Quantum algorithms for searching, resampling, and hidden shift problems

Maris Ozols
University of Waterloo
IQC

November 7, 2011

Outline

1. Quantum algorithms for searching
2. Quantum rejection sampling
3. Boolean hidden shift problem

Previous work

- [arXiv:1004.2721] **Adiabatic condition and the quantum hitting time of Markov chains**

  Hari Krovi, Maris Ozols, Jérémie Roland
  - Phys. Rev. A, vol. 82(2), pp. 022333 (2010)
- [arXiv:1002.2419] **Finding is as easy as detecting for quantum walks**

  Hari Krovi, Frédéric Magniez, Maris Ozols, Jérémie Roland
  - Lecture Notes in Computer Science, vol. 6198, pp. 540–551 (2010)
  - ICALP 2010
  - QIP 2011 (featured talk)
- [arXiv:1009.1195] **Entanglement can increase asymptotic rates of zero-error classical communication over classical channels**

  Debbie Leung, Laura Mancinska, William Matthews, Maris Ozols, Aidan Roy
  - Communications in Mathematical Physics (submitted)
  - QIP 2011 (featured talk)
- [arXiv:1103.2774] **Quantum rejection sampling**

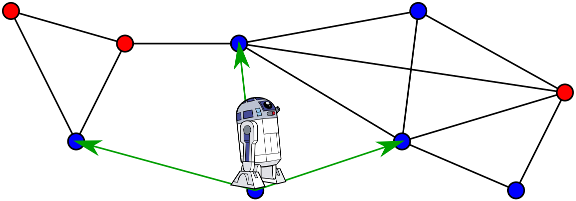
  Maris Ozols, Martin Roetteler, Jérémie Roland
  - QIP 2012 (invited talk)
  - ITCS 2012

Quantum algorithms for searching

Spatial search on a graph

Setup

- Graph with vertex set X
- Marked vertices: unknown $M \subseteq X$
- Vertex register: current position
- Edges: legal moves



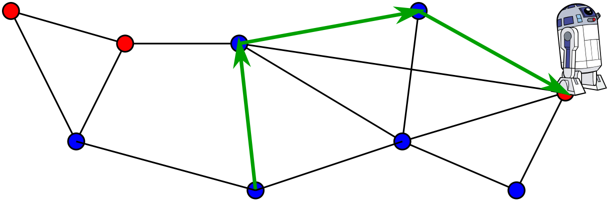
Spatial search on a graph

Setup

- Graph with vertex set X
- Marked vertices: unknown $M \subseteq X$
- Vertex register: current position
- Edges: legal moves

The problem

- Move the robot to a marked vertex $x \in M$
- Complexity: # moves

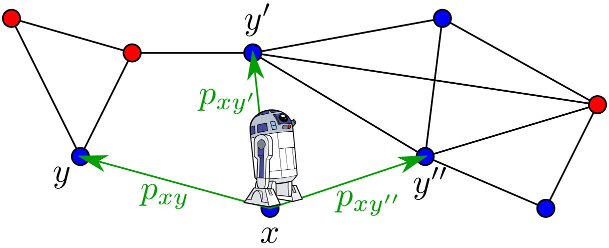


Search via random walk

Markov chain on the graph

Stochastic matrix $P = (p_{xy})$

- $p_{xy} \neq 0$ only if (x, y) is an edge
- stationary distribution: $\pi = \pi P$



Search via random walk

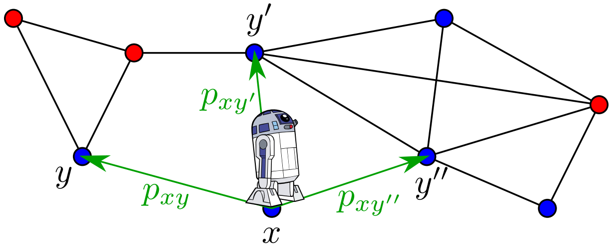
Markov chain on the graph

Stochastic matrix $P = (p_{xy})$

- $p_{xy} \neq 0$ only if (x, y) is an edge
- stationary distribution: $\pi = \pi P$

Algorithm

- Start from random $x \sim \pi$
- Apply P until x is marked



Search via random walk

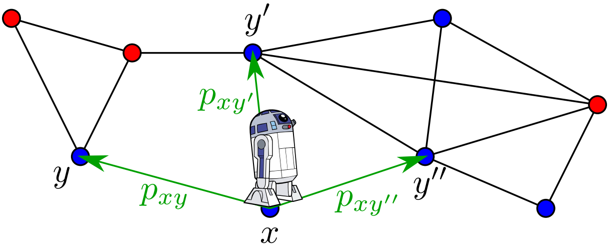
Markov chain on the graph

Stochastic matrix $P = (p_{xy})$

- $p_{xy} \neq 0$ only if (x, y) is an edge
- stationary distribution: $\pi = \pi P$

