1	$\lvert 00\cdots 0\rangle\lvert 1\rangle$
$\lvert 00\cdots 0\rangle\lvert 1\rangle$	1	$\lvert 00\cdots 0\rangle\lvert 0\rangle$

Quantum Oracle for Unary Function

Suppose we are given a unary function $f(\mathrm{x}): \{0,1\}^n \rightarrow \{0,1\}$.

Consider when $\mathrm{x} = 00 \cdots 0$.

$$|\mathrm{x}\rangle|y\rangle \xrightarrow{U_f} |\mathrm{x}\rangle|f(x) \oplus y\rangle$$

$\|\mathrm{x}\rangle\|y\rangle$	$f(\mathrm{x})$	$U_f(\|\mathrm{x}\rangle\|y\rangle)$
$\|00\cdots 0\rangle\|0\rangle$	0	$\|00\cdots 0\rangle\|0\rangle$
$\|00\cdots 0\rangle\|1\rangle$	0	$\|00\cdots 0\rangle\|1\rangle$
$\|00\cdots 0\rangle\|0\rangle$	1	$\|00\cdots 0\rangle\|1\rangle$
$\|00\cdots 0\rangle\|1\rangle$	1	$\|00\cdots 0\rangle\|0\rangle$

When $f(\mathrm{x}) = 0$, U_f looks like

$$\left(\begin{array}{cc|c} 1 & 0 & ? \\ 0 & 1 & \\ \hline 0 & 0 & \\ \vdots & \vdots & ? \\ 0 & 0 & \end{array}\right)$$

Quantum Oracle for Unary Function

Suppose we are given a unary function $f(\mathrm{x}): \{0,1\}^n \rightarrow \{0,1\}$.

Consider when $\mathrm{x} = 00 \cdots 0$.

$$|\mathrm{x}\rangle|y\rangle \xrightarrow{U_f} |\mathrm{x}\rangle|f(x) \oplus y\rangle$$

$\|\mathrm{x}\rangle\|y\rangle$	$f(\mathrm{x})$	$U_f(\|\mathrm{x}\rangle\|y\rangle)$
$\|00\cdots0\rangle\|0\rangle$	0	$\|00\cdots0\rangle\|0\rangle$
$\|00\cdots0\rangle\|1\rangle$	0	$\|00\cdots0\rangle\|1\rangle$
$\|00\cdots0\rangle\|0\rangle$	1	$\|00\cdots0\rangle\|1\rangle$
$\|00\cdots0\rangle\|1\rangle$	1	$\|00\cdots0\rangle\|0\rangle$

When $f(\mathrm{x}) = 1$, U_f looks like

$$\left(\begin{array}{cc|c} 0 & 1 & \\ 1 & 0 & ? \\ \hline 0 & 0 & \\ \vdots & \vdots & ? \\ 0 & 0 & \end{array}\right)$$

Quantum Oracle for Unary Function

Suppose we are given a unary function $f(\mathrm{x})\colon \{0,1\}^n \to \{0,1\}$.

$$|\mathrm{x}\rangle|y\rangle \xrightarrow{U_f} |\mathrm{x}\rangle|f(x) \oplus y\rangle$$

Then U_f looks like

$$\begin{pmatrix} [\mathrm{I\ or\ }\sigma_x] & & & & \\ & [\mathrm{I\ or\ }\sigma_x] & & & \\ & & \ddots & & \\ & & & [\mathrm{I\ or\ }\sigma_x] & \\ & & & & [\mathrm{I\ or\ }\sigma_x] \end{pmatrix}$$

and it is unitary.

Quantum Algorithm’s Complexity

Relativized Time Complexity

- The time complexity **without** knowledge of the oracle’s design. (If we allow small error, no speed-up.)

- Deutsch-Jozsa’s algorithm offers a deterministic exponential speed-up *relative to the oracle.*

- Bernstein-Vazirani’s algorithm offers a polynomial speed-up *relative to the oracle.* (even if small error is allowed)

Absolute Time Complexity

- The time complexity **with** knowledge of the oracle’s design

- Shor’s algorithm provides *absolute* speed-up.

Problem

Given a function $f(\mathrm{x}): \{0,1\}^n \rightarrow \{0,1\}$, find **the $\boldsymbol{n}$-bit target string** x^* such that $f(\mathrm{x}^*) = 1$.

Let $N = 2^n$.

Requires $O(N)$ function calls in the classical model.

Given a quantum oracle $\boldsymbol{O_f}$, find a (n-bit) target string x^*.

Grover's Algorithm. Requires $\Theta(\sqrt{N})$ calls to the **quantum oracle**.

Grover’s Algorithm

Presented by Changyeol Lee

Grover operator G

Let $|\psi\rangle \coloneqq \frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + |00\cdots 10\rangle + \cdots + |11\cdots 11\rangle)$ be the uniform superposition.

