ACCQ 206 - Simon's algorithm, Quantum Fourier transform and Shor's algorithm

Alex Bredariol Grilo
Alex.Bredariol-Grilo@lip6.fr

Quantum operations

Evolution of quantum states is described by unitary operators

- $UU^\dagger = U^\dagger U = I$
  - For every quantum state $|\psi\rangle$, $U|\psi\rangle$ is also a quantum state
  - Reversible: no information loss
- Equivalent models of quantum computation:
  - Quantum Turing Machines
  - Quantum circuits
  - Adiabatic quantum computation
  - Measurement-based quantum computation
  - ...

Universal gateset

Definition

ε-approximation An n-qubit unitary U ε-approximates an n-qubit unitary U' if

$$\max_{|\psi\rangle \in \mathbb{C}^{2^n}} ||U|\psi\rangle - U'|\psi\rangle|| \leq \varepsilon.$$

Definition

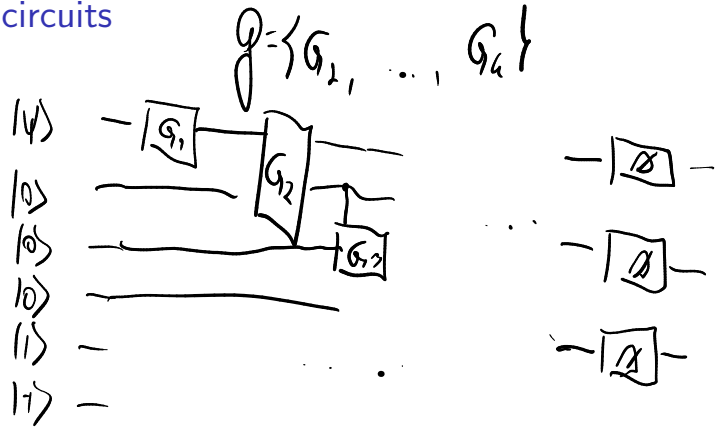
Universal gateset A gateset $\mathcal{G}$ is universal if for every unitary U, there exists a unitary U' composed by gates in $\mathcal{G}$ such that U' ε-approximates U.

Lemma

The following gatesets are universal:

- $\{1\text{-qubit gates}, CNOT\}$
- $\{CNOT, H, T\}$
- $\{H, CCNOT\}$ *(for unitaries with real entries)*

Quantum circuits



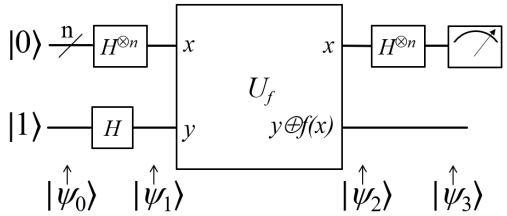
Deutsch-Josza algorithm (1992)

Problem

Given oracle access to $f : \{0,1\}^n \to \{0,1\}$ with the promise that:

- f is constant
- f is balanced

Find out which is the case.



Quantum parallelism: $|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle\,|f(x)\rangle$

$(H^{\otimes n} \otimes I)\ \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|0\rangle = \sum_x \sum_y (-1)^{xy} |y\rangle$

$= |00\ldots0\rangle$

- if f is constant then the outcome is $|0\ldots0\rangle$
- if f is balanced then the outcome is $\neq |0\ldots0\rangle$

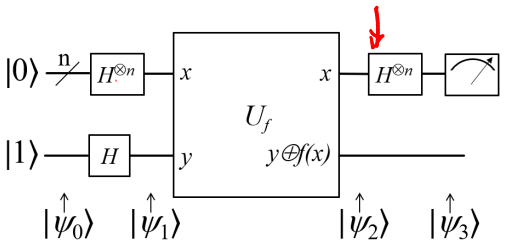
Deutsch-Josza algorithm (1992)

Problem

Given oracle access to $f : \{0,1\}^n \to \{0,1\}$ with the promise that:

- f is constant
- f is balanced

Find out which is the case.