Algorithm

- Start from random $x \sim \pi$
- Apply P until x is marked



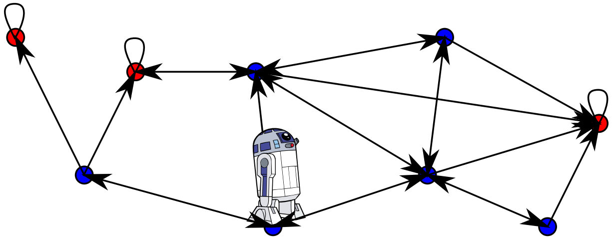
Hitting time

$\mathrm{HT}(P, M) =$ expected $\#$ steps of P to reach any $x \in M$

Classical intuition

Absorbing walk

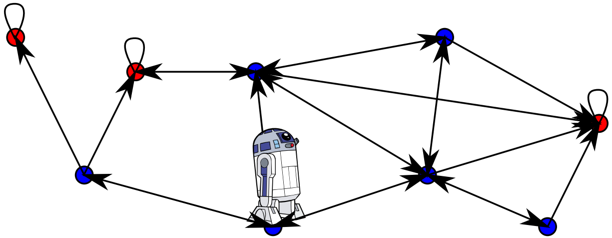
- Turn all outgoing transitions from marked vertices into self-loops: $P = \begin{pmatrix} P_{UU} & P_{UM} \\ P_{MU} & P_{MM} \end{pmatrix} \Rightarrow P' = \begin{pmatrix} P_{UU} & P_{UM} \\ 0 & I \end{pmatrix}$
- Stationary distribution: $\pi_M =$ "π restricted to M"



Classical intuition

Absorbing walk

- Turn all outgoing transitions from marked vertices into self-loops: $P = \begin{pmatrix} P_{UU} & P_{UM} \\ P_{MU} & P_{MM} \end{pmatrix} \Rightarrow P' = \begin{pmatrix} P_{UU} & P_{UM} \\ 0 & I \end{pmatrix}$
- Stationary distribution: $\pi_M =$ "π restricted to M"



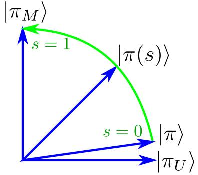
Interpolation

- $P(s) = (1-s)P + sP'$
- Stationary distribution: $\pi(s) \sim \big((1-s)\pi_U \; \pi_M\big)$

The algorithm

Adiabatic version

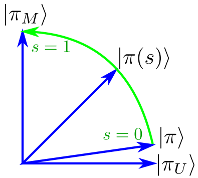
- Define a Hamiltonian $H(s)$ corresponding to $P(s)$
- Interpolate s from 0 to 1



The algorithm

Adiabatic version

- Define a Hamiltonian $H(s)$ corresponding to $P(s)$
- Interpolate s from 0 to 1



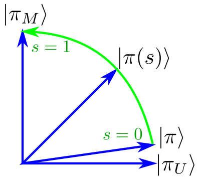
Circuit version

- Use Szegedy's method to define a unitary $W(P(s))$
- $W(P(s))$ has a unique 1-eigenvector $|\pi(s)\rangle$
- Use phase estimation to measure in the eigenbasis of $W(P(s))$

The algorithm

Adiabatic version

- Define a Hamiltonian $H(s)$ corresponding to $P(s)$
- Interpolate s from 0 to 1

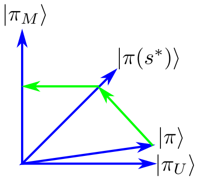


Circuit version

- Use Szegedy's method to define a unitary $W(P(s))$
- $W(P(s))$ has a unique 1-eigenvector $|\pi(s)\rangle$
- Use phase estimation to measure in the eigenbasis of $W(P(s))$

Algorithm

1. Prepare $|\pi\rangle$
2. Project onto $|\pi(s^*)\rangle = \frac{|\pi_U\rangle + |\pi_M\rangle}{\sqrt{2}}$
3. Measure current vertex



The main result

Theorem

Let P be a reversible, ergodic Markov chain on a set X and $M \subseteq X$ be a set of marked elements. A quantum algorithm can find an element in M within $\sqrt{\mathrm{HT}(P, M)}$ steps

Quantum rejection sampling

Classical rejection sampling

Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s

Classical rejection sampling

Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known

Classical rejection sampling

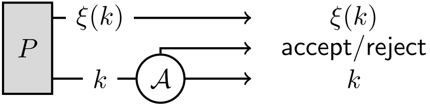
Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known, but samples are pairs $(k, \xi(k))$ where $\xi(k)$ is not accessible

Classical rejection sampling

Classical resampling problem

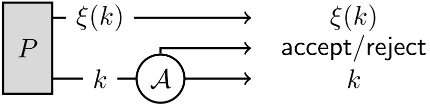
- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known, but samples are pairs $(k, \xi(k))$ where $\xi(k)$ is not accessible



Classical rejection sampling

Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known, but samples are pairs $(k, \xi(k))$ where $\xi(k)$ is not accessible



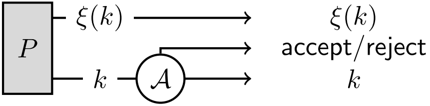
Classical algorithm

- Accept k with probability $\gamma \dfrac{s_k}{p_k}$

Classical rejection sampling

Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known, but samples are pairs $(k, \xi(k))$ where $\xi(k)$ is not accessible



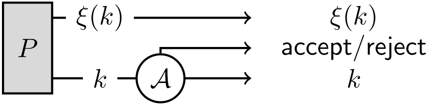
Classical algorithm

- Accept k with probability $\gamma \dfrac{s_k}{p_k}$ where $\forall k : \gamma \dfrac{s_k}{p_k} \leq 1$

Classical rejection sampling

Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known, but samples are pairs $(k, \xi(k))$ where $\xi(k)$ is not accessible



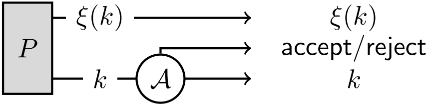
Classical algorithm

- Accept k with probability $\gamma \dfrac{s_k}{p_k}$ where $\gamma = \min_k \dfrac{p_k}{s_k}$

