Quantum Computing

András Gilyén

Summer School in Post-Quantum Cryptography, 2nd & 4th August 2022

4 postulates of quantum mechanics

1: (Pure) states are unit-length vectors of a Hilbert space

We consider the finite-dimensional case. (Complex Euclidean vector space.)

1: (Pure) states are unit-length vectors of a Hilbert space

We consider the finite-dimensional case. (Complex Euclidean vector space.)

- E.g.: a qubit has state space $\mathbb{C}^2 = \mathrm{Span}\{|0\rangle, |1\rangle\}$

$$|\psi\rangle = \underbrace{a|0\rangle + b|1\rangle}_{\text{superposition}} \text{ such that } |a|^2 + |b|^2 = 1$$

1: (Pure) states are unit-length vectors of a Hilbert space

We consider the finite-dimensional case. (Complex Euclidean vector space.)

- E.g.: a qubit has state space $\mathbb{C}^2 = \mathrm{Span}\{|0\rangle, |1\rangle\}$

$$|\psi\rangle = \underbrace{a|0\rangle + b|1\rangle}_{\text{superposition}} \text{ such that } |a|^2 + |b|^2 = 1$$

- Dirac notation: ket vector $|\psi\rangle = \begin{pmatrix} a \\ b \end{pmatrix} = a|0\rangle + b|1\rangle$ — a, b are called amplitudes

  bra vector $\langle\psi| = |\psi\rangle^\dagger = \begin{pmatrix} \bar{a} & \bar{b} \end{pmatrix}$

1: (Pure) states are unit-length vectors of a Hilbert space

We consider the finite-dimensional case. (Complex Euclidean vector space.)

- E.g.: a qubit has state space $\mathbb{C}^2 = \mathrm{Span}\{|0\rangle, |1\rangle\}$

$$|\psi\rangle = \underbrace{a|0\rangle + b|1\rangle}_{\text{superposition}} \text{ such that } |a|^2 + |b|^2 = 1$$

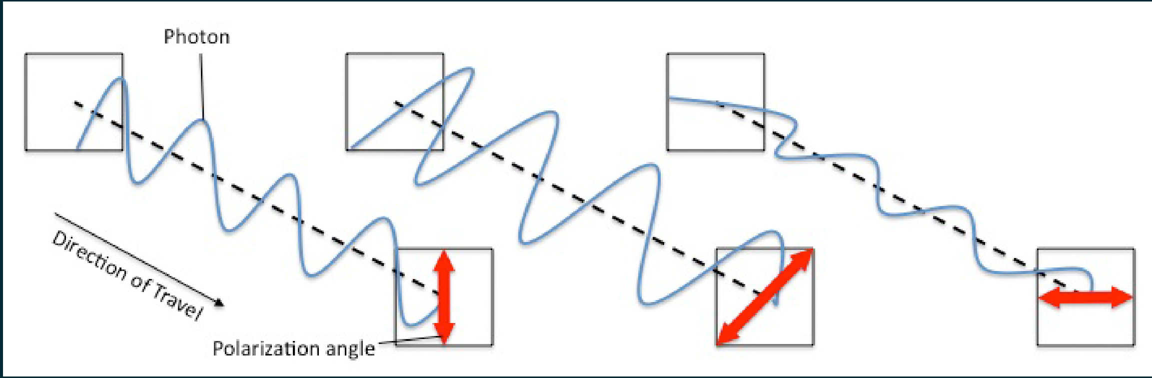
- Dirac notation: ket vector $|\psi\rangle = \begin{pmatrix} a \\ b \end{pmatrix} = a|0\rangle + b|1\rangle$ — a, b are called amplitudes

  bra vector $\langle\psi| = |\psi\rangle^\dagger = \begin{pmatrix} \overline{a} & \overline{b} \end{pmatrix}$

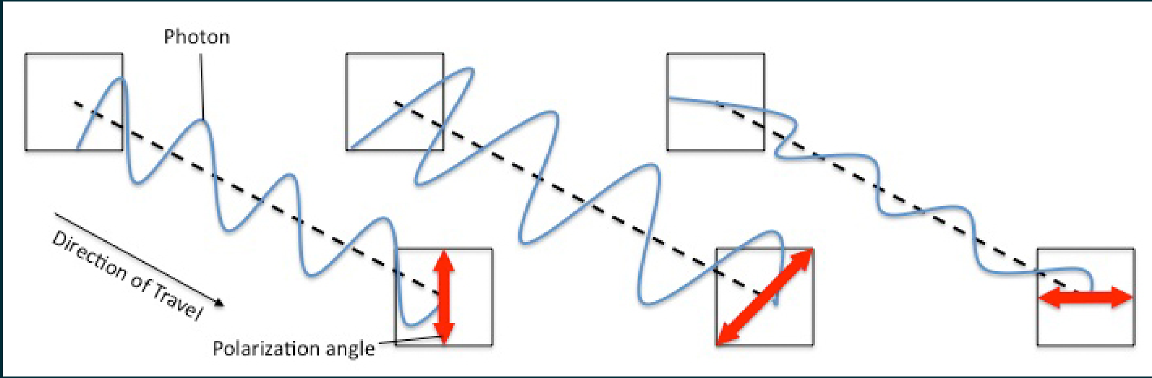
  Motivation: inner product notation (let $|\phi\rangle = c|0\rangle + d|1\rangle$)

$$\langle\psi, \phi\rangle = \langle\psi| \cdot |\phi\rangle = \overline{a}c + \overline{b}d$$