(Shorthand) $|\psi\rangle = \frac{1}{\sqrt{N}}\sum_{i=0}^{N-1}|i\rangle = \frac{1}{\sqrt{N}}\sum_i |i\rangle$

$$|\psi\rangle\langle\psi| = \frac{1}{N}\begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ 1 & 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{pmatrix}$$

Grover operator $G \coloneqq \big((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\big)O_f$

The action of $2|\psi\rangle\langle\psi| - I_N$ on an arbitrary state $|\phi\rangle = \sum_i a_i|i\rangle = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{N-1} \end{pmatrix}$

$$I_N = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

$$(2|\psi\rangle\langle\psi| - I_N)|\phi\rangle = \sum_i \left(2\frac{a_0 + \cdots + a_{N-1}}{N} - a_i\right)|i\rangle$$

$$I_2 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Grover's Algorithm

Step 1. Perform state initialization

- (n qubits) $|00\cdots 0\rangle \longrightarrow \frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + |00\cdots 10\rangle + \cdots + |11\cdots 11\rangle)$

- (ancillary qubit) $|0\rangle \longrightarrow \frac{|0\rangle - |1\rangle}{\sqrt{2}}$

Step 2. Apply Grover operator $\left\lfloor \frac{\pi\sqrt{N}}{4} \right\rfloor$ times

Step 3. Perform measurement on all qubit (except the ancillary qubit)

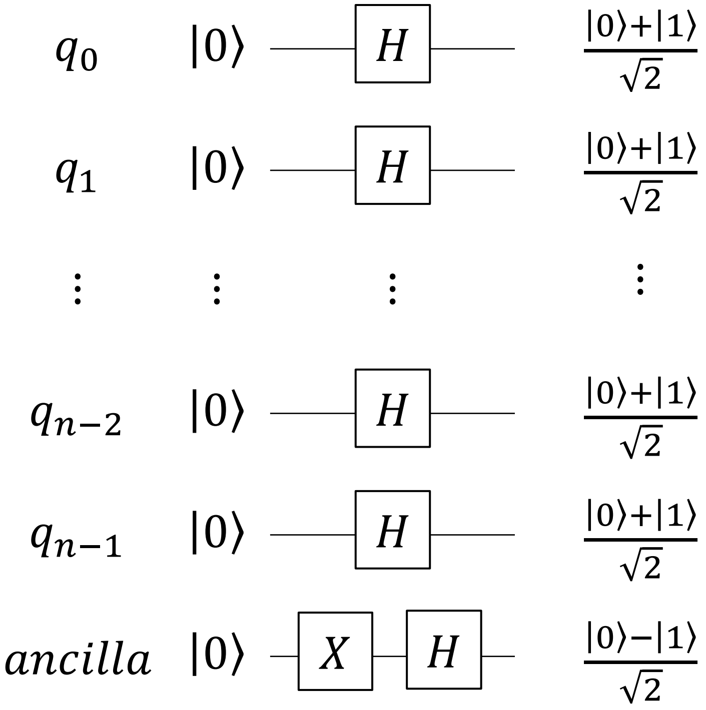
Grover's Algorithm

Step 1. Initialization

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

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

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



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

$$= \frac{1}{\sqrt{N}}(|00\cdots00\rangle + |00\cdots01\rangle + |00\cdots10\rangle + \cdots + |11\cdots11\rangle)$$

$$\frac{1}{\sqrt{N}}(|00\cdots00\rangle + |00\cdots01\rangle + |00\cdots10\rangle + \cdots + |11\cdots1\rangle) \otimes \frac{|0\rangle-|1\rangle}{\sqrt{2}}$$

Grover's Algorithm

Step 2. Apply $G \coloneqq \big((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\big) O_f$

$$|\mathrm{x}\rangle|q\rangle \longrightarrow |\mathrm{x}\rangle|f(\mathrm{x}) \oplus q\rangle$$

$$O_f\left(\frac{1}{\sqrt{N}}(|00\cdots00\rangle + |00\cdots01\rangle + \cdots + |\mathrm{x}^*\rangle + \cdots + |11\cdots11\rangle) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

$$= \frac{1}{\sqrt{N}}\left(O_f\left(|00\cdots00\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) + \cdots + \underline{O_f\left(|\mathrm{x}^*\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)} + \cdots + O_f\left(|11\cdots11\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)\right)$$

$$= O_f\frac{|\mathrm{x}^*\rangle|0\rangle - |\mathrm{x}^*\rangle|1\rangle}{\sqrt{2}} = \frac{O_f(|\mathrm{x}^*\rangle|0\rangle) - O_f(|\mathrm{x}^*\rangle|1\rangle)}{\sqrt{2}}$$

$$= \frac{|\mathrm{x}^*\rangle|1\rangle - |\mathrm{x}^*\rangle|0\rangle}{\sqrt{2}}$$

$$= |\mathrm{x}^*\rangle \otimes \frac{|1\rangle - |0\rangle}{\sqrt{2}}$$

Grover's Algorithm

Step 2. Apply $G \coloneqq \big((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\big)O_f$