Classical rejection sampling

Classical resampling problem

- **Given:** Ability to sample from distribution p
- **Task:** Sample from distribution s
- **Note:** Distributions p and s are known, but samples are pairs $(k, \xi(k))$ where $\xi(k)$ is not accessible



Classical algorithm

- Accept k with probability $\gamma \dfrac{s_k}{p_k}$ where $\gamma = \min_k \dfrac{p_k}{s_k}$
- Complexity: $\Theta(1/\gamma)$ where $1/\gamma = \max_k \dfrac{s_k}{p_k}$

Quantum rejection sampling

Quantum resampling problem

- **Given:** Oracle $O : |0\rangle \mapsto \sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle$

Quantum rejection sampling

Quantum resampling problem

- **Given:** Oracle $O : |0\rangle \mapsto \sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle$
- **Task:** Perform transformation

$$\sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle \mapsto \sum_{k=1}^{n} \sigma_k |\xi_k\rangle |k\rangle$$

Quantum rejection sampling

Quantum resampling problem

- **Given:** Oracle $O : |0\rangle \mapsto \sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle$
- **Task:** Perform transformation

$$\sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle \mapsto \sum_{k=1}^{n} \sigma_k |\xi_k\rangle |k\rangle$$

- **Note:** Amplitudes π_k and σ_k are known, but states $|\xi_k\rangle$ are not known

Quantum rejection sampling

Quantum resampling problem

- **Given:** Oracle $O : |0\rangle \mapsto \sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle$
- **Task:** Perform transformation

$$\sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle \mapsto \sum_{k=1}^{n} \sigma_k |\xi_k\rangle |k\rangle$$

- **Note:** Amplitudes π_k and σ_k are known, but states $|\xi_k\rangle$ are not known

Theorem

The quantum query complexity of the $\boldsymbol{\pi} \to \boldsymbol{\sigma}$ quantum resampling problem is $\Theta(1/\gamma)$ where $1/\gamma = \max_k \left|\frac{\sigma_k}{\pi_k}\right|$

Quantum rejection sampling

Quantum resampling problem

- **Given:** Oracle $O : |0\rangle \mapsto \sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle$
- **Task:** Perform transformation

$$\sum_{k=1}^{n} \pi_k |\xi_k\rangle |k\rangle \mapsto \sum_{k=1}^{n} \sigma_k |\xi_k\rangle |k\rangle$$

- **Note:** Amplitudes π_k and σ_k are known, but states $|\xi_k\rangle$ are not known

Theorem

The quantum query complexity of the $\boldsymbol{\pi} \to \boldsymbol{\sigma}$ quantum resampling problem is $\Theta(1/\gamma)$ where $1/\gamma = \max_k \left| \frac{\sigma_k}{\pi_k} \right|$

Proof idea: Algorithm is based on amplitude amplification, but the lower bound is based on a hybrid argument

Applications

Implicit use

- synthesis of quantum states [Grover, 2000]
- linear systems of equations [Harrow, Hassidim and Lloyd 2009]
- fast amplification of QMA [Nagaj, Wocjan, Zhang, 2009]

New applications

- speed up quantum Metropolis sampling algorithm by [Temme, Osborne, Vollbrecht, Poulin, Verstraete, 2011]
- new quantum algorithm for the hidden shift problem of any Boolean function

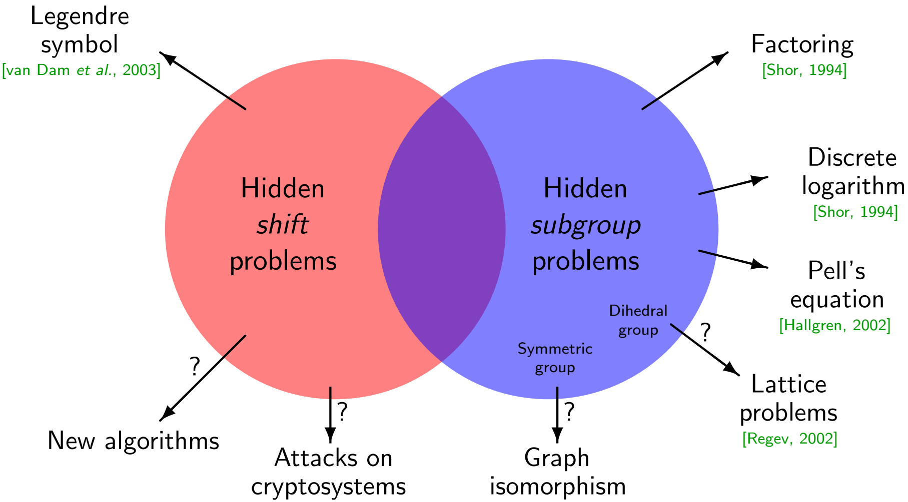
New applications by others

- preparing PEPS states [Schwarz, Temme, Verstraete, 2011]

Boolean hidden shift problem

Motivation

Hidden *shift* and *subgroup* problems



Boolean hidden shift problem (BHSP)

Problem

- **Given:** Complete knowledge of $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ and access to a black-box oracle for $f_s(x) := f(x+s)$

$$x \Rightarrow \boxed{} \Rightarrow f_s(x)$$

- **Determine:** The hidden shift s

Boolean hidden shift problem (BHSP)

