

# Quantum rejection sampling

Maris Ozols

University of Waterloo



Martin Rötteler

NEC Laboratories America

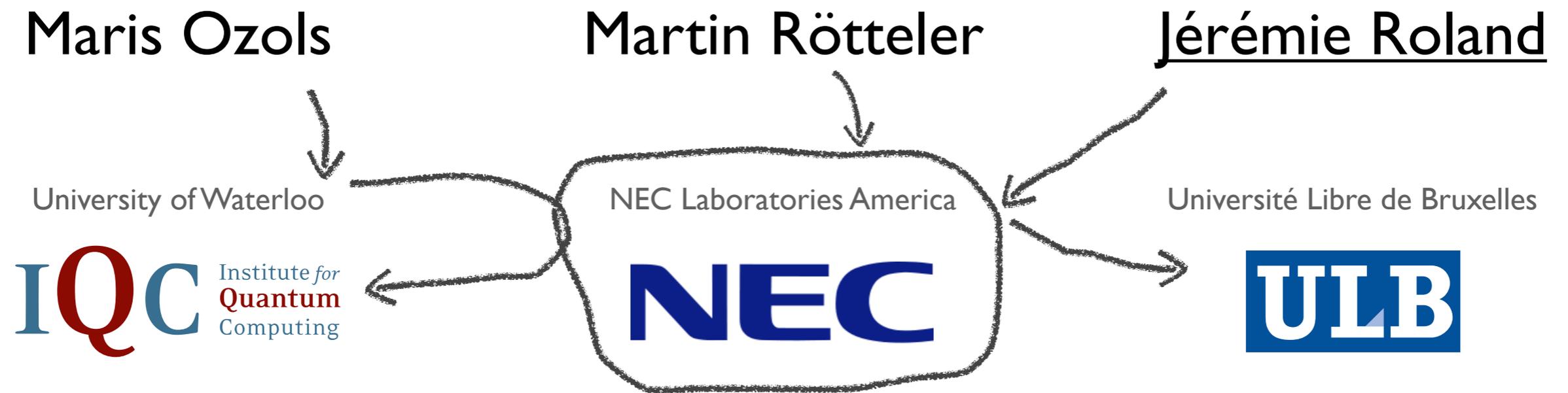


Jérémie Roland

Université Libre de Bruxelles



# Quantum rejection sampling



# Introduction

- (Classical) rejection sampling
  - Algorithmic tool introduced by von Neumann (1951)
  - Can be used to sample from arbitrary distributions
  - Numerous applications:



# Introduction

- (Classical) rejection sampling
  - Algorithmic tool introduced by von Neumann (1951)
  - Can be used to sample from arbitrary distributions
  - Numerous applications:
    - \* Metropolis algorithm [MRRTT53]
    - \* Monte-Carlo simulations
    - \* optimization (simulated annealing)
    - \* etc...



# Introduction

## ○ (Classical) rejection sampling

- Algorithmic tool introduced by von Neumann (1951)
- Can be used to sample from arbitrary distributions
- Numerous applications:
  - \* Metropolis algorithm [MRRTT53]
  - \* Monte-Carlo simulations
  - \* optimization (simulated annealing)
  - \* etc...

## ○ Quantum rejection sampling

- Natural quantum analogue: probabilities  $\rightarrow$  amplitudes
- New algorithmic tool
- Applications:



# Introduction

## ○ (Classical) rejection sampling

- Algorithmic tool introduced by von Neumann (1951)
- Can be used to sample from arbitrary distributions
- Numerous applications:
  - \* Metropolis algorithm [MRRTT53]
  - \* Monte-Carlo simulations
  - \* optimization (simulated annealing)
  - \* etc...

## ○ Quantum rejection sampling

- Natural quantum analogue: probabilities → amplitudes
- New algorithmic tool
- Applications:
  - \* Linear system of equations [HHL09]
  - \* Quantum Metropolis algorithm
  - \* Boolean hidden shift problem



# Classical resampling problem

Setup:

- $P$  and  $S$ : two probability distributions

# Classical resampling problem

Setup:

- $P$  and  $S$ : two probability distributions

## Resampling problem

Given the ability to sample according to  $P$ , produce a sample distributed according to  $S$ .