1: Example qubit — photon polarization



1: Example qubit — photon polarization



Note: two states are “fully” distinguishable iff they are orthogonal

2: Composite systems are formed by tensor products

$|v\rangle \in V, |w\rangle \in W \implies$ Joint state: $|v\rangle \otimes |w\rangle \in V \otimes W$

2: Composite systems are formed by tensor products

$|v\rangle \in V, |w\rangle \in W \implies$ Joint state: $|v\rangle \otimes |w\rangle \in V \otimes W$

- E.g.: n qubits have state space $\mathbb{C}^{2^n} = \mathrm{Span}\{|i\rangle : i \in \{0,1\}^n\}$ (computational basis)

2: Composite systems are formed by tensor products

$|v\rangle \in V, |w\rangle \in W \implies$ Joint state: $|v\rangle \otimes |w\rangle \in V \otimes W$

- E.g.: n qubits have state space $\mathbb{C}^{2^n} = \mathrm{Span}\{|i\rangle : i \in \{0,1\}^n\}$ (computational basis)

$$2 \text{ qubits: } |0\rangle \otimes |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}; \; |0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}; \ldots$$

$$\text{Generic two-qubit state } |\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle) = \begin{pmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{pmatrix}$$

2: Composite systems are formed by tensor products

$|v\rangle \in V, |w\rangle \in W \Longrightarrow$ Joint state: $|v\rangle \otimes |w\rangle \in V \otimes W$

- E.g.: n qubits have state space $\mathbb{C}^{2^n} = \mathrm{Span}\{|i\rangle : i \in \{0,1\}^n\}$ (computational basis)

$$2 \text{ qubits: } |0\rangle \otimes |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}; \ |0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}; \ldots$$

$$\text{Generic two-qubit state } |\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle) = \begin{pmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{pmatrix}$$

- $|v\rangle \otimes |w\rangle$ is a <u>product state</u>; non-product sates are called <u>entangled</u>

2: Composite systems are formed by tensor products

$|v\rangle \in V, |w\rangle \in W \implies$ Joint state: $|v\rangle \otimes |w\rangle \in V \otimes W$

- E.g.: n qubits have state space $\mathbb{C}^{2^n} = \mathrm{Span}\{|i\rangle : i \in \{0,1\}^n\}$ (computational basis)

  $$2 \text{ qubits: } |0\rangle \otimes |0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}; \; |0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}; \ldots$$

  $$\text{Generic two-qubit state } |\psi\rangle = a_{00}|00\rangle + a_{01}|01\rangle + a_{10}|10\rangle + a_{11}|11\rangle) = \begin{pmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{pmatrix}$$

- $|v\rangle \otimes |w\rangle$ is a product state; non-product sates are called entangled

  E.g., Einstein-Podolsky-Rosen (EPR)-pair: $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

3: Measurement is described by Born's rule

Measure state $|\psi\rangle = \sum_{i=0}^{N-1} a_i|i\rangle$ in orthonormal basis $(|0\rangle, |1\rangle, \ldots, |N-1\rangle) \implies$

We get outcome i with probability $|a_i|^2$, and the state collapses to $|i\rangle$

3: Measurement is described by Born's rule

Measure state $|\psi\rangle = \sum_{i=0}^{N-1} a_i|i\rangle$ in orthonormal basis $(|0\rangle, |1\rangle, \ldots, |N-1\rangle) \Longrightarrow$

We get outcome i with probability $|a_i|^2$, and the state collapses to $|i\rangle$

- Extended Born's rule for partial measurememts:

$$|\psi\rangle = \sum_{i=0}^{N-1} |i\rangle \otimes |\phi_i\rangle \Longrightarrow \Pr(\text{outcome } i) = ||\phi_i||^2 \text{ \& state “collapses” to } \frac{|i\rangle \otimes |\phi_i\rangle}{||\phi_i||}$$

3: Measurement is described by Born's rule

Measure state $|\psi\rangle = \sum_{i=0}^{N-1} a_i|i\rangle$ in orthonormal basis $(|0\rangle, |1\rangle, \ldots, |N-1\rangle) \Longrightarrow$

We get outcome i with probability $|a_i|^2$, and the state collapses to $|i\rangle$

- Extended Born's rule for partial measurememts:

$$|\psi\rangle = \sum_{i=0}^{N-1} |i\rangle \otimes |\phi_i\rangle \Longrightarrow \Pr(\text{outcome } i) = \|\phi_i\|^2 \ \&\ \text{ state "collapses" to } \frac{|i\rangle \otimes |\phi_i\rangle}{\|\phi_i\|}$$