Problem

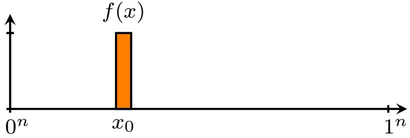
- **Given:** Complete knowledge of $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ and access to a black-box oracle for $f_s(x) := f(x+s)$

$$x \Rightarrow \blacksquare \Rightarrow f_s(x)$$

- **Determine:** The hidden shift s

Delta functions are hard

- $f(x) := \delta_{x,x_0}$



Boolean hidden shift problem (BHSP)

Problem

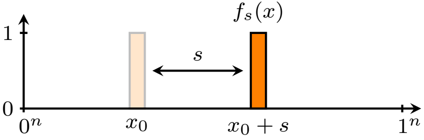
- **Given:** Complete knowledge of $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ and access to a black-box oracle for $f_s(x) := f(x+s)$

$$x \Rightarrow \blacksquare \Rightarrow f_s(x)$$

- **Determine:** The hidden shift s

Delta functions are hard

- $f(x) := \delta_{x,x_0}$



Boolean hidden shift problem (BHSP)

Problem

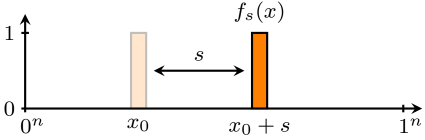
- **Given:** Complete knowledge of $f : \mathbb{Z}_2^n \to \mathbb{Z}_2$ and access to a black-box oracle for $f_s(x) := f(x + s)$

$$x \Rightarrow \blacksquare \Rightarrow f_s(x)$$

- **Determine:** The hidden shift s

Delta functions are hard

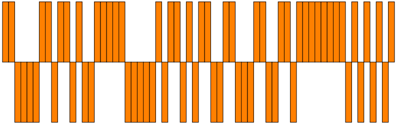
- $f(x) := \delta_{x,x_0}$
- Equivalent to Grover's search: $\Theta(\sqrt{2^n})$



Fourier transform of Boolean functions

The ± 1-function (normalized)

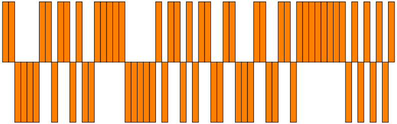
- $F(x) := \frac{1}{\sqrt{2^n}}(-1)^{f(x)}$



Fourier transform of Boolean functions

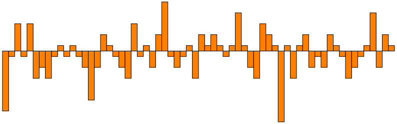
The ± 1-function (normalized)

- $F(x) := \frac{1}{\sqrt{2^n}}(-1)^{f(x)}$



Fourier transform

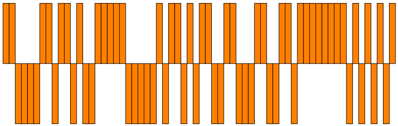
- $\hat{F}(w) := \langle w|H^{\otimes n}|F\rangle$



Fourier transform of Boolean functions

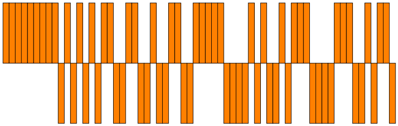
The ± 1-function (normalized)

- $F(x) := \frac{1}{\sqrt{2^n}}(-1)^{f(x)}$



Fourier transform

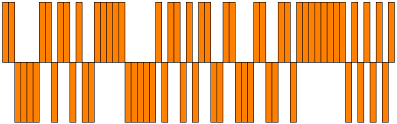
- $\hat{F}(w) := \langle w|H^{\otimes n}|F\rangle$



Fourier transform of Boolean functions

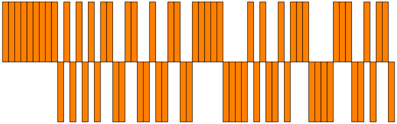
The ± 1-function (normalized)

- $F(x) := \frac{1}{\sqrt{2^n}}(-1)^{f(x)}$



Fourier transform

- $\hat{F}(w) := \langle w | H^{\otimes n} | F \rangle$



Function f is **bent** if $\forall w : |\hat{F}(w)| = \frac{1}{\sqrt{2^n}}$

Bent functions are easy

Preparing the "phase state"

- Phase oracle $O_{f_s} : |x\rangle \mapsto (-1)^{f_s(x)}|x\rangle$

Bent functions are easy

Preparing the "phase state"

- Phase oracle $O_{f_s} : |x\rangle \mapsto (-1)^{f_s(x)}|x\rangle$

$$|0\rangle^{\otimes n} \;-\!\boxed{H^{\otimes n}}\!-\!\boxed{O_{f_s}}\!-\!\boxed{H^{\otimes n}}\!-\; |\Phi(s)\rangle$$

- $|\Phi(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle$

Bent functions are easy

Preparing the "phase state"

- Phase oracle $O_{f_s} : |x\rangle \mapsto (-1)^{f_s(x)}|x\rangle$

$$|0\rangle^{\otimes n} \; — \boxed{H^{\otimes n}} — \boxed{O_{f_s}} — \boxed{H^{\otimes n}} — \; |\Phi(s)\rangle$$

- $|\Phi(s)\rangle := \sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} \hat{F}(w)|w\rangle$

Algorithm [Rötteler'10]

- If f is bent then $\forall w : |\hat{F}(w)| = \frac{1}{\sqrt{2^n}}$ and thus
$$H^{\otimes n} \operatorname{diag}\left(\frac{|\hat{F}(w)|}{\hat{F}(w)}\right)|\Phi(s)\rangle = |s\rangle$$

Bent functions are easy

Preparing the "phase state"

- Phase oracle $O_{f_s} : |x\rangle \mapsto (-1)^{f_s(x)}|x\rangle$

$$|0\rangle^{\otimes n} \;\text{—}\; \boxed{H^{\otimes n}} \;\text{—}\; \boxed{O_{f_s}} \;\text{—}\; \boxed{H^{\otimes n}} \;\text{—}\; |\Phi(s)\rangle$$

- $|\Phi(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle$

Algorithm [Rötteler'10]

- If f is bent then $\forall w : |\hat{F}(w)| = \frac{1}{\sqrt{2^n}}$ and thus
$$H^{\otimes n}\operatorname{diag}\Big(\frac{|\hat{F}(w)|}{\hat{F}(w)}\Big)|\Phi(s)\rangle = |s\rangle$$
- Complexity: $\Theta(1)$

Other Boolean functions?