Quantum parallelism: $|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle\,|f(x)\rangle$

$\frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} |x\rangle|-\rangle$

$H^{\otimes n}|0\ldots0\rangle = \frac{1}{\sqrt{2^n}} \sum_x (-1)^{0 \cdot y} |y\rangle = \frac{1}{\sqrt{2^n}} \sum_y |y\rangle$

$(H^{\otimes n})^2 = I$

Simon's algorithm (1994)

Problem

Given oracle access to a function $f : \{0,1\}^n \to \{0,1\}^n$ such that:

$$\exists s \neq 0^n \text{ such that } f(x) = f(y) \text{ iff } y \in \{x, s \oplus x\}.$$

Find s.

Simon's algorithm

Problem

Given oracle access to a function $f : \{0,1\}^n \to \{0,1\}^n$ such that:

$$\exists s \neq 0^n \text{ such that } f(x) = f(y) \text{ iff } y \in \{x, s \oplus y\}.$$

Find s.

Randomized algorithms: pick $x_1, \ldots, x_k$ until $\exists i, j$ s.t. $f(x_i) = f(x_j)$

$O(\sqrt{n})$

lower bound on query complexity to find s is $\Omega(\sqrt{n})$

Simon's algorithm

Lemma

With a single quantum query, we can compute a random $d \in \{0,1\}^n$ such that

$$d \cdot s = 0.$$

$\sum d_i s_i \mod(2) = 0$

Theorem

There is a quantum algorithm that retrieves s with high probability with $O(n)$ queries.

Proof

linearly independent $d_1, \ldots, d_n$ s.t

$\begin{cases} d_1 \cdot s = 0 \\ d_2 \cdot s = 0 \\ \vdots \\ d_n \cdot s = 0 \end{cases}$ $\Rightarrow$ gives you s

d_i s.t $d_1 \ldots d_i$ are linearly indep.

w.p. $\geq \frac{i+1}{n}$

Simon's algorithm

Lemma

With a single quantum query, we can compute a random $d \in \{0,1\}^n$ such that

$$d \cdot s = 0.$$

Theorem

There is a quantum algorithm that retrieves s with high probability with $O(n)$ queries.

Proof

When we sample d_i, $d_1 \dots d_i$ is LI wp. $1 - \frac{2^i}{2^n}$

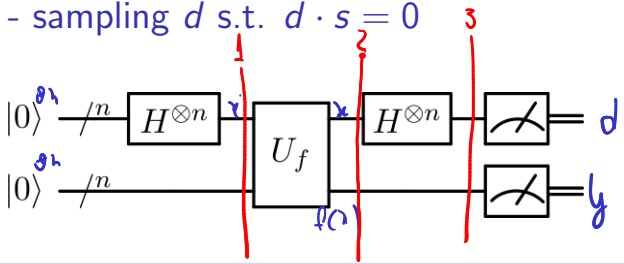
d_1 = Take

$d_2 \neq 2^n - 1$

$d_1, \dots, d_{i-1}$

2^{i-1} strings in $\mathrm{span}(d_1, \dots, d_{i-1})$

Simon's algorithm - sampling d s.t. $d \cdot s = 0$



Analysis

1) $H^{\otimes n}|0\ldots0\rangle = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle$

2) $U_f\left(\frac{1}{\sqrt{2^n}} \sum_x |x\rangle|0\rangle\right) \to \frac{1}{\sqrt{2^n}} \sum_x |x\rangle|f(x)\rangle =$

$\frac{1}{\sqrt{2^n}} \sum_{y \in Im(f)} \left(|f_1^{-1}(y)\rangle + |f_2^{-1}(y)\rangle\right)|y\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in Im(f)} \left(|f_1^{-1}(y)\rangle + |f_1^{-1}(y) \oplus s\rangle\right)|y\rangle$

Simon's algorithm

Analysis (cont.)

Simon's algorithm

Analysis (cont.)

Simon's algorithm

Analysis (cont.)

Simon's algorithm

Analysis (cont.)