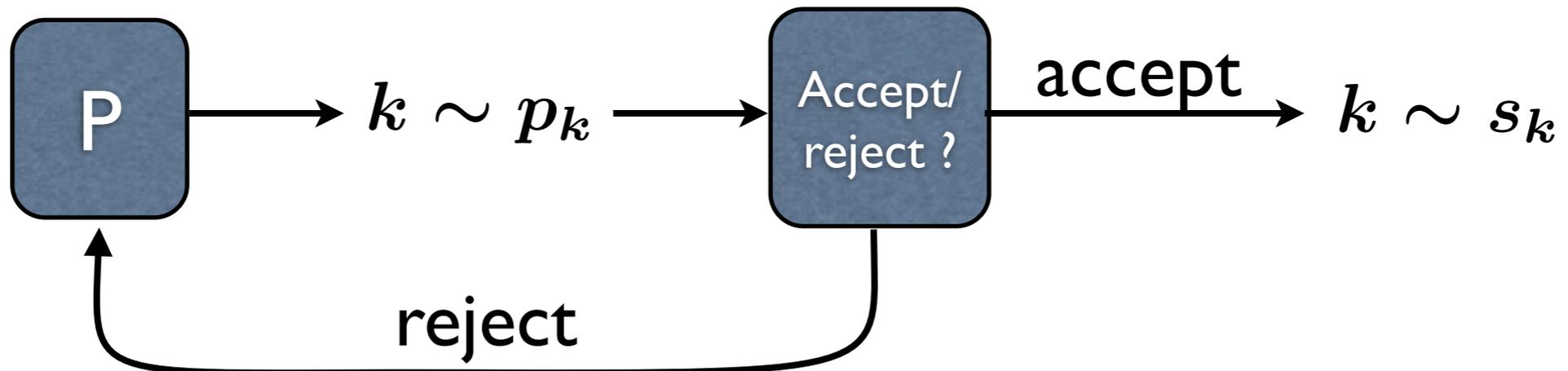
# Classical resampling problem

Setup:

- $P$  and  $S$ : two probability distributions

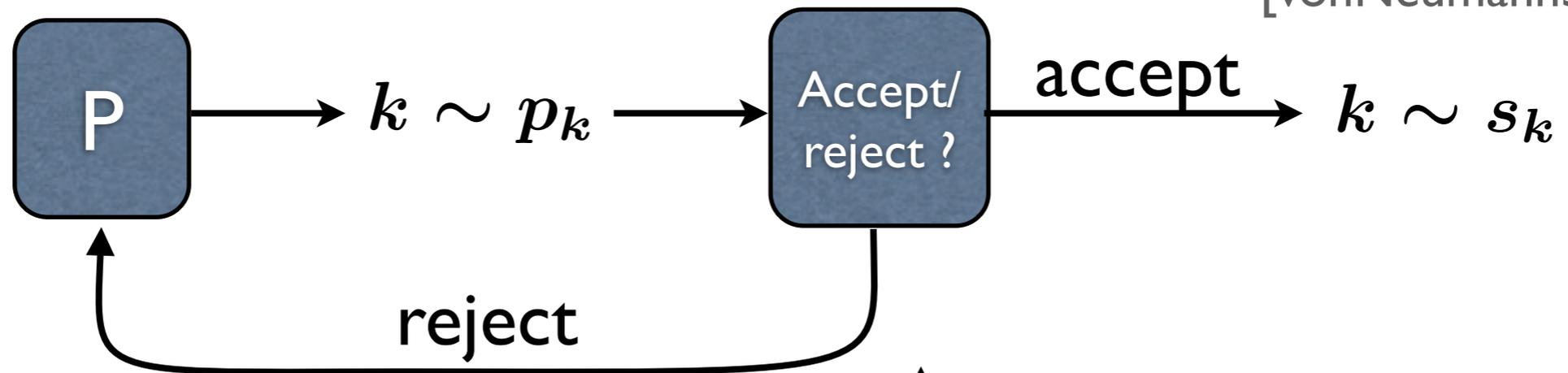
Resampling problem

Given the ability to sample according to  $P$ , produce a sample distributed according to  $S$ .