Known

- delta functions are hard
- bent functions are easy

Problem

What is the quantum query complexity of the hidden shift problem for an arbitrary Boolean function?

Other Boolean functions?

Known

- delta functions are hard
- bent functions are easy

Problem

What is the quantum query complexity of the hidden shift problem for an arbitrary Boolean function?

Three approaches

1. Grover-like [Grover'00] / quantum rejection sampling [ORR'11]
2. Pretty good measurement
3. Simon-like [Rötteler'10, GRR'11]

Algorithm 1: Grover-like / quantum rejection sampling

Quantum resampling

$$\sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} \hat{F}(w) |w\rangle \mapsto \sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} \frac{1}{\sqrt{2^n}} |w\rangle$$

Algorithm 1: Grover-like / quantum rejection sampling

Quantum resampling

$$\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle \mapsto \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\frac{1}{\sqrt{2^n}}|w\rangle$$

Complexity: $O(1/\gamma)$ where $1/\gamma = \max_w \dfrac{\sigma_w}{\pi_w} = \dfrac{1}{\sqrt{2^n}}\cdot\dfrac{1}{\hat{F}_{\min}}$

Algorithm 1: Grover-like / quantum rejection sampling

Quantum resampling

$$\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle \mapsto \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\frac{1}{\sqrt{2^n}}|w\rangle$$

Complexity: $O(1/\gamma)$ where $1/\gamma = \max_w \dfrac{\sigma_w}{\pi_w} = \dfrac{1}{\sqrt{2^n}}\cdot\dfrac{1}{\hat{F}_{\min}}$

Performance

- Delta functions: $O(\sqrt{2^n})$
- Bent functions: $O(1)$

Algorithm 1: Grover-like / quantum rejection sampling

Quantum resampling

$$\sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} \hat{F}(w) |w\rangle \mapsto \sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} \frac{1}{\sqrt{2^n}} |w\rangle$$

Complexity: $O(1/\gamma)$ where $1/\gamma = \max_w \dfrac{\sigma_w}{\pi_w} = \dfrac{1}{\sqrt{2^n}} \cdot \dfrac{1}{\hat{F}_{\min}}$