If we measure the second register, the outcome is a random $y \in \mathrm{Im}(f)$, and the post-meas. state is

$$\frac{1}{\sqrt{2}}\left(|f_i^{-1}(y)\rangle + |f_i^{-1}(y) \oplus s\rangle\right)|y\rangle$$

3) $H^{\otimes n}$

Simon's algorithm

Analysis (cont.)

$$H^{\otimes n}\left(\frac{1}{\sqrt{2}}|z\rangle + \frac{1}{\sqrt{2}}|z \oplus s\rangle\right) = \frac{1}{\sqrt{2}} H^{\otimes n}|z\rangle + \frac{1}{\sqrt{2}} H^{\otimes n}|z \oplus s\rangle =$$

$$= \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2^n}} \left(\sum_d (-1)^{d \cdot z} |d\rangle + \sum_d (-1)^{d \cdot (z \oplus s)} |d\rangle \right) \qquad (-1)^{d \cdot z \oplus d \cdot s} = (-1)^{d \cdot z} \cdot (-1)^{d \cdot s}$$

$$= \frac{1}{\sqrt{2}} \frac{1}{\sqrt{2^n}} \left(\sum_d \underbrace{\left((-1)^{d \cdot z} + (-1)^{d \cdot z} \cdot (-1)^{d \cdot s} \right)}_{(-1)^{d \cdot z}\left(1 + (-1)^{d \cdot s}\right)} |d\rangle \right)$$

(*)

if $d \cdot s = 1$: $(*) = (-1)^{d \cdot z}\left(1 + (-1)\right) = 0$

if $d \cdot s = 0$: $(*) = (-1)^{d \cdot z}(1 + 1) = 2(-1)^{d \cdot z}$

Simon's algorithm

Analysis (cont.)

$$= \frac{1}{\sqrt{2}} \cdot \frac{1}{\sqrt{2^n}} \sum_{d : d \cdot s = 0} 2 (-1)^{d \cdot z} |d\rangle = \frac{1}{\sqrt{2^{n-1}}} \sum_{d : d \cdot s = 0} (-1)^{d \cdot z} |d\rangle$$

state after step 3 φ

4) After measuring the measurement $d^* : d^* \cdot s = 0$ is the outcome of $w.p.$ $\left(\frac{(-1)^{d \cdot z}}{\sqrt{2^{n-1}}} \right)^2 = \frac{1}{2^{n-1}}$

Simon's algorithm

Analysis (cont.)

Simon's algorithm - recap

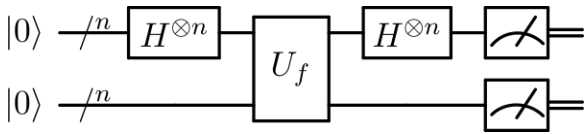
Problem

Given oracle access to a function $f : \{0,1\}^n \to \{0,1\}^n$ such that:

$$\exists s \neq 0^n \text{ such that } f(x) = f(y) \text{ iff } y \in \{x, s \oplus x\}.$$

Find s.

The following circuit samples random d such that $d \cdot s = 0$



Sampling it $O(n)$ times, with high probability we have n linearly independent d_i's and we can solve the following linear system of equations to compute s

$$\forall 1 \leq i \leq n, d_i \cdot s = 0.$$

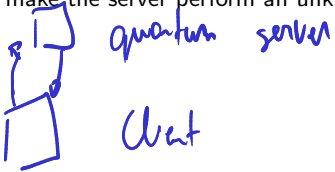
Simon's algorithm - applications

Kaplan, Leurent, Leverrier and Naya-Plasencia. Breaking Symmetric Cryptosystems using Quantum Period Finding. CRYPTO 2016

Authentication schemes that are secure classically are broken in the quantum query model using Simon's algorithm.

Mahadev. Classical Verification of Quantum Computations. FOCS 2018

Breakthrough result on delegation of quantum computation by classical clients, it uses a variant of Simon's algorithm to make the server perform an unknown measurement.



Factoring

Problem input size is $\log N = n$

For a composite number N, find a non-trivial factor of N.