$$|\mathrm{x}\rangle|q\rangle \longrightarrow |\mathrm{x}\rangle|f(\mathrm{x}) \oplus q\rangle$$

$$O_f\left(\frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + \cdots + |\mathrm{x}^*\rangle + \cdots + |11\cdots 11\rangle) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

$$= \frac{1}{\sqrt{N}}\left(O_f\left(|00\cdots 00\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) + \cdots + O_f\left(|\mathrm{x}^*\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) + \cdots + O_f\left(|11\cdots 11\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)\right)$$

$$= \frac{1}{\sqrt{N}}\left(|00\cdots 00\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}} + \cdots + |\mathrm{x}^*\rangle \otimes \frac{-|\mathbf{0}\rangle + |\mathbf{1}\rangle}{\sqrt{\mathbf{2}}} + \cdots + |11\cdots 11\rangle \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

$$= \frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + \cdots + (-\mathbf{1})|\mathrm{x}^*\rangle + \cdots + |11\cdots 11\rangle) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Grover's Algorithm

Step 2. Apply $G := \left((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\right)O_f$

$$\left((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\right) \left(\frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + \cdots + (-1)|\mathrm{x}^*\rangle + \cdots + |11\cdots 11\rangle) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

$$= (2|\psi\rangle\langle\psi| - I_N)\left(\frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + \cdots + (-1)|\mathrm{x}^*\rangle + \cdots + |11\cdots 11\rangle)\right) \otimes \left(I_2\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

$$= (2|\psi\rangle\langle\psi| - I_N)\left(\frac{1}{\sqrt{N}}(|00\cdots 00\rangle + |00\cdots 01\rangle + \cdots + (-1)|\mathrm{x}^*\rangle + \cdots + |11\cdots 11\rangle)\right) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Grover's Algorithm

Step 2. Apply $G \coloneqq \big((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\big) O_f$

$$(2|\psi\rangle\langle\psi| - I_N)\underline{|\phi\rangle} = \sum_i \left(\frac{2}{N}\underline{(a_0 + \cdots + a_{N-1})} - a_i\right)|i\rangle$$

$$\boxed{\frac{N-2}{\sqrt{N}}}$$

$$(2|\psi\rangle\langle\psi| - I_N)\underline{\left(\frac{1}{\sqrt{N}}(|00\cdots00\rangle + |00\cdots01\rangle + \cdots + (-1)|\mathrm{x}^*\rangle + \cdots + |11\cdots11\rangle)\right)}$$

$$= \left(\frac{2}{N}\frac{N-2}{\sqrt{N}} - \frac{1}{\sqrt{N}}\right)|00\cdots00\rangle + \cdots + \left(\frac{2}{N}\frac{N-2}{\sqrt{N}} + \frac{\mathbf{1}}{\sqrt{\boldsymbol{N}}}\right)|\mathrm{x}^*\rangle + \cdots + \left(\frac{2}{N}\frac{N-2}{\sqrt{N}} - \frac{1}{\sqrt{N}}\right)|11\cdots11\rangle$$

$$= \frac{1}{\sqrt{N}}\left(\frac{N-4}{N}|00\cdots00\rangle + \cdots + \underbrace{\frac{3N-4}{N}}_{\text{amplified}}|\mathrm{x}^*\rangle + \cdots + \frac{N-4}{N}|11\cdots11\rangle\right)$$

Grover's Algorithm

Step 2. Apply $G \coloneqq \left((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\right) O_f$ again

$$|\mathrm{x}\rangle|q\rangle \longrightarrow |\mathrm{x}\rangle|f(\mathrm{x}) \oplus q\rangle$$

$$O_f\left(\frac{1}{\sqrt{N}}\left(\frac{N-4}{N}|00\cdots00\rangle + \cdots + \frac{3N-4}{N}|\mathrm{x}^*\rangle + \cdots + \frac{N-4}{N}|11\cdots11\rangle\right) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}\right)$$

$$= \frac{1}{\sqrt{N}}\left(\frac{N-4}{N}|00\cdots00\rangle + \cdots + \underbrace{(-\mathbf{1})\frac{3N-4}{N}|\mathrm{x}^*\rangle}_{\text{flipped}} + \cdots + \frac{N-4}{N}|11\cdots11\rangle\right) \otimes \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

Grover's Algorithm

Step 2. Apply $G \coloneqq \left((2|\psi\rangle\langle\psi| - I_N) \otimes I_2\right)O_f$ again

$$\boxed{(2|\psi\rangle\langle\psi| - I_N)|\phi\rangle = \sum_i \left(\frac{2}{N}(a_0 + \cdots + a_{N-1}) - a_i\right)|i\rangle}$$

$$(2|\psi\rangle\langle\psi| - I_N)\left(\frac{1}{\sqrt{N}}\left(\frac{N-4}{N}|00\cdots 00\rangle + \cdots + (-1)\frac{3N-4}{N}|\mathrm{x}^*\rangle + \cdots + \frac{N-4}{N}|11\cdots 11\rangle\right)\right)$$

$$= \frac{1}{\sqrt{N}}\left(\frac{N^2-12N+16}{N^2}|00\cdots 00\rangle + \cdots + \underbrace{\frac{5N^2-20N+16}{N^2}}_{\text{more amplified}}|\mathrm{x}^*\rangle + \cdots + \frac{N^2-12N+16}{N^2}|11\cdots 11\rangle\right)$$