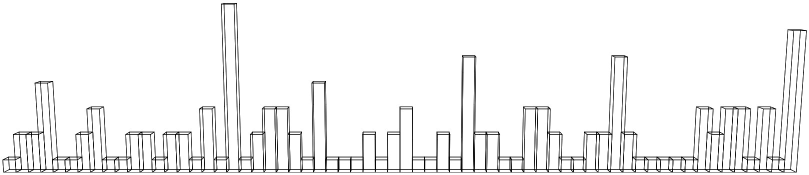
Performance

- Delta functions: $O(\sqrt{2^n})$
- Bent functions: $O(1)$

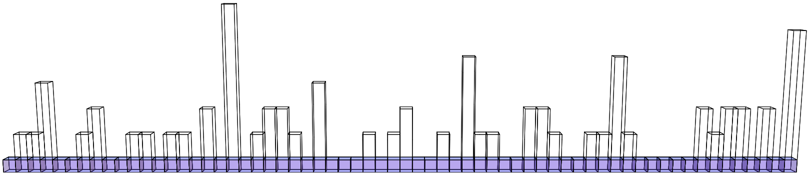
Issues

- What if $\hat{F}_{\min} = 0$?
- Undetectable anti-shifts: $f(x+s) = f(x) + 1$

Algorithm 1: Approximate version

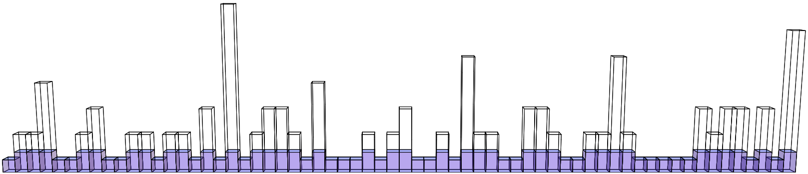


Algorithm 1: Approximate version



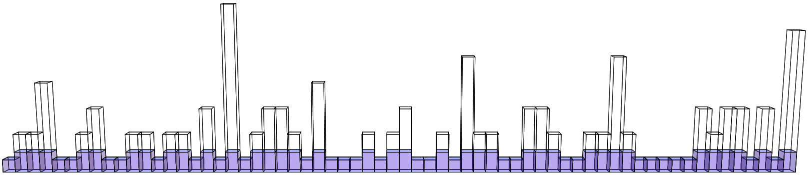
Algorithm 1: Approximate version

- Aim for *approximately* flat state



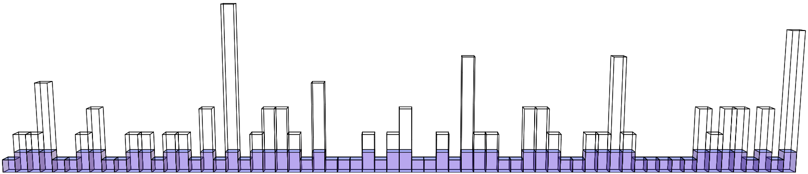
Algorithm 1: Approximate version

- Aim for *approximately* flat state
- Fix success probability p



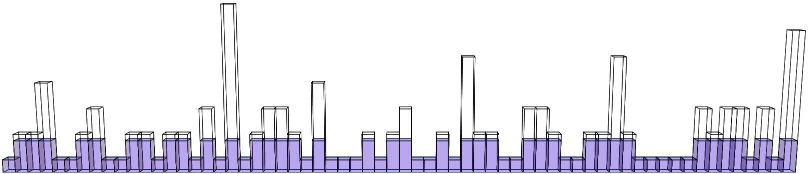
Algorithm 1: Approximate version

- Aim for *approximately* flat state
- Fix success probability p
- Optimal target amplitudes are given by the "water filling" vector $\boldsymbol{\varepsilon}_p$ such that $\boldsymbol{\mu}^\mathsf{T} \cdot \frac{\boldsymbol{\varepsilon}_p}{\|\boldsymbol{\varepsilon}_p\|_2} \geq \sqrt{p}$ where $\mu_w = \frac{1}{\sqrt{2^n}}$



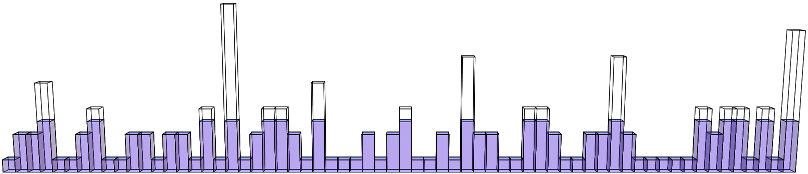
Algorithm 1: Approximate version

- Aim for *approximately* flat state
- Fix success probability p
- Optimal target amplitudes are given by the "water filling" vector $\boldsymbol{\varepsilon}_p$ such that $\boldsymbol{\mu}^{\mathsf{T}} \cdot \frac{\boldsymbol{\varepsilon}_p}{\|\boldsymbol{\varepsilon}_p\|_2} \geq \sqrt{p}$ where $\mu_w = \frac{1}{\sqrt{2^n}}$



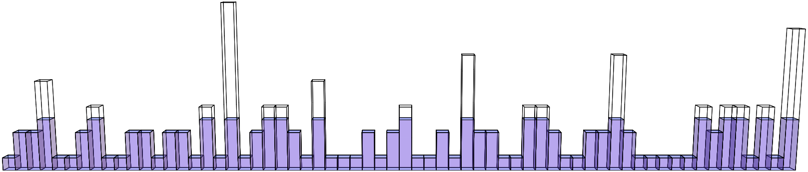
Algorithm 1: Approximate version

- Aim for *approximately* flat state
- Fix success probability p
- Optimal target amplitudes are given by the "water filling" vector $\boldsymbol{\varepsilon}_p$ such that $\boldsymbol{\mu}^\mathsf{T} \cdot \frac{\boldsymbol{\varepsilon}_p}{\|\boldsymbol{\varepsilon}_p\|_2} \geq \sqrt{p}$ where $\mu_w = \frac{1}{\sqrt{2^n}}$

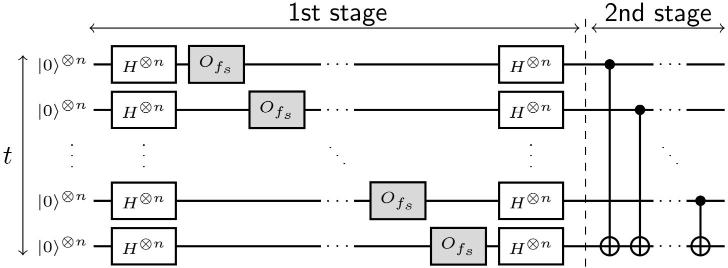


Algorithm 1: Approximate version

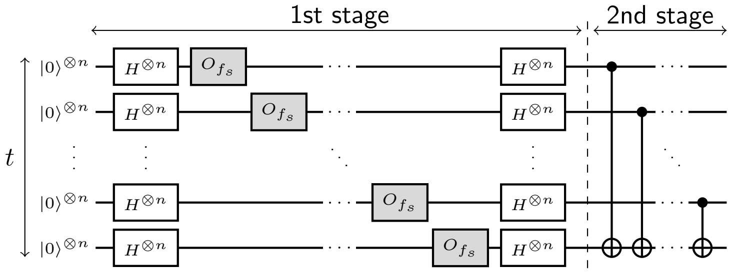
- Aim for *approximately* flat state
- Fix success probability p
- Optimal target amplitudes are given by the "water filling" vector $\boldsymbol{\varepsilon}_p$ such that $\boldsymbol{\mu}^{\mathsf{T}} \cdot \frac{\boldsymbol{\varepsilon}_p}{\|\boldsymbol{\varepsilon}_p\|_2} \geq \sqrt{p}$ where $\mu_w = \frac{1}{\sqrt{2^n}}$
- Queries: $O(1/\|\boldsymbol{\varepsilon}_p\|_2)$