Classical algorithms:

$2^{(\log N)^{\alpha}}$ → if heuristic, $\alpha = 1/3$

→ if worst-case, $\alpha = 1/2$ ⇒ $2^{\sqrt{n}}$

Belief: factoring is hard

↳ most crypto constructions are based on hardness of factoring

Shor's algorithm (1994) - overview

↳ poly $(\log N)$ - time quantum alg. for factoring N

① "classical part"
↳ reduction from factoring to period finding

② "quantum part"
↳ period finding w/ ① Fourier transform

Periodic functions

Definition

A function $f : \mathbb{F} \to \mathbb{F}$ is periodic if $\exists\, r$ such that $f(a) = f(b)$ iff $b = a + kr \pmod{p}$.

Example

For every a, N, $f_a(x) = a^x \pmod{N}$ is periodic.

Shor's algorithm - from period finding to factoring

Algorithm 1: Factoring from period finding

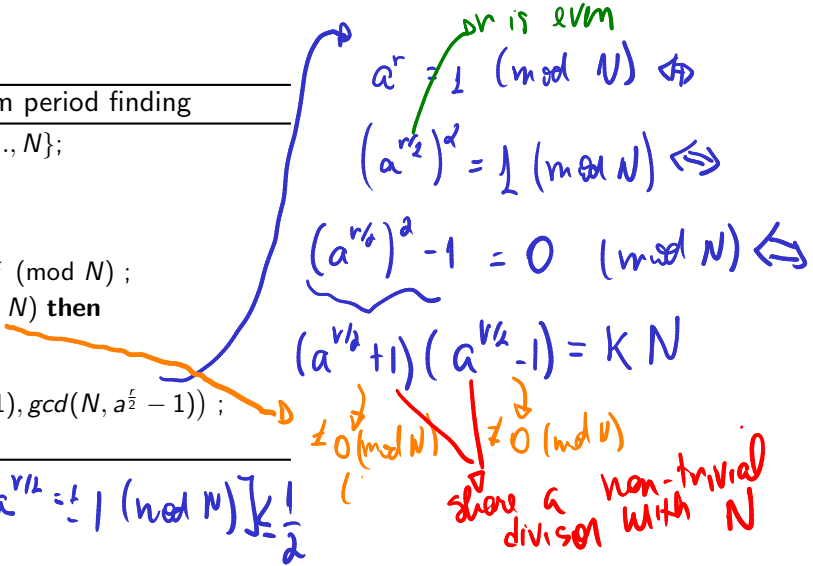
Pick an unif. random $a \in \{2, ..., N-1\}$;
if $gcd(a, N) > 1$ **then**
  return $gcd(a, N)$; → already have a non-trivial factor of N (we got lucky :))
end
Find the period r of $f_a(x) = a^x \pmod N$;
if r *is odd* or $a^{r/2} = \pm 1 \pmod N$ **then**
  return $\perp$;
else
  return $\max\left(gcd(N, a^{\frac{r}{2}} + 1), gcd(N, a^{\frac{r}{2}} - 1)\right)$;
end

Shor's algorithm - from period finding to factoring

Algorithm 1: Factoring from period finding

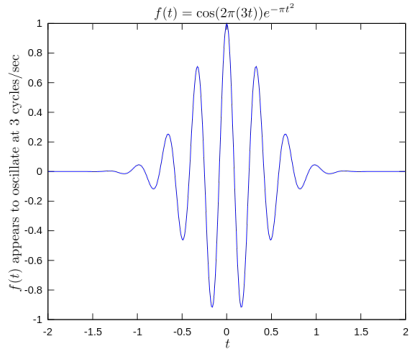
Pick an unif. random $a \in \{2, ..., N\}$;
if $gcd(x, N) > 1$ **then**
  return $gcd(a, N)$;
end
Find the period r of $f_a(x) = a^x \pmod N$;
if r *is odd* or $a^{r/2} = \pm 1 \pmod N$ **then**
  return $\perp$;
else
  return $\max\left(gcd(N, a^{\frac{r}{2}} + 1), gcd(N, a^{\frac{r}{2}} - 1)\right)$;
end