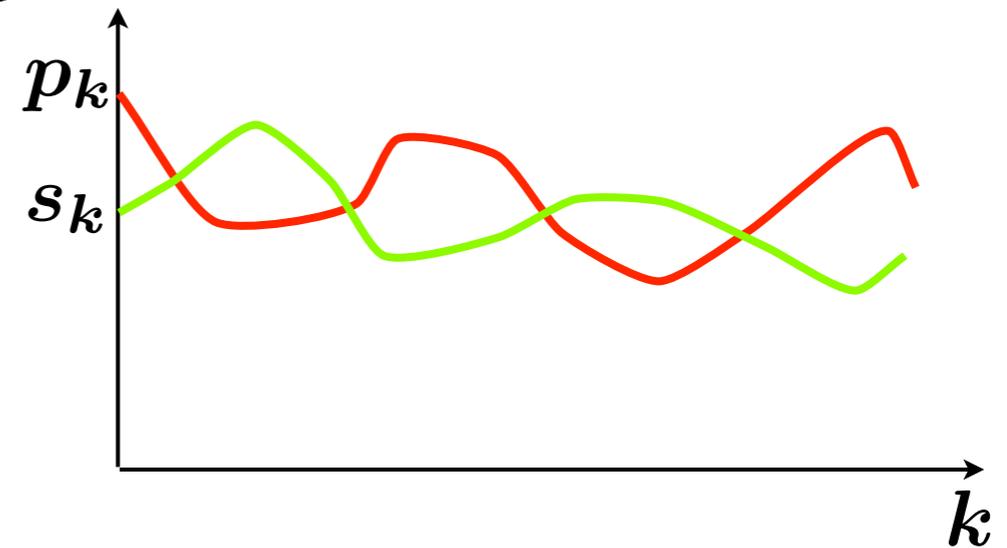


# Rejection sampling

[vonNeumann51]