Grover's Algorithm

Step 2. Apply G **fixed amount**

$$\text{(informally)} \quad \frac{1}{\sqrt{N}}\left(\epsilon|00\cdots00\rangle+\cdots+\underbrace{\left(\sqrt{N}-\epsilon'\right)}_{\text{amplified a lot}}|\mathrm{x}^*\rangle+\cdots+\epsilon|11\cdots11\rangle\right)$$

for some small ϵ, ϵ'.

Grover's Algorithm

Step 3. Measurement

$$\text{(informally)} \quad \frac{1}{\sqrt{N}}\left(\epsilon|00\cdots 00\rangle + \cdots + \left(\sqrt{N}-\epsilon'\right)|\mathrm{x}^*\rangle + \cdots + \epsilon|11\cdots 11\rangle\right)$$

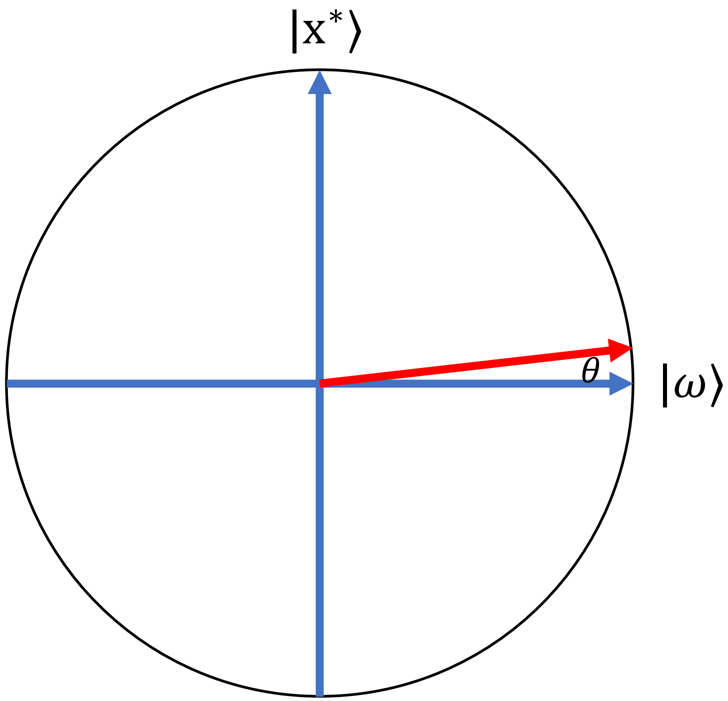
Obtain $|x^*\rangle$ with probability close to 1.

Analysis

Presented by Changyeol Lee

Grover's Algorithm, Geometric Explanation

Why applying Grover operator (exactly) $\left\lceil \frac{\pi\sqrt{N}}{4} \right\rceil$ times?



Let $|\omega\rangle = \frac{1}{\sqrt{N-1}}\left(\sum_i |i\rangle - |x^*\rangle\right)$

Note that $|\omega\rangle$ and $|x^*\rangle$ are orthonormal.

Note. After the step 1, the state is

$$\frac{1}{\sqrt{N}}(|00\cdots00\rangle + |00\cdots01\rangle + |00\cdots10\rangle + \cdots + |11\cdots1\rangle)$$

$$= \frac{\sqrt{N-1}}{\sqrt{N}}|\omega\rangle + \frac{1}{\sqrt{N}}|x^*\rangle$$

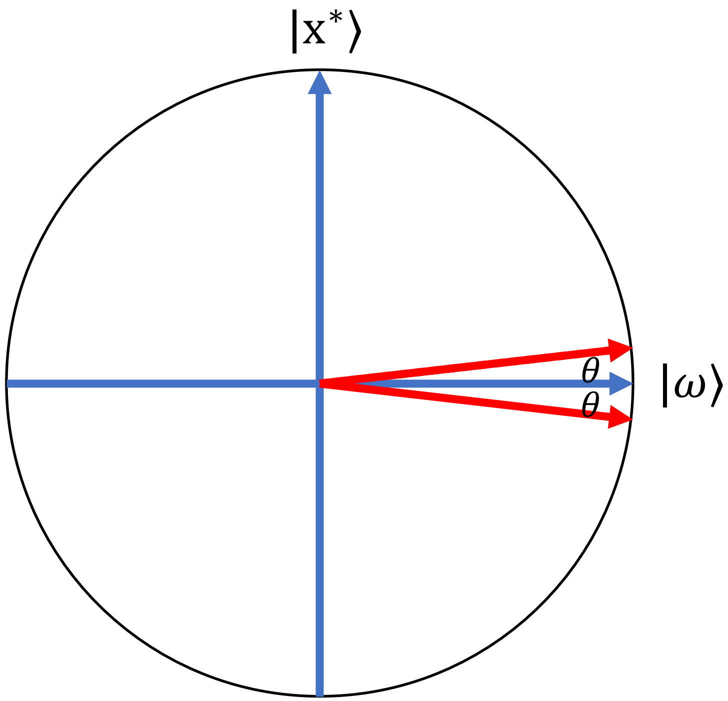
$$= \cos\theta\,|\omega\rangle + \sin\theta\,|x^*\rangle$$