$P_a[\, r \text{ is odd} \text{ or } a^{r/2} = -1 \pmod N] \leq \frac{1}{2}$

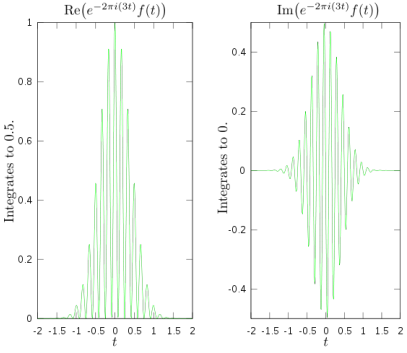


Fourier Transform

Original signal



Fourier transform



$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx$$

Discrete Fourier Transform

$x_1, \dots, x_N \in \mathbb{C} \xrightarrow{\text{DFT}} y_1, \dots, y_N \in \mathbb{C}$

$$y_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N} x_k e^{-2\pi i jk/N}$$

$$F_N = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & \dots & 1 & \dots & 1 \\ 1 & \omega & \dots & \omega^{j-1} & \dots & \omega^{N-1} \\ & & & \dots & & \\ 1 & \omega^{i-1} & \dots & \omega^{(i-1)(j-1)} & \dots & \omega^{i(N-1)} \\ & & & \dots & & \\ 1 & \omega^{N-1} & \dots & \omega^{(N-1)(j-1)} & \dots & \omega^{(N-1)^2} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} y_1 \\ \vdots \\ y_N \end{pmatrix}$$

Unitary (exercise!)

N-th root of unity $\omega = e^{2\pi i/N}$

naive DFT computation $O(N^2)$.

FFT $\mathrm{DFT}(x_1, \dots, x_N)$ on time $O(N \log N)$

Quantum Fourier Transform

$|\psi\rangle = \begin{pmatrix} \alpha_0 \\ \vdots \\ \alpha_{N-1} \end{pmatrix} \Big\} 2^n \xrightarrow{QFT} |\phi\rangle = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_{N-1} \end{pmatrix} \Big\} 2^n \qquad \sum_j \beta_j |j\rangle$

n

$N = 2^n$

$\sum \alpha_j |j\rangle$

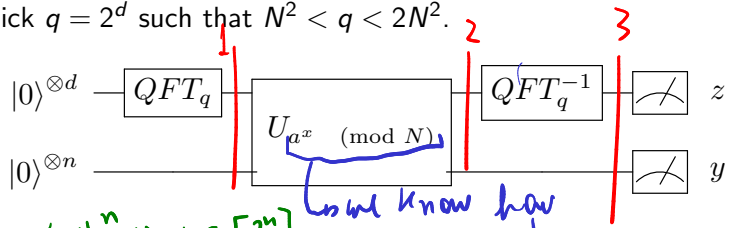
$$\beta_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \alpha_k \, e^{-2\pi i \, jk/N}$$

we can compute QFT using $O((\log N)^2)$ "simple" gates

$O(n^2)$

Shor's algorithm - quantum algorithm for period finding

Pick $q = 2^d$ such that $N^2 < q < 2N^2$.



+ classical post-processing

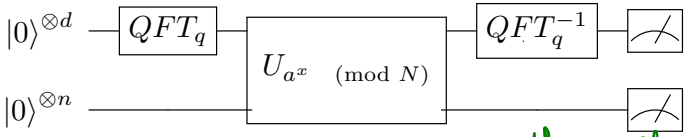
$x \in \{0,1\}^n \Leftrightarrow y \in [2^n]$

1) $QFT\,|0\rangle = \frac{1}{\sqrt{q}} \sum_{j=0}^{q-1} e^{2\pi i j \cdot 0/q} |j\rangle = \frac{1}{\sqrt{q}} \sum_{j=0}^{q-1} |j\rangle = H^{\otimes d} |0\rangle$

2) $U_{f_a}\left(\frac{1}{\sqrt{q}} \sum_{j=0}^{q-1} |j\rangle |0\rangle^{\otimes n}\right) \to \frac{1}{\sqrt{q}} \sum_{j=0}^{q-1} |j\rangle |a^j \pmod N\rangle$

Shor's algorithm - quantum algorithm for period finding

Pick $q = 2^d$ such that $N^2 < q < 2N^2$.