- Analogous to conditioning probability distributions

$$\Pr(\text{outcome } ij) = \underbrace{\Pr(\text{outcome } j|i)}_{\text{determined by the collapsed state}} \cdot \Pr(\text{outcome } i)$$

3: Measurement is described by Born's rule

Measure state $|\psi\rangle = \sum_{i=0}^{N-1} a_i|i\rangle$ in orthonormal basis $(|0\rangle, |1\rangle, \ldots, |N-1\rangle) \Longrightarrow$

We get outcome i with probability $|a_i|^2$, and the state collapses to $|i\rangle$

- Extended Born's rule for partial measurememts:

$$|\psi\rangle = \sum_{i=0}^{N-1} |i\rangle \otimes |\phi_i\rangle \Longrightarrow \Pr(\text{outcome } i) = \|\phi_i\|^2 \ \& \ \text{state “collapses” to } \frac{|i\rangle \otimes |\phi_i\rangle}{\|\phi_i\|}$$

- Analogous to conditioning probability distributions

$$\Pr(\text{outcome } ij) = \underbrace{\Pr(\text{outcome } j|i)}_{\text{determined by the collapsed state}} \cdot \Pr(\text{outcome } i)$$

- More general projective measurement:

$$\Pi_j \text{ orth. projectors s.t. } I = \sum_j \Pi_j \Longrightarrow \Pr(\text{outcome } j) = \left\|\Pi_j|\psi\rangle\right\|^2 \text{ collapse: } \frac{\Pi_j|\psi\rangle}{\left\|\Pi_j|\psi\rangle\right\|}$$

4: “Time-evolution” is described by unitary operators

Linear map mapping states to states $\Longleftrightarrow$ unitary operator

4: “Time-evolution” is described by unitary operators

Linear map mapping states to states $\iff$ unitary operator

- Quantum algorithm: unitary matrix U (i.e., $U^\dagger U = I = UU^\dagger$)

4: “Time-evolution” is described by unitary operators

Linear map mapping states to states $\Longleftrightarrow$ unitary operator

- Quantum algorithm: unitary matrix U (i.e., $U^\dagger U = I = UU^\dagger$)

Quantum Circuits and algorithms

Quantum circuits

- Quantum algorithm: unitary matrix U (i.e., $U^\dagger U = I = UU^\dagger$)

Quantum circuits

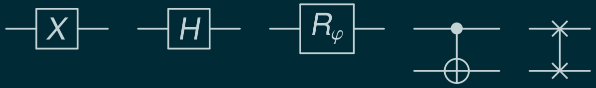
- Quantum algorithm: unitary matrix U (i.e., $U^\dagger U = I = UU^\dagger$)
- (circuit) complexity: number of elementary gates

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \qquad R_\varphi = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i \varphi} \end{pmatrix}$$

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

(Gates extend by $\otimes I$ to the other qubits.)

- quantum circuit notation for X, H, R_φ, $CNOT$, and $SWAP$:



Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.

Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.
- Quantum computers can implement reversible logical operations.

Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.
- Quantum computers can implement reversible logical operations.
- Every classical Boolean circuit can be made reversible by using ancilla (qu)bits!

Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.
- Quantum computers can implement reversible logical operations.
- Every classical Boolean circuit can be made reversible by using ancilla (qu)bits!

Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.
- Quantum computers can implement reversible logical operations.
- Every classical Boolean circuit can be made reversible by using ancilla (qu)bits!

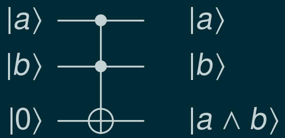
- Classical and gate:

Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.
- Quantum computers can implement reversible logical operations.
- Every classical Boolean circuit can be made reversible by using ancilla (qu)bits!

- Classical and gate:

- Reversible quantum version (a.k.a. Toffoli gate):

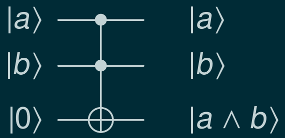


Classical circuits to quantum circuits

- Quantum circuits implement unitary operations which are reversible.
- Quantum computers can implement reversible logical operations.
- Every classical Boolean circuit can be made reversible by using ancilla (qu)bits!

- Classical and gate:

- Reversible quantum version (a.k.a. Toffoli gate):



- For general logical operation $f\colon \{0,1\}^n \to \{0,1\}^m$:

The Quantum Fourier Transform

The (quantum) Fourier transform for $k \in \mathbb{Z}_N$ is defined as

$$F_N : |k\rangle \mapsto \sum_{j=0}^{N-1} e^{\frac{2\pi i}{N} j \cdot k} |j\rangle,$$

where $j \cdot k$ is the usual product of two integers in $\mathbb{Z}_N$.