Grover's Algorithm, Geometric Explanation

What happens we apply O_f?

$$\cos\theta\,|\omega\rangle + \sin\theta\,|\mathrm{x}^*\rangle \longrightarrow \cos\theta\,|\omega\rangle - \sin\theta\,|\mathrm{x}^*\rangle$$

Applying O_f =Reflection about $|\omega\rangle$



Grover's Algorithm, Geometric Explanation

What happens we apply $(2|\psi\rangle\langle\psi| - I_N)$?

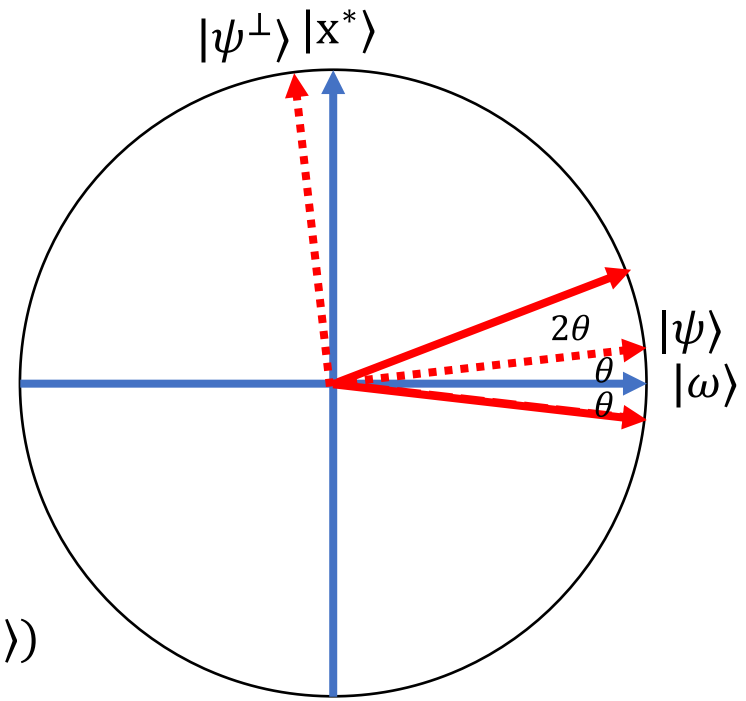
Any state $|\phi\rangle$ of this plane can be decomposed into

$$|\phi\rangle = \alpha|\psi\rangle + \beta|\psi^{\perp}\rangle$$

Then,

$$\begin{aligned}
&(2|\psi\rangle\langle\psi| - I_N)|\phi\rangle \\
&= 2|\psi\rangle\langle\psi|(\alpha|\psi\rangle + \beta|\psi^{\perp}\rangle) - (\alpha|\psi\rangle + \beta|\psi^{\perp}\rangle) \\
&= 2\alpha|\psi\rangle\langle\psi||\psi\rangle + 2\beta|\psi\rangle\langle\psi||\psi^{\perp}\rangle - (\alpha|\psi\rangle + \beta|\psi^{\perp}\rangle) \\
&= 2\alpha|\psi\rangle - (\alpha|\psi\rangle + \beta|\psi^{\perp}\rangle) \\
&= \boldsymbol{\alpha|\psi\rangle - \beta|\psi^{\perp}\rangle}
\end{aligned}$$

Applying $(2|\psi\rangle\langle\psi| - I_N)$ = Reflection about $|\psi\rangle$



$|\psi\rangle\langle\psi|$: a mapping $|\psi\rangle \rightarrow |\psi\rangle$

inner product $\langle\psi||\psi^{\perp}\rangle = 0$

Grover's Algorithm, Geometric Explanation

After first iteration,

$$\cos\theta\,|\omega\rangle + \sin\theta\,|\mathrm{x}^*\rangle \longrightarrow \cos 3\theta\,|\omega\rangle + \sin 3\theta\,|\mathrm{x}^*\rangle$$