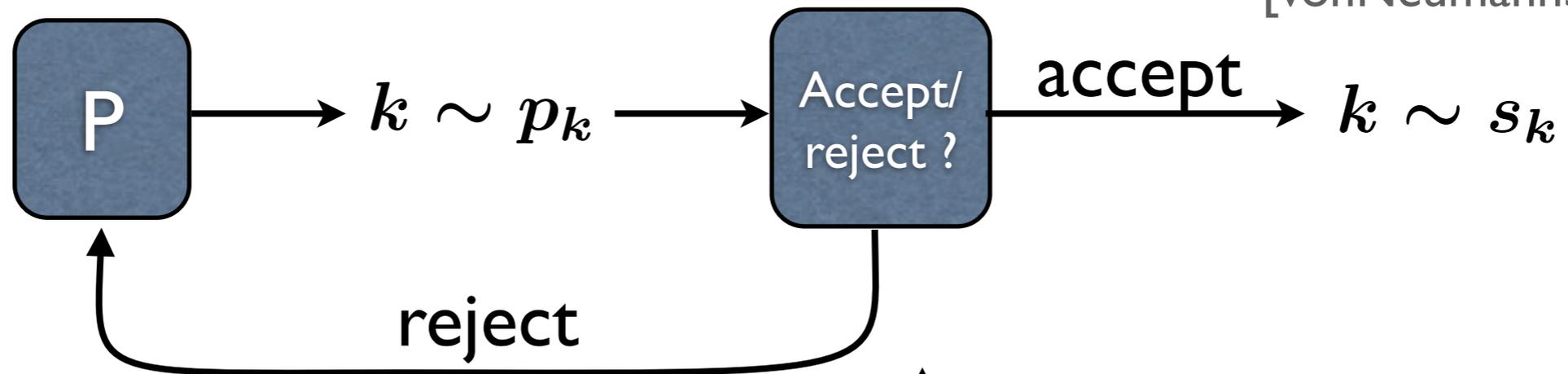


○  $\Pr[\text{accept } k] = \gamma \frac{s_k}{p_k}$



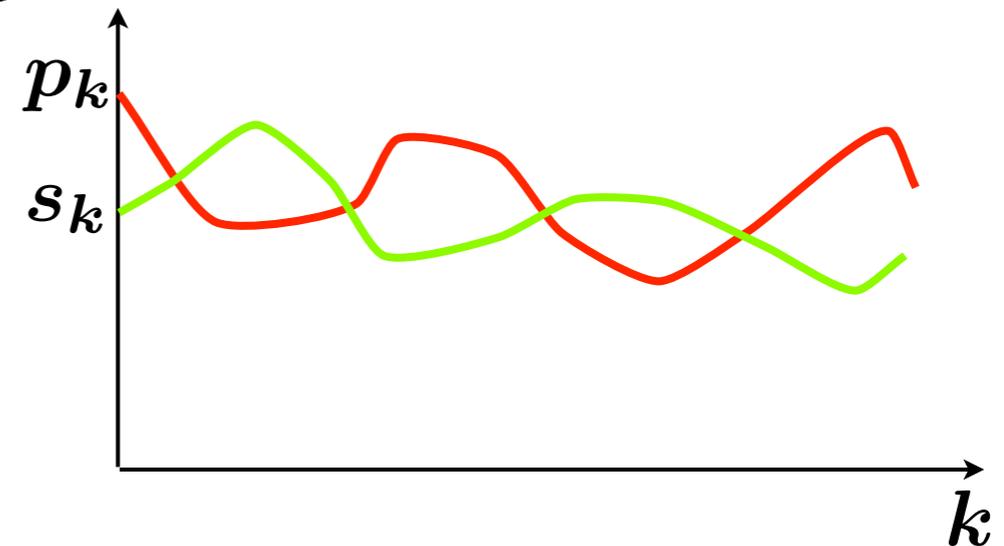
# Rejection sampling

[vonNeumann51]



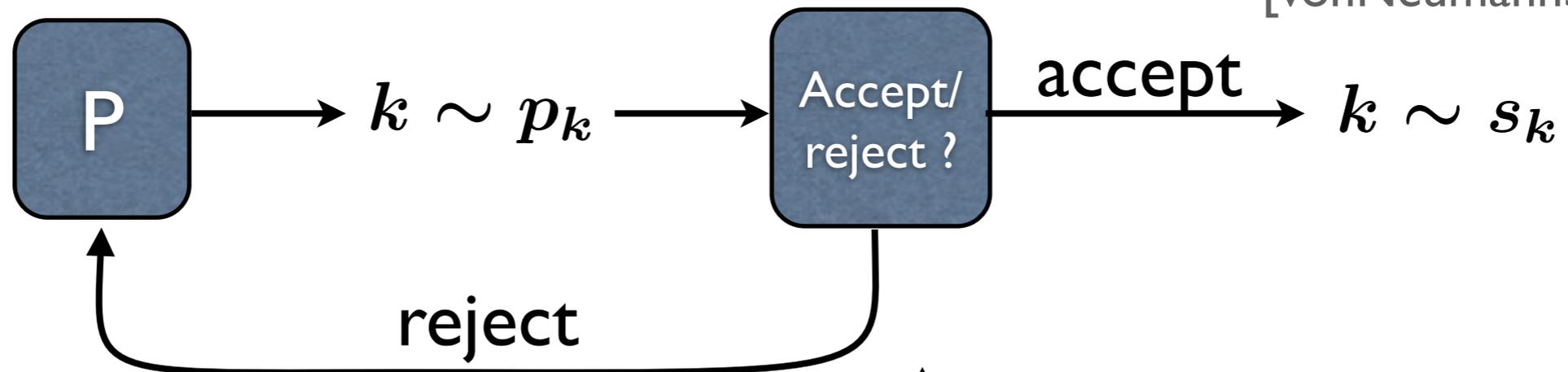
○  $\Pr[\text{accept } k] = \gamma \frac{s_k}{p_k}$