The Quantum Fourier Transform

The (quantum) Fourier transform for $k \in \mathbb{Z}_N$ is defined as

$$F_N \colon |k\rangle \mapsto \sum_{j=0}^{N-1} e^{\frac{2\pi i}{N} j \cdot k} |j\rangle,$$

where $j \cdot k$ is the usual product of two integers in $\mathbb{Z}_N$.
Let $\omega_N := e^{2\pi i/N}$, in matrix notation we can write:

$$F_N = \frac{1}{\sqrt{N}} \begin{pmatrix} & \vdots & \\ \cdots & \omega_N^{jk} & \cdots \\ & \vdots & \end{pmatrix}.$$

The Quantum Fourier Transform

The (quantum) Fourier transform for $k \in \mathbb{Z}_N$ is defined as

$$F_N \colon |k\rangle \mapsto \sum_{j=0}^{N-1} e^{\frac{2\pi i}{N} j \cdot k} |j\rangle,$$

where $j \cdot k$ is the usual product of two integers in $\mathbb{Z}_N$.
Let $\omega_N := e^{2\pi i/N}$, in matrix notation we can write:

$$F_N = \frac{1}{\sqrt{N}} \begin{pmatrix} & \vdots & \\ \cdots & \omega_N^{jk} & \cdots \\ & \vdots & \end{pmatrix}.$$

Note that for $\mathbb{Z}_2$ we have $H = F_2 = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$.

The Quantum Fourier Transform

F_N is a unitary matrix, since each column has norm 1, and any two distinct columns k and k' are orthogonal:

$$\begin{aligned}
(F_N|k\rangle)^\dagger \cdot (F_N|k'\rangle) &= \left(\frac{1}{\sqrt{N}}\sum_{j=0}^{N-1}(\omega_N^{jk})^*\langle j|\right)\cdot\left(\frac{1}{\sqrt{N}}\sum_{j'=0}^{N-1}(\omega_N^{j'k'})|j'\rangle\right) \\
&= \sum_{j=0}^{N-1}\frac{1}{\sqrt{N}}(\omega_N^{jk})^*\frac{1}{\sqrt{N}}\omega_N^{jk'} \\
&= \frac{1}{N}\sum_{j=0}^{N-1}\omega_N^{j(k'-k)} \\
&= \begin{cases} 1 & \text{if } k = k' \\ 0 & \text{otherwise} \end{cases}
\end{aligned}$$

Since F_N is unitary we have that $F_N^{-1} = F_N^\dagger$. As F_N is also symmetric we further get $F_N^{-1} = F_N^\dagger = F_N^*$, i.e., F_N^{-1} can be computed by simply conjugating each entry of F_N.

Shor's factoring algorithm from period finding

For an $1 < x < N$, $x \nmid N$ consider the sequence

$$1 = x^0 \pmod{N}, \quad x^1 \pmod{N}, \quad x^2 \pmod{N}, \ldots$$

This sequence will cycle after a while: there is a least $0 < r \leq N$ such that $x^r = 1 \pmod{N}$. This r is called the <u>period</u> of the sequence (a.k.a. the <u>order</u> of the element x in the group $\mathbb{Z}_N^*$).

Assuming N is odd and not a prime power (those cases are easy to factor anyway), it can be shown that with probability $\geq 1/2$, the period r is even and $x^{r/2} + 1$ and $x^{r/2} - 1$ are not multiples of N.

Shor's factoring algorithm from period finding

For an $1 < x < N$, $x \nmid N$ consider the sequence

$$1 = x^0 \pmod N, \quad x^1 \pmod N, \quad x^2 \pmod N, \ldots$$

This sequence will cycle after a while: there is a least $0 < r \leq N$ such that $x^r = 1 \pmod N$. This r is called the period of the sequence (a.k.a. the order of the element x in the group $\mathbb{Z}_N^*$).

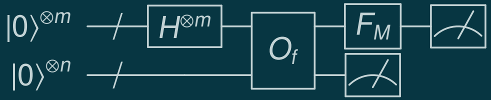
Assuming N is odd and not a prime power (those cases are easy to factor anyway), it can be shown that with probability $\geq 1/2$, the period r is even and $x^{r/2} + 1$ and $x^{r/2} - 1$ are not multiples of N.

In that case we have:

$$\begin{aligned} x^r &\equiv 1 \mod N &\iff \\ (x^{r/2})^2 &\equiv 1 \mod N &\iff \\ (x^{r/2}+1)(x^{r/2}-1) &\equiv 0 \mod N &\iff \\ \underbrace{(x^{r/2}+1)}_{N\nmid}\underbrace{(x^{r/2}-1)}_{N\nmid} &= kN \text{ for some } k. \end{aligned}$$

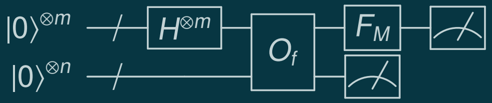
Quantum Period Finding

Suppose f has period r and for all $x = 1, \ldots, r$ the value $f(x)$ is distinct. Let $M := 2^m$.