After each iteration,

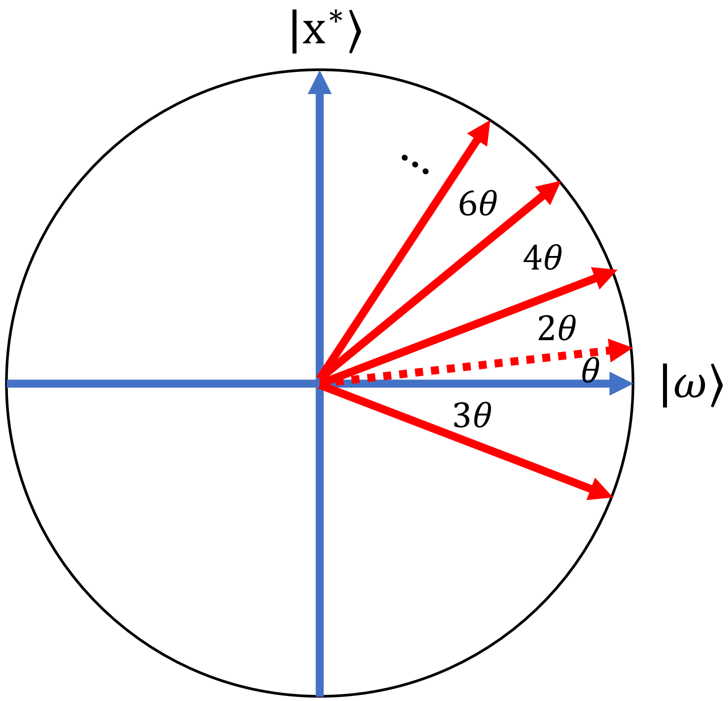
$$\cos 5\theta\,|\omega\rangle + \sin 5\theta\,|\mathrm{x}^*\rangle$$

$$\cos 7\theta\,|\omega\rangle + \sin 7\theta\,|\mathrm{x}^*\rangle$$

$$\vdots$$

After applying k times,

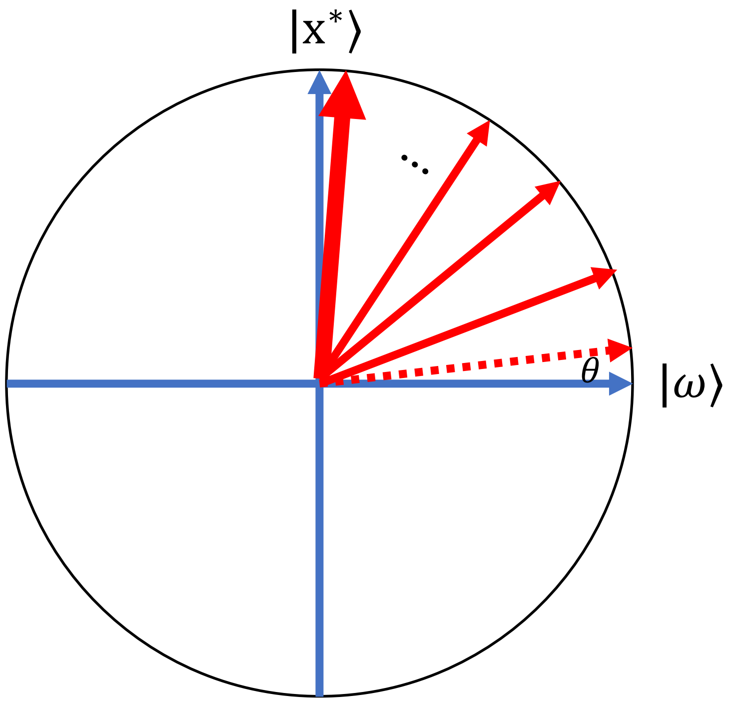
$$\cos(\theta + 2k\theta)\,|\omega\rangle + \sin(\theta + 2k\theta)\,|\mathrm{x}^*\rangle$$



Grover's Algorithm, Geometric Explanation

Recall $\frac{\sqrt{N-1}}{\sqrt{N}}|\omega\rangle + \frac{1}{\sqrt{N}}|\mathrm{x}^*\rangle = \cos\theta\,|\omega\rangle + \sin\theta\,|\mathrm{x}^*\rangle$.

- $\theta = \arccos\sqrt{\frac{N-1}{N}}$



Find k that maximizes $\sin^2(\theta + 2k\theta) = \sin^2\left((2k+1)\arccos\sqrt{\frac{N-1}{N}}\right)$

or find k such that $\frac{\pi}{2} \sim (2k+1)\arccos\sqrt{\frac{N-1}{N}}$

$$k_{optimal} = \frac{\pi}{4}\sqrt{N} - \frac{1}{2} - O\left(\sqrt{1/N}\right)$$

Applications

Factoring

Factor an integer $M = p \times q$ into p and q where p and q are primes ($2^{2n-1} \le M < 2^{2n}$ for some n)

WLOG, $p \le \sqrt{M} < 2^n$.

Given a function $f(x)$ that outputs 1 if $x = p$; 0 otherwise

Given a function $f(\mathrm{x})\colon \{0,1\}^n \to \{0,1\}$, find the target string p.

<u>Grover's Algorithm</u>.