+ classical post-processing

$\frac{1}{\sqrt{q}} \sum_{j=0}^{q-1} |j\rangle |a^j \pmod N\rangle$ measure the 2nd register

if outcome is y

$\sum_{j:\, a^j = y \pmod N} |j\rangle|y\rangle = \sum_{k=0}^{m-1} |x_0 + ks\rangle|y\rangle$

period of a^x

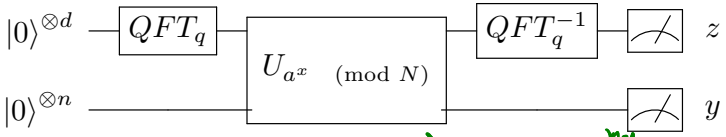
$x_0 \in \mathbb{Z}$

$\exists\ s < r \quad f_a(s) = y$

$m = \#$ pre-images of y

Shor's algorithm - quantum algorithm for period finding

Pick $q = 2^d$ such that $N^2 < q < 2N^2$.



+ classical post-processing

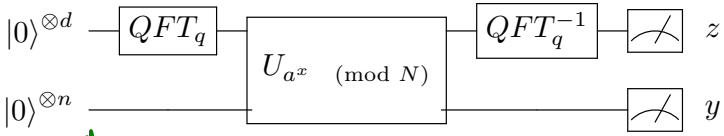
3) $QFT^{-1}\left(\frac{1}{\sqrt{m}}\sum_k |kr+s\rangle\right) = \frac{1}{\sqrt{m}}\sum_{k=0}^{m-1}\frac{1}{\sqrt{q}}\sum_{\ell=0}^{q-1} e^{2\pi i(kr+s)\ell/q}|\ell\rangle$

$= \frac{1}{\sqrt{mq}}\sum_{\ell=0}^{q-1} e^{2\pi i s\ell/q}\left(\sum_{k=0}^{m-1}\left(e^{2\pi i r\ell/q}\right)^k\right)|\ell\rangle$

for different ℓ's

Shor's algorithm - quantum algorithm for period finding

Pick $q = 2^d$ such that $N^2 < q < 2N^2$.



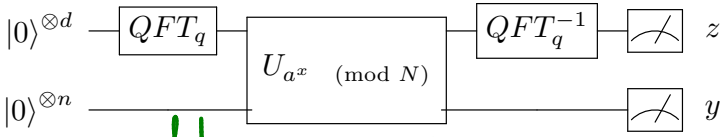
+ classical post-processing

$(**)$ $\sum_{k=0}^{m-1} \left(e^{\pi i r \ell / q}\right)^k$ for diff. values of ℓ

- $e^{\pi i r \ell / q} = 1 \Rightarrow (**) = \sum_{k=0}^{m-1} (1)^k = m$
- $e^{\pi i r \ell / q} \neq 1 \Rightarrow (**) = \frac{1 - e^{\pi i r m \ell / q}}{1 - e^{\pi i r \ell / q}}$

Shor's algorithm - quantum algorithm for period finding

Pick $q = 2^d$ such that $N^2 < q < 2N^2$.



+ classical post-processing

Assume that r divides q:

$e^{2\pi i r\ell/q} = 1 \iff \ell = c \cdot q/r \Rightarrow \exists$ m multiples of q/r in $\{0, \dots, q-1\}$

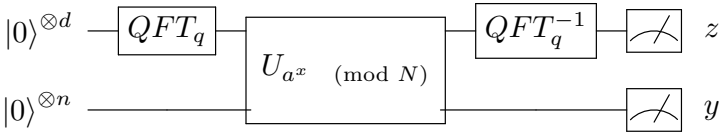
$r\ell/q$ is an integer

amplitude on original state is $\frac{m}{\sqrt{mq}}$

probs of measuring such ℓ's is $\frac{m^2}{mq} = \frac{m}{q} = \frac{1}{r}$

Shor's algorithm - quantum algorithm for period finding

Pick $q = 2^d$ such that $N^2 < q < 2N^2$.