Algorithm 2: Pretty good measurement

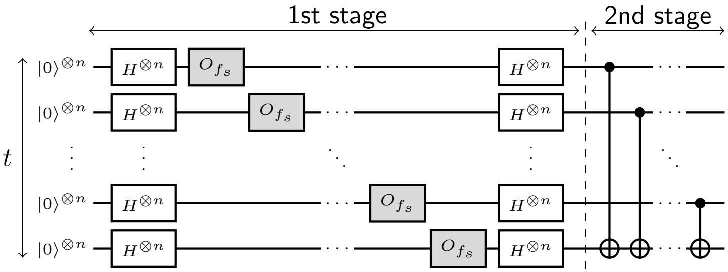


Algorithm 2: Pretty good measurement



After stage 1: $|\Phi(s)\rangle^{\otimes t} = \left(\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle\right)^{\otimes t}$

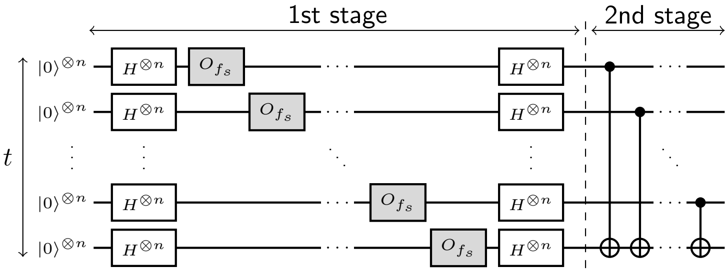
Algorithm 2: Pretty good measurement



After stage 1: $|\Phi(s)\rangle^{\otimes t} = \left(\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle\right)^{\otimes t}$

After stage 2: $|\Phi^t(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}|\mathcal{F}_w^t\rangle|w\rangle$

Algorithm 2: Pretty good measurement

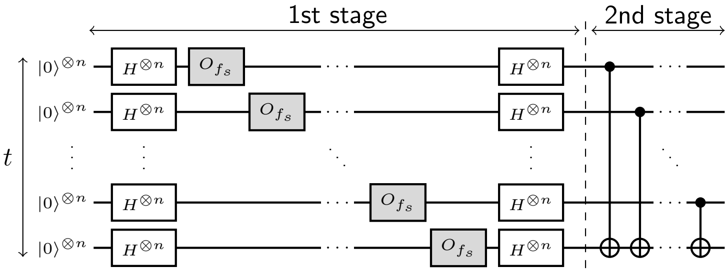


After stage 1: $|\Phi(s)\rangle^{\otimes t} = \left(\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle\right)^{\otimes t}$

After stage 2: $|\Phi^t(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}|\mathcal{F}_w^t\rangle|w\rangle$

PGM: $|E_s^t\rangle := \frac{1}{\sqrt{2^n}}\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\frac{|\mathcal{F}_w^t\rangle}{\||\mathcal{F}_w^t\rangle\|_2}|w\rangle$

Algorithm 2: Pretty good measurement



After stage 1: $|\Phi(s)\rangle^{\otimes t} = \left(\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\hat{F}(w)|w\rangle\right)^{\otimes t}$

After stage 2: $|\Phi^t(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}|\mathcal{F}_w^t\rangle|w\rangle$

PGM: $|E_s^t\rangle := \frac{1}{\sqrt{2^n}}\sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}\frac{|\mathcal{F}_w^t\rangle}{\||\mathcal{F}_w^t\rangle\|_2}|w\rangle$

Success probability:

$$\left|\langle E_s^t|\Phi^t(s)\rangle\right|^2 = \frac{1}{2^n}\left(\sum_{w\in\mathbb{Z}_2^n}\sqrt{\frac{1}{\sqrt{2^n}}\,\widehat{(F*F)^t}(w)}\right)^2$$

Algorithm 2: Pros / cons

Performance

- Bent functions: $O(1)$
- Random functions: $O(1)$
- No issues with undetectable anti-shifts

Issues

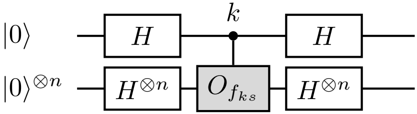
- Delta functions: $O(2^n)$, no speedup

Note

- For some $t \leq n$ all amplitudes will be non-zero!

Algorithm 3: Simon-like

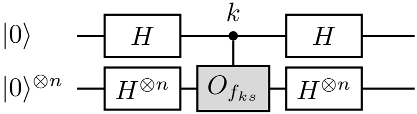
- Oracle $O_{f_{ks}} : |k\rangle|w\rangle \mapsto (-1)^{f(x+ks)}|k\rangle|w\rangle$



$$|\Psi(s)\rangle := \sum_{w\in\mathbb{Z}_2^n} \hat{F}(w)|s\cdot w\rangle|w\rangle$$

Algorithm 3: Simon-like

- Oracle $O_{f_{ks}} : |k\rangle|w\rangle \mapsto (-1)^{f(x+ks)}|k\rangle|w\rangle$

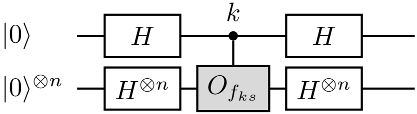


$$|\Psi(s)\rangle := \sum_{w \in \mathbb{Z}_2^n} \hat{F}(w)|s \cdot w\rangle|w\rangle$$

- Complexity: $O(n/\sqrt{I_f})$

Algorithm 3: Simon-like

- Oracle $O_{f_{ks}} : |k\rangle|w\rangle \mapsto (-1)^{f(x+ks)}|k\rangle|w\rangle$



$$|\Psi(s)\rangle := \sum_{w \in \mathbb{Z}_2^n} \hat{F}(w)|s \cdot w\rangle|w\rangle$$