Quantum Period Finding

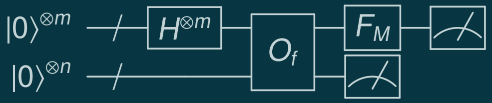
Suppose f has period r and for all $x = 1, \ldots, r$ the value $f(x)$ is distinct. Let $M := 2^m$.



$$|0\rangle^{\otimes m}|0\rangle^{\otimes n} \xrightarrow{H^{\otimes m}} \sum_{j=0}^{M-1} |j\rangle|0\rangle^{\otimes n} \xrightarrow{O_f} \sum_{j=0}^{M-1} |j\rangle\big|f(j)\big\rangle \xrightarrow{\text{measure}} \propto \sum_{k=0}^{\lfloor \frac{M-1-s}{r} \rfloor} |s + k \cdot r\rangle\big|f(s)\big\rangle$$

Quantum Period Finding

Suppose f has period r and for all $x = 1, \ldots, r$ the value $f(x)$ is distinct. Let $M := 2^m$.



$$|0\rangle^{\otimes m}|0\rangle^{\otimes n} \xrightarrow{H^{\otimes m}} \sum_{j=0}^{M-1} |j\rangle|0\rangle^{\otimes n} \xrightarrow{O_f} \sum_{j=0}^{M-1} |j\rangle\big|f(j)\rangle \xrightarrow{\text{measure}} \propto \sum_{k=0}^{\lfloor \frac{M-1-s}{r} \rfloor} |s + k \cdot r\rangle\big|f(s)\rangle$$

For simplicity let us assume that $r \mid M$, then

$$\sum_{k=0}^{\frac{M}{r}-1} |s + k \cdot r\rangle \xrightarrow{F_M} \sum_{k=0}^{\frac{M}{r}-1} \sum_{j=0}^{M-1} e^{\frac{2\pi i}{M} j \cdot (s+k\cdot r)} |j\rangle = \sum_{j=0}^{M-1} e^{\frac{2\pi i}{M} j \cdot s} |j\rangle \underbrace{\sum_{k=0}^{\frac{M}{r}-1} e^{\frac{2\pi i}{M} j \cdot r \cdot k}}_{(e^{\frac{2\pi i}{M} j \cdot r \frac{M}{r}} - 1)/(e^{\frac{2\pi i}{M} j \cdot r} - 1)} = \begin{cases} \frac{M}{r} & \text{if } j = c \cdot \frac{M}{r} \\ 0 & \text{otherwise} \end{cases}$$

The Hidden Subgroup Problem

Given a known group G and a function $f : G \to S$ where S is some finite set.

The Hidden Subgroup Problem

Given a known group G and a function $f : G \to S$ where S is some finite set. Suppose f has the property that there exists a subgroup $H \leq G$ such that f is constant within each coset, and distinct on different cosets: $f(g) = f(g')$ iff $gH = g'H$.

The Hidden Subgroup Problem

Given a known group G and a function $f : G \to S$ where S is some finite set.
Suppose f has the property that there exists a subgroup $H \leq G$ such that f is constant within each coset, and distinct on different cosets: $f(g) = f(g')$ iff $gH = g'H$.
Goal: find H (for example output a set of generators).

- For Abelian groups G, a generalized version of Shor's algorithm works.

The Hidden Subgroup Problem

Given a known group G and a function $f : G \to S$ where S is some finite set.
Suppose f has the property that there exists a subgroup $H \leq G$ such that f is constant within each coset, and distinct on different cosets: $f(g) = f(g')$ iff $gH = g'H$.
Goal: find H (for example output a set of generators).

- For Abelian groups G, a generalized version of Shor's algorithm works.
- This breaks discrete logartihm, elliptic curve based crypto, Diffie-Hellman, etc.

The Hidden Subgroup Problem

Given a known group G and a function $f : G \to S$ where S is some finite set.
Suppose f has the property that there exists a subgroup $H \leq G$ such that f is constant within each coset, and distinct on different cosets: $f(g) = f(g')$ iff $gH = g'H$.
Goal: find H (for example output a set of generators).

- For Abelian groups G, a generalized version of Shor's algorithm works.
- This breaks discrete logartihm, elliptic curve based crypto, Diffie-Hellman, etc.
- For some types of non-Abelian groups we have efficient quantum algorithms.

The Hidden Subgroup Problem

Given a known group G and a function $f : G \to S$ where S is some finite set.
Suppose f has the property that there exists a subgroup $H \leq G$ such that f is constant within each coset, and distinct on different cosets: $f(g) = f(g')$ iff $gH = g'H$.
Goal: find H (for example output a set of generators).

- For Abelian groups G, a generalized version of Shor's algorithm works.
- This breaks discrete logartihm, elliptic curve based crypto, Diffie-Hellman, etc.
- For some types of non-Abelian groups we have efficient quantum algorithms.
- For the dihedral group D_n (containing the symmetries of a regular n-gon), Kuperberg's sieve solves the problem in subexponential time (about $\mathcal{O}\big(2^{\sqrt{n}}\big)$).