○  $\gamma = \min_k \frac{p_k}{s_k}$



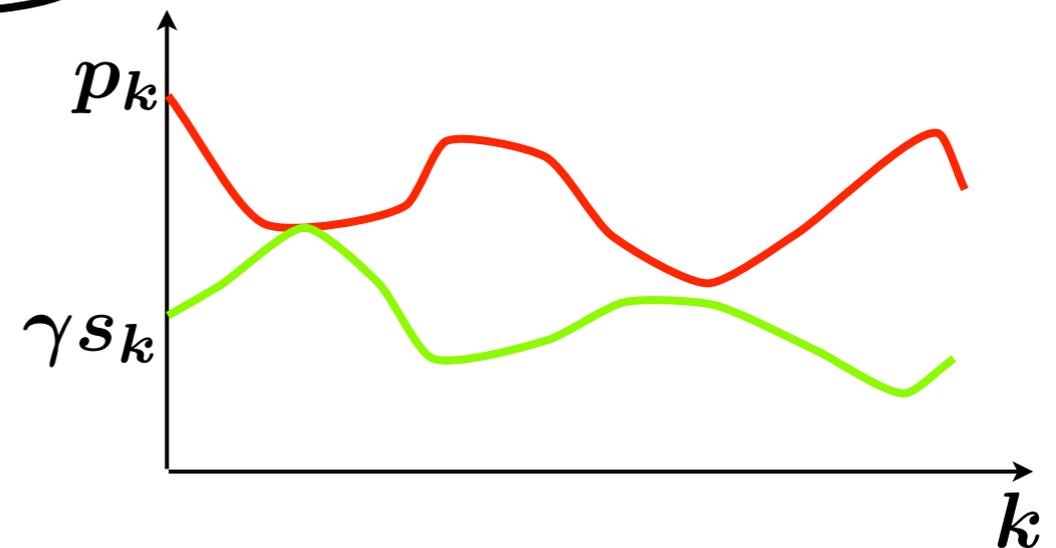
# Rejection sampling

[vonNeumann51]



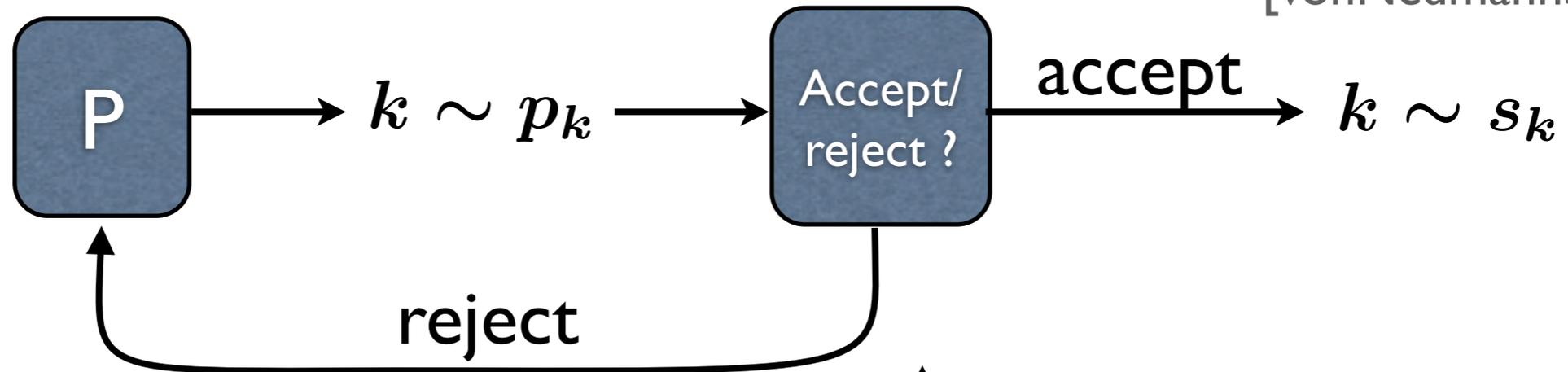
○  $\Pr[\text{accept } k] = \gamma \frac{s_k}{p_k}$