+ classical post-processing

Bottom line : in the "simple" core $\boxed{2} = c \cdot \frac{q}{r}$

$\Downarrow$

$\frac{1}{\log\log N}$ $\boxed{\text{poly}(\log N)}$

quantum measurement of parameter of algo.

$\frac{2}{q} = \frac{c}{r}$ — period that we want to find

c and r are co-prime w.p. $\frac{1}{\log\log N}$

Factoring in quantum polynomial time

Unstructured search

Computational problem

Let us assume that a function $f : \{0,1\}^n \to \{0,1\}$ has the following property:

$$\exists \text{ a unique } x^* \text{ such that } f(x^*) = 1.$$

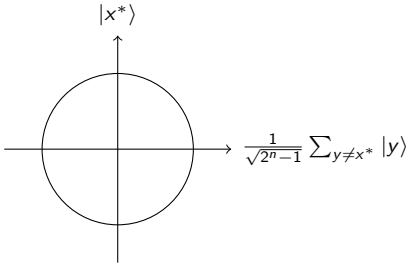
Find x^*.

Classical algorithms: 2^n inputs of f

Grover search - amplitude amplification

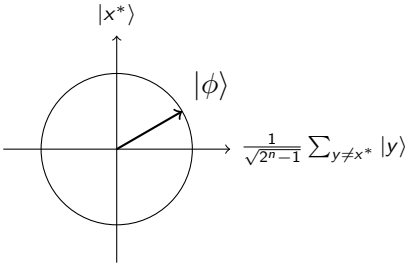
Grover search - amplitude amplification

Phase inversion



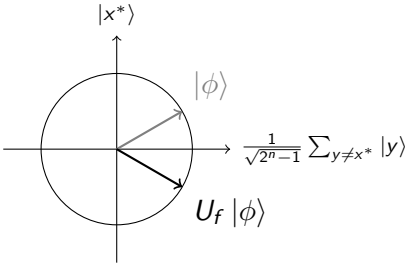
Grover search - amplitude amplification

Phase inversion



Grover search - amplitude amplification

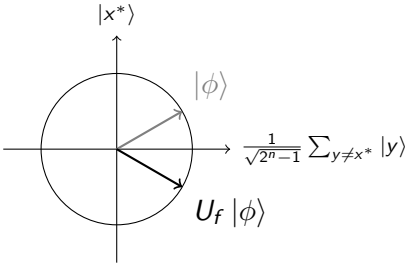
Phase inversion



$$U_f |i\rangle = (-1)^{f(i)} |i\rangle$$

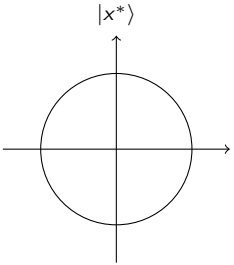
Grover search - amplitude amplification

Phase inversion



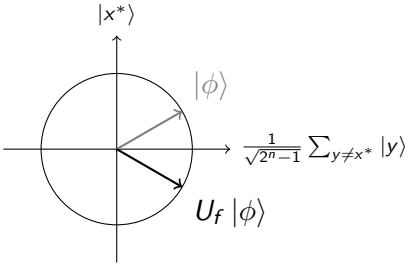
$U_f |i\rangle = (-1)^{f(i)} |i\rangle$

Inversion about the mean



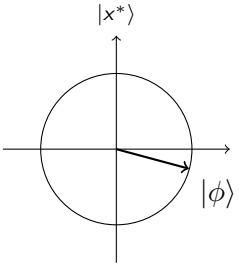
Grover search - amplitude amplification

Phase inversion



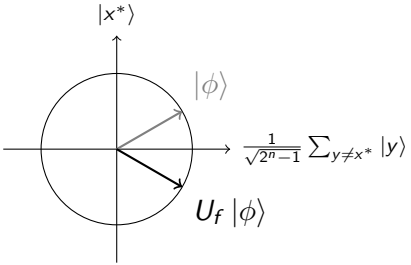
$U_f |i\rangle = (-1)^{f(i)} |i\rangle$

Inversion about the mean



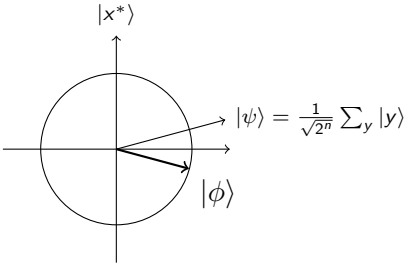
Grover search - amplitude amplification

Phase inversion



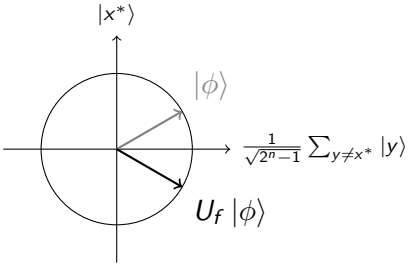
$U_f |i\rangle = (-1)^{f(i)} |i\rangle$

Inversion about the mean



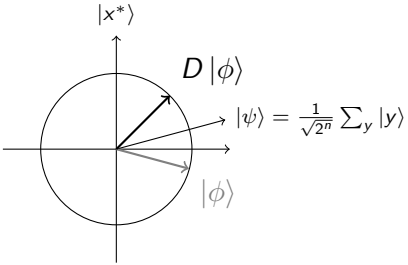
Grover search - amplitude amplification