Grover's algorithm and amplitude amplification

Suppose we have a probabilistic algorithm that detects "success"

$$U|0\rangle^{\otimes n} = |\psi\rangle = \sqrt{p}\left|\psi_{\text{good}}\right\rangle|1\rangle + \sqrt{1-p}|\psi_{\text{bad}}\rangle|0\rangle.$$

Grover's algorithm and amplitude amplification

Suppose we have a probabilistic algorithm that detects "success"

$$U|0\rangle^{\otimes n} = |\psi\rangle = \sqrt{p}\Big|\psi_{\mathrm{good}}\Big\rangle|1\rangle + \sqrt{1-p}|\psi_{\mathrm{bad}}\rangle|0\rangle.$$

The Grover operator G_U is defined as follows

$$G_U = (2|\psi\rangle\langle\psi| - I_n) \cdot (2I_{n-1} \otimes |0\rangle\langle 0| - I_n).$$

Grover's algorithm and amplitude amplification

Suppose we have a probabilistic algorithm that detects "success"

$$U|0\rangle^{\otimes n} = |\psi\rangle = \sqrt{p}\Big|\psi_{\mathrm{good}}\Big\rangle|1\rangle + \sqrt{1-p}|\psi_{\mathrm{bad}}\rangle|0\rangle.$$

The Grover operator G_U is defined as follows

$$G_U = (2|\psi\rangle\langle\psi| - I_n) \cdot (2I_{n-1} \otimes |0\rangle\langle 0| - I_n).$$

G_U acts as a 2θ-angle rotation in a two-dimensional invariant subspace, where

$$\theta = \arcsin\Big(\Big\|(I_{n-1} \otimes |1\rangle\langle 1|)|\psi\rangle\Big\|\Big) = \arcsin\big(\sqrt{p}\big).$$

Grover's algorithm and amplitude amplification

Suppose we have a probabilistic algorithm that detects "success"

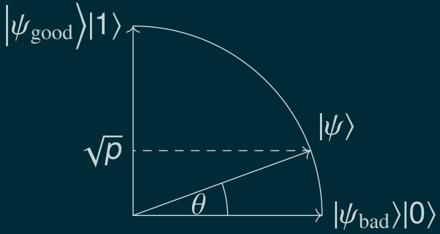
$$U|0\rangle^{\otimes n} = |\psi\rangle = \sqrt{p}\left|\psi_{\mathrm{good}}\right\rangle|1\rangle + \sqrt{1-p}|\psi_{\mathrm{bad}}\rangle|0\rangle.$$

The Grover operator G_U is defined as follows

$$G_U = (2|\psi\rangle\langle\psi| - I_n) \cdot (2I_{n-1} \otimes |0\rangle\langle 0| - I_n).$$

G_U acts as a 2θ-angle rotation in a two-dimensional invariant subspace, where

$$\theta = \arcsin\left(\left\|(I_{n-1} \otimes |1\rangle\langle 1|)|\psi\rangle\right\|\right) = \arcsin\left(\sqrt{p}\right).$$



Grover's algorithm and amplitude amplification

Suppose we have a probabilistic algorithm that detects "success"

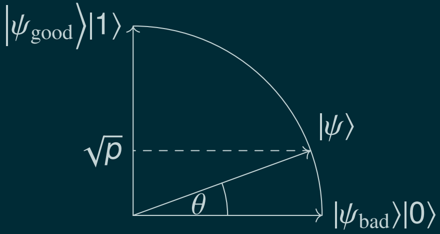
$$U|0\rangle^{\otimes n} = |\psi\rangle = \sqrt{p}\left|\psi_{\mathrm{good}}\right\rangle|1\rangle + \sqrt{1-p}|\psi_{\mathrm{bad}}\rangle|0\rangle.$$

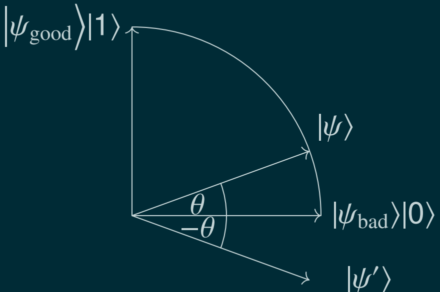
The Grover operator G_U is defined as follows

$$G_U = (2|\psi\rangle\langle\psi| - I_n)\cdot(2I_{n-1}\otimes|0\rangle\langle 0| - I_n).$$

G_U acts as a 2θ-angle rotation in a two-dimensional invariant subspace, where

$$\theta = \arcsin\Big(\big\|(I_{n-1}\otimes|1\rangle\langle 1|)|\psi\rangle\big\|\Big) = \arcsin\big(\sqrt{p}\big).$$





Grover's algorithm and amplitude amplification

Suppose we have a probabilistic algorithm that detects "success"