○  $\gamma = \min_k \frac{p_k}{s_k}$



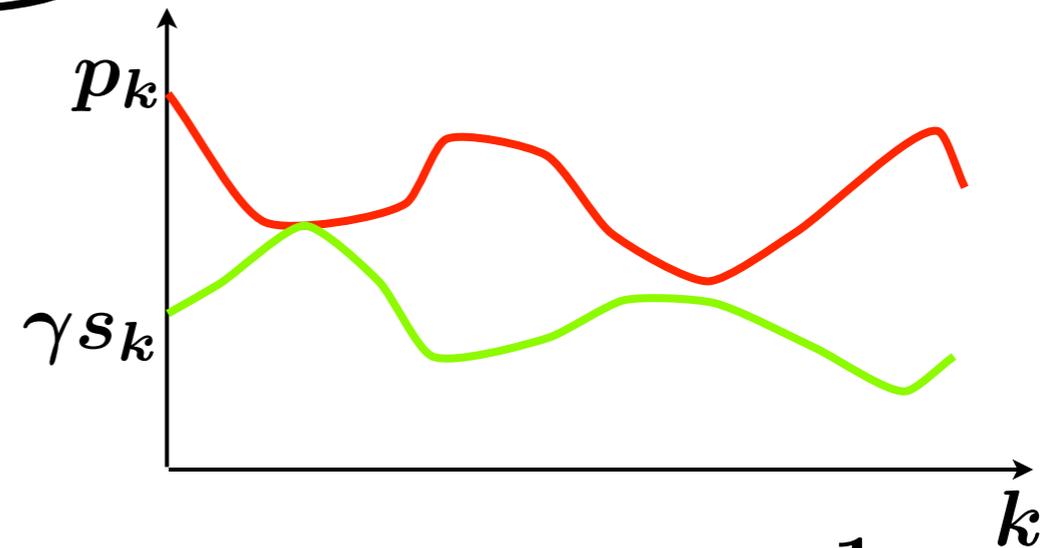
# Rejection sampling

[vonNeumann51]



○  $\Pr[\text{accept } k] = \gamma \frac{s_k}{p_k}$

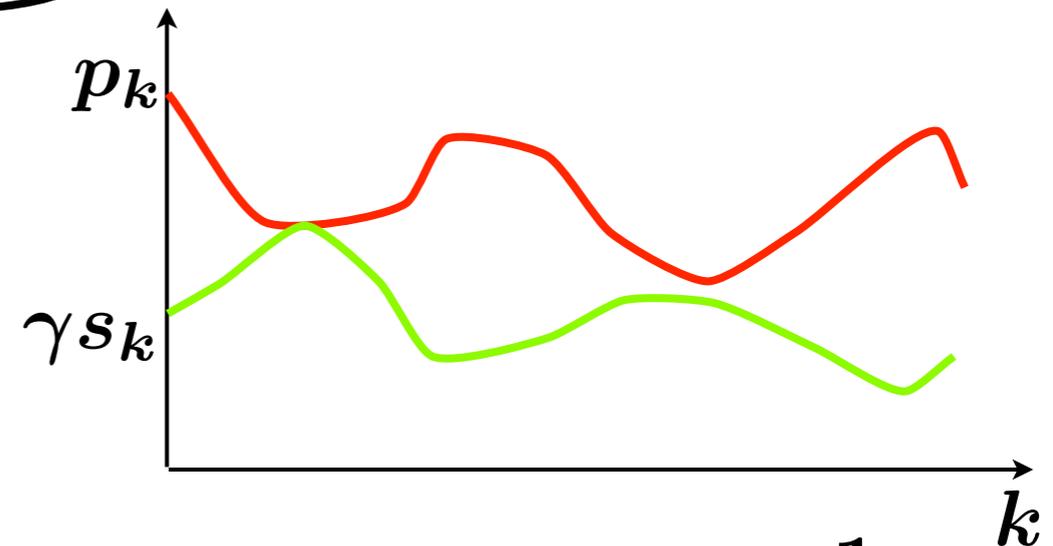