- Complexity: $O(n/\sqrt{I_f})$
- Where $I_f(w)$ is the *influence* of $w \in \mathbb{Z}_2^n$ on f:

$$I_f(w) := \Pr_x\left[f(x) \neq f(x+w)\right]$$

and $I_f := \min_w I_f(w)$

Summary

Comparison

	delta	bent	random	$\hat{F}(w) = 0$ issues
Grover-like	$O(\sqrt{2^n})$	$O(1)$	$O(1)$	yes
PGM	$O(2^n)$	$O(1)$	$O(1)$	no
Simon-like	$O(n\sqrt{2^n})$	$O(n)$	$O(n)$	no

Conclusions

- PGM and Simon-like are suboptimal in some cases
- the Grover-like algorithm fails when lots of Fourier coefficients are equal to zero

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$
  - Extracting parity $w \cdot s$: $O\big(1/\hat{F}(w)\big)$

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$
  - Extracting parity $w \cdot s$: $O\big(1/\hat{F}(w)\big)$
- What is the classical query complexity of this problem?

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$
  - Extracting parity $w \cdot s$: $O\big(1/\hat{F}(w)\big)$
- What is the classical query complexity of this problem?
- What can we say about the time complexity?

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$
  - Extracting parity $w \cdot s$: $O\big(1/\hat{F}(w)\big)$
- What is the classical query complexity of this problem?
- What can we say about the time complexity?
- Generalize everything from $\mathbb{Z}_2$ to $\mathbb{Z}_d$

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$
  - Extracting parity $w \cdot s$: $O\big(1/\hat{F}(w)\big)$
- What is the classical query complexity of this problem?
- What can we say about the time complexity?
- Generalize everything from $\mathbb{Z}_2$ to $\mathbb{Z}_d$
- Applications

Open problems

The main goals

- Find an optimal quantum query algorithm for solving BHSP
- Prove a matching quantum query lower bound

Intermediate problems

- Find an intermediate class of functions as a new test case
  - Decision trees?
- Related problems:
  - Verification of s: $O\big(1/\sqrt{I_f}\big)$
  - Extracting parity $w \cdot s$: $O\big(1/\hat{F}(w)\big)$
- What is the classical query complexity of this problem?
- What can we say about the time complexity?
- Generalize everything from $\mathbb{Z}_2$ to $\mathbb{Z}_d$
- Applications
  - Breaking cryptosystems?

...any questions?

Algorithm 2: Pretty good measurement

Why does it work?

- States: $|\Phi^t(s)\rangle := \sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} |\mathcal{F}_w^t\rangle |w\rangle$

Algorithm 2: Pretty good measurement

Why does it work?

- States: $|\Phi^t(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}|\mathcal{F}_w^t\rangle|w\rangle$
  where $\||\mathcal{F}_w^t\rangle\|_2^2 = \left[\hat{F}^2\right]^{*t}(w) = \frac{1}{\sqrt{2^n}}\widehat{(F*F)^t}(w)$

Algorithm 2: Pretty good measurement

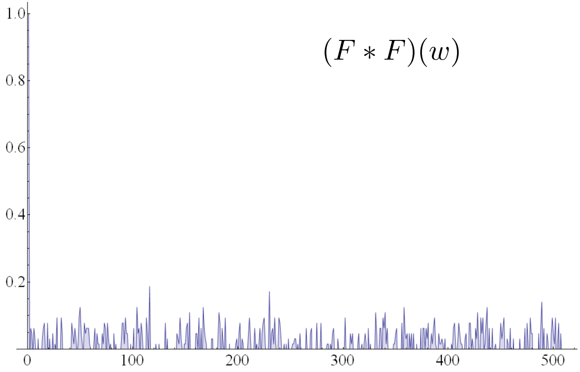
Why does it work?

- States: $|\Phi^t(s)\rangle := \sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} |\mathcal{F}_w^t\rangle |w\rangle$
  where $\| |\mathcal{F}_w^t\rangle \|_2^2 = \left[\hat{F}^2\right]^{*t}(w) = \frac{1}{\sqrt{2^n}} \widehat{(F * F)^t}(w)$
- Convolution: $(F * F)(w) = \sum_{x \in \mathbb{Z}_2^n} F(x) F(w - x)$

Algorithm 2: Pretty good measurement

Why does it work?

- States: $|\Phi^t(s)\rangle := \sum_{w \in \mathbb{Z}_2^n} (-1)^{s \cdot w} |\mathcal{F}_w^t\rangle |w\rangle$
  where $\| |\mathcal{F}_w^t\rangle \|_2^2 = \left[\hat{F}^2\right]^{*t}(w) = \frac{1}{\sqrt{2^n}} \widehat{(F * F)^t}(w)$
- Convolution: $(F * F)(w) = \sum_{x \in \mathbb{Z}_2^n} F(x) F(w - x)$



Algorithm 2: Pretty good measurement

Why does it work?

- States: $|\Phi^t(s)\rangle := \sum_{w\in\mathbb{Z}_2^n}(-1)^{s\cdot w}|\mathcal{F}_w^t\rangle|w\rangle$
  where $\||\mathcal{F}_w^t\rangle\|_2^2 = \left[\hat{F}^2\right]^{*t}(w) = \frac{1}{\sqrt{2^n}}\,\widehat{(F*F)^t}\,(w)$
- Convolution: $(F*F)(w) = \sum_{x\in\mathbb{Z}_2^n} F(x)F(w-x)$