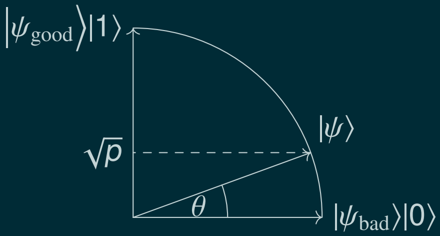
$$U|0\rangle^{\otimes n} = |\psi\rangle = \sqrt{p}\left|\psi_{\mathrm{good}}\right\rangle|1\rangle + \sqrt{1-p}|\psi_{\mathrm{bad}}\rangle|0\rangle.$$

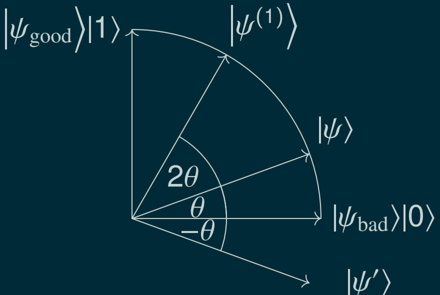
The Grover operator G_U is defined as follows

$$G_U = (2|\psi\rangle\langle\psi| - I_n)\cdot(2I_{n-1}\otimes|0\rangle\langle 0| - I_n).$$

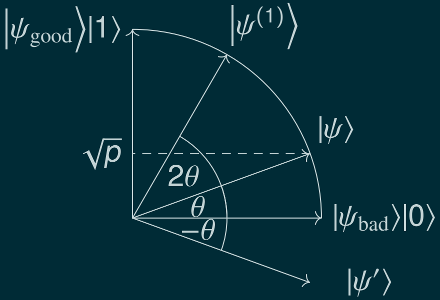
G_U acts as a 2θ-angle rotation in a two-dimensional invariant subspace, where

$$\theta = \arcsin\left(\left\|(I_{n-1}\otimes|1\rangle\langle 1|)|\psi\rangle\right\|\right) = \arcsin\left(\sqrt{p}\right).$$



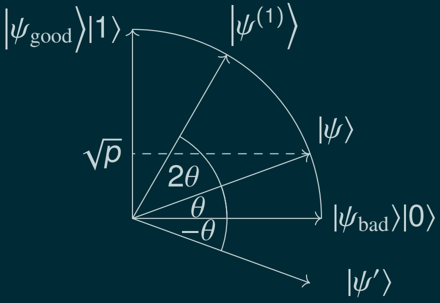


Grover's algorithm and amplitude amplification



The success probability after k iteration is $\sin^2((2k+1)\theta)$!

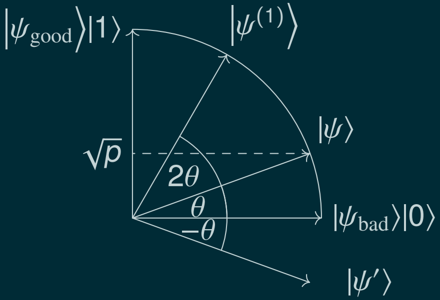
Grover's algorithm and amplitude amplification



The success probability after k iteration is $\sin^2((2k+1)\theta)$!

- For small p we have $\theta \approx \sqrt{p} \gg p$.

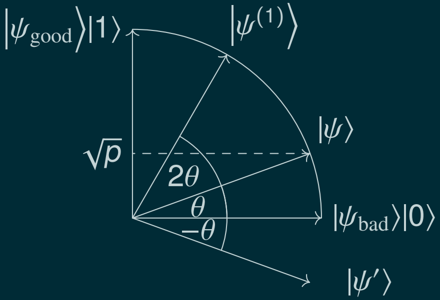
Grover's algorithm and amplitude amplification



The success probability after k iteration is $\sin^2((2k+1)\theta)$!

- For small p we have $\theta \approx \sqrt{p} \gg p$.
- It is possible to over-rotate.

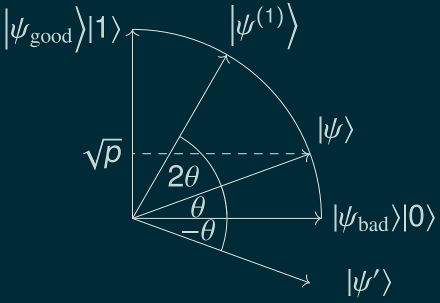
Grover's algorithm and amplitude amplification



The success probability after k iteration is $\sin^2((2k+1)\theta)$!

- For small p we have $\theta \approx \sqrt{p} \gg p$.
- It is possible to over-rotate.
- Grover's original problem – find a (unique) marked element m among N choices.

Grover's algorithm and amplitude amplification



The success probability after k iteration is $\sin^2((2k+1)\theta)$!