Phase inversion



$U_f |i\rangle = (-1)^{f(i)} |i\rangle$

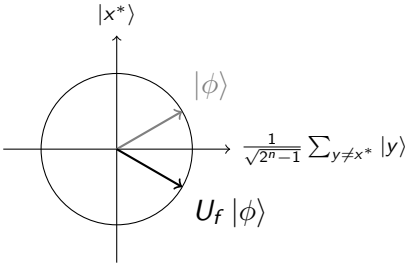
Inversion about the mean



$D |i\rangle = H^{\otimes n}(I - 2 |0\rangle\langle 0|)H^{\otimes n} |i\rangle$

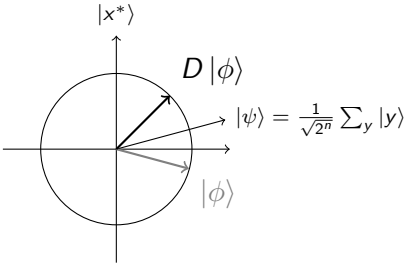
Grover search - amplitude amplification

Phase inversion



$U_f|i\rangle = (-1)^{f(i)}|i\rangle$

Inversion about the mean



$D|i\rangle = H^{\otimes n}(I - 2|0\rangle\langle 0|)H^{\otimes n}|i\rangle$

Grover's algorithm

Start from $\frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle$ and repeat these two procedures $\sqrt{2^n}$ times.

Grover search and amplitude amplification

- The algorithm can be generalized for different settings: more than one marked element, amplification of success probability, quantum rewinding in quantum cryptographic protocols
- Quadratic speedup over classical algorithm
- Not so drastic, but it solves a generic problem
- It can be reframed into different frameworks: quantum walks, block encoding and Hamiltonian simulations, ...

Recent (somewhat) advances in quantum algorithms

- Quantum linear algebra
  - Linear system of equations
  - Semi-definite programming
  - Applications to Quantum Machine Learning: faster recommendation systems, ...
- Quantum learning theory
  - Learning theory: ML from the perspective of complexity theory
  - Exponential separation between some quantum and classical models
  - Connections to cryptography and circuit lower bounds
- Quantum chemistry
  - Using quantum computers to simulate quantum systems
- NISQ: Near-term intermediate scale quantum computers
  - What can we do with current Google, IBM, D-Wave devices?

• quantum random walks 'Quantum Fourier sampling
• Block encodings