Requires $\frac{\pi}{4}\sqrt{N} - \frac{1}{2} - O\left(\sqrt{1/N}\right)$ calls to the quantum oracle. ($N = 2^n$)

<u>Shor's algorithm</u> runs in $O(n^3 \log n)$ times, using $O(n^2 \log n \log\log n)$ gates.

Multiple Target Strings

If there are t number of target strings, by calling $k_{optimal} = \frac{\pi}{4}\sqrt{N/t} - \frac{1}{2} - O\left(\sqrt{t/N}\right)$

or $\Theta\left(\sqrt{N/t}\right)$ calls to the oracle, we can find a target string.

What if the number of target strings t is unknown?

Theorem (Boyer et al., 1996).

There is an (randomized) algorithm that find a target string in expected time in $O\left(\sqrt{N/t}\right)$.

Minimum Searching

Given a function $F(\mathrm{x})\colon \{0,1\}^n \to \mathbb{R}$, find a n-bit string x^* that minimizes $F(\mathrm{x}^*)$.

- We are given an oracle that "compares" F with some constant c.

E.g.) $F(0) = 9, F(1) = 5, F(2) = 6, F(3) = 7$ (Here, $n = 2$.)

Suppose $c = F(3) = 7$.

Then, $f(0) = 0, f(1) = 1, f(2) = 1, f(3) = 0$.

Construct a Grover-Search-Problem instance, i.e., given a function $f(\mathrm{x})\colon \{0,1\}^n \to \{0,1\}$,

find a n-bit string x^* such that $f(\mathrm{x}^*) = 1$ where the number of target strings, say t, is unknown.

$\Rightarrow O\left(\sqrt{N/t}\right)$ calls to the oracle (in expectation).

E.g.) output = 2

Minimum Searching

Given a function $F(\mathrm{x})\colon \{0,1\}^n \to \mathbb{R}$, find a n-bit string x^* that minimizes $F(\mathrm{x}^*)$.

- We are given an oracle that "compares" F with some constant c.

E.g.) $F(0) = 9, F(1) = 5, F(2) = 6, F(3) = 7$ (Here, $n = 2$.)

Suppose $c = \boldsymbol{F(2)} = 6$.

Then, $f(0) = 0, f(1) = 1, f(2) = 0, f(3) = 0$.

Construct a Grover-Search-Problem instance, i.e., given a function $f(\mathrm{x})\colon \{0,1\}^n \to \{0,1\}$,

find a n-bit string x^* such that $f(\mathrm{x}^*) = 1$ where the number of target strings, say t, is unknown.

$\Rightarrow O\left(\sqrt{N/t}\right)$ calls to the oracle (in expectation).

E.g.) output = 1

Minimum Searching

Given a function $F(\mathrm{x})\colon \{0,1\}^n \to \mathbb{R}$, find a n-bit string x^* that minimizes $F(\mathrm{x}^*)$.

- We are given an oracle that “compares” F with some constant c.

E.g.) $F(0) = 9, F(1) = 5, F(2) = 6, F(3) = 7$ (Here, $n = 2$.)

Suppose $c = F(\mathbf{1}) = 5$.

Then, $f(0) = 0, f(1) = 0, f(2) = 0, f(3) = 0$.

...

Minimum Searching

Given a function $F(\mathrm{x})\colon \{0,1\}^n \to \mathbb{R}$, find a n-bit string x^* that minimizes $F(\mathrm{x}^*)$.

- We are given an oracle that "compares" F with some constant c.

Step 1. Pick an index j uniformly at random among $\{0,1,\cdots,N-1\}$

Step 2. Repeat the following until the total running time is more than $\Theta(\sqrt{N})$:

Let O_f be the comparison oracle with $c = F(j)$

Construct a Grover-Search-Problem instance and run the algorithm.

Let j' be the output. Update j to j' if $F(j') < F(j)$.

Theorem (Dürr and Høyer, 1996).

This algorithm finds the index of the minimum value of F with probability at least ½ and runs in $\Theta(\sqrt{N})$.

Minimum Searching with Different Types

Given a function $F(\mathrm{x}): \{0,1\}^n \to \mathbb{R}$ and an onto function $\mathrm{type}(\mathrm{x}): \{0,1\}^n \to \{1,2,\cdots,d\}$,

find $\{\mathrm{x}_1^*, \cdots, \mathrm{x}_d^*\}$ where $\mathrm{x}_i^* = \mathrm{argmin}_{\mathrm{type}(\mathrm{x})=i} F(\mathrm{x})$ for each type $i = 1,2,\cdots,d$.

Naïve application of the previous algorithm – run d times for each type, having $\Theta\left(d\sqrt{N}\right)$ running time.

Theorem (Dürr et al., 2006).

There is an algorithm solves the problem with probability at least ½ and runs in $O\left(\sqrt{dN}\right)$.

Minimum Spanning Tree

Quantumize the classical (Boruvka's algorithm) MST algorithm

input : Adjacency list-array implementation of undirected $G = (V, E)$ with cost function $c: E \to \mathbb{R}$

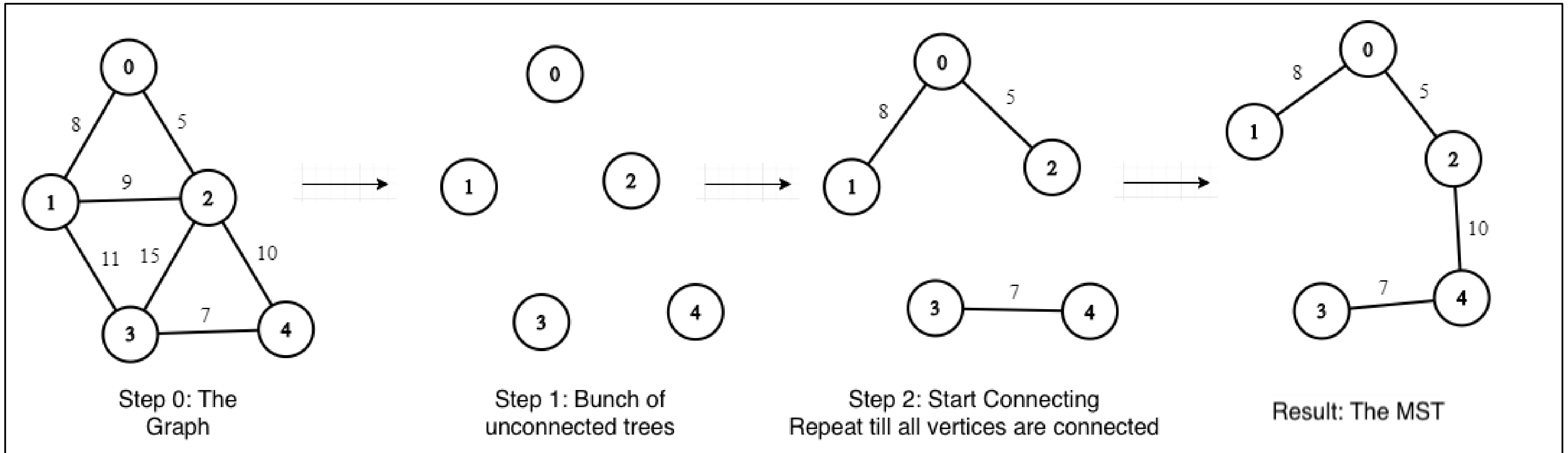
Initialization $\mathcal{T} = \{\{1\}, \{2\}, \cdots, \{|V|\}\}$

Repeat until $|\mathcal{T}| = 1$:

Let $\mathcal{T} = \{T_1, \cdots, T_k\}$. Find $e_1, \cdots, e_k$ where e_i is a minimum cost edge leaving T_i.

Merge T_i and e_i for each i and update $\mathcal{T}$.

Return T_1



https://www.baeldung.com/java-boruvka-algorithm

Minimum Spanning Tree

Quantumize the classical (Boruvka's algorithm) MST algorithm

input : Adjacency list-array implementation of undirected $G = (V, E)$ with cost function $c: E \to \mathbb{R}$

Initialization $\mathcal{T} = \{\{1\}, \{2\}, \cdots, \{|V|\}\}$

Repeat until $|\mathcal{T}| = 1$:

Let $\mathcal{T} = \{T_1, \cdots, T_k\}$. Find $e_1, \cdots, e_k$ where e_i is a minimum cost edge leaving T_i.

Merge T_i and e_i for each i and update $\mathcal{T}$.

Return T_1

Consider the directed version of the graph, i.e., $(u, v) \to \langle u, v \rangle, \langle v, u \rangle$

$F(\langle u, v \rangle) = c\big((u, v)\big)$ if $\langle u, v \rangle$ is leaving some tree and $F(\langle u, v \rangle) = \infty$ if not.

$\mathrm{type}(\langle u, v \rangle) = i$ such that $u \in T_i$

Then, apply the algorithm for the **Minimum Searching with Different Types** (with *more* queries).

Minimum Spanning Tree

Let $n = |V|$ and $m = |E|$.

On ℓ^{th} iteration, queries $(\ell + 2)O\left(\sqrt{mk}\right)$ times.

#of queries

- Observe that $k \leq n/2^{\ell-1}$.

$$\sum_{\ell \geq 1} (\ell + 2)O\left(\sqrt{mk}\right) \leq \sum_{\ell \geq 1} (\ell + 2)O\left(\sqrt{\frac{mn}{2^{\ell-1}}}\right) = O(\sqrt{nm})$$

Error probability at most

$$\sum_{\ell \geq 1} \frac{1}{2^{\ell+2}} \leq \frac{1}{4}$$

Other applications

Connectivity

Network flow problem. (Finding Max Flow)

Matching on graph

Graph coloring

3-SAT

⋮

approximation algorithms

Thank you