- For small p we have $\theta \approx \sqrt{p} \gg p$.
- It is possible to over-rotate.
- Grover's original problem – find a (unique) marked element m among N choices. Prepare a uniform superposition and check:

$$|0\rangle \xrightarrow{\text{H}} \frac{1}{\sqrt{N}} \sum_{j=0}^{N} |j\rangle|0\rangle \xrightarrow{\text{check}} \frac{1}{\sqrt{N}} \sum_{j=0}^{N} |j\rangle\left|\delta_{mj}\right\rangle \Rightarrow \left|\psi_{\text{good}}\right\rangle = |m\rangle \quad (p = \frac{1}{N})$$

Rewinding & post-quantum security

Classically secure protocol = (classically) hard problem + security reduction

Efficient A wins “security game” $\Rightarrow$ We get efficient A' solving hard problem

Rewinding & post-quantum security

Classically secure protocol = (classically) hard problem + security reduction

Efficient A wins "security game" $\Rightarrow$ We get efficient A' solving hard problem

Post-quantum security = quantum-hard problem + (classical) reduction?

Rewinding & post-quantum security

Classically secure protocol = (classically) hard problem + security reduction

Efficient A wins "security game" $\Rightarrow$ We get efficient A' solving hard problem

Post-quantum security = quantum-hard problem + (classical) reduction?

[BCMVV18] protocol: Prover $\leftrightarrows$ Verifier: accept/reject

Rewinding & post-quantum security

Classically secure protocol = (classically) hard problem + security reduction

Efficient A wins “security game” $\Rightarrow$ We get efficient A' solving hard problem

Post-quantum security = quantum-hard problem + (classical) reduction?

[BCMVV18] protocol: Prover $\leftrightarrows$ Verifier: accept/reject

- Efficient classical P cannot make V accept assuming LWE hard

Rewinding & post-quantum security

Classically secure protocol = (classically) hard problem + security reduction

Efficient A wins "security game" $\Rightarrow$ We get efficient A' solving hard problem

Post-quantum security = quantum-hard problem + (classical) reduction?

[BCMVV18] protocol: Prover $\leftrightarrows$ Verifier: accept/reject

- Efficient classical P cannot make V accept assuming LWE hard
- Efficient quantum P can convince V to accept

Rewinding & post-quantum security

Classically secure protocol = (classically) hard problem + security reduction

Efficient A wins "security game" $\Rightarrow$ We get efficient A' solving hard problem

Post-quantum security = quantum-hard problem + (classical) reduction?

[BCMVV18] protocol: Prover $\leftrightarrows$ Verifier: accept/reject

- Efficient classical P cannot make V accept assuming LWE hard
- Efficient quantum P can convince V to accept

For more details see the "Quantum Rewinding Tutorial" of Alex Lombardi (MIT) and Fermi Ma (UC Berkeley) recorded at the Simons Institute (available on YouTube).

[BCMVV18]: Zvika Brakerski, Paul Christiano, Urmila Mahadev, Umesh V. Vazirani, and Thomas Vidick. A cryptographic test of quantumness and certifiable randomness from a single quantum device. J. ACM (August 2021). Earlier version at FOCS 2018.

Rewinding a'la Mariott-Watrous

You hold a useful quantum state

- If you measure in basis $A \Rightarrow$ you can solve problem A
- If you measure in basis $B \Rightarrow$ you can solve problem B

Rewinding a’la Mariott-Watrous

You hold a useful quantum state

- If you measure in basis $A \Rightarrow$ you can solve problem A
- If you measure in basis $B \Rightarrow$ you can solve problem B

Mariott-Watrous trick

- Suppose you can measure projector Π and any state in the image of Π is good for you.

Rewinding a'la Mariott-Watrous

You hold a useful quantum state

- If you measure in basis $A \Rightarrow$ you can solve problem A
- If you measure in basis $B \Rightarrow$ you can solve problem B

Mariott-Watrous trick

- Suppose you can measure projector Π and any state in the image of Π is good for you.
- Suppose you can solve problem A via a binary measurement $(\Pi_A, I - \Pi_A)$.

Trick: alternately repeat the two measurements $(\Pi_A, I - \Pi_A)$ and $(\Pi, I - \Pi)$ until you get lucky and get back a state in the image of Π.

Rewinding a'la Mariott-Watrous

You hold a useful quantum state

- If you measure in basis $A \Rightarrow$ you can solve problem A
- If you measure in basis $B \Rightarrow$ you can solve problem B

Mariott-Watrous trick

- Suppose you can measure projector Π and any state in the image of Π is good for you.
- Suppose you can solve problem A via a binary measurement $(\Pi_A, I - \Pi_A)$.

Trick: alternately repeat the two measurements $(\Pi_A, I - \Pi_A)$ and $(\Pi, I - \Pi)$ until you get lucky and get back a state in the image of Π.

In expectation 4 measurements suffice to get back such a state!

Further reading

- Parts of this presentation come from Ronald de Wolf's Quantum Coputing Lecture Notes – arXiv: `1907.09415`.
- See also the "Quantum Rewinding Tutorial" Part 1,2, & 3 of Alex Lombardi (MIT) and Fermi Ma (UC Berkeley) recorded at the Simons Institute on June 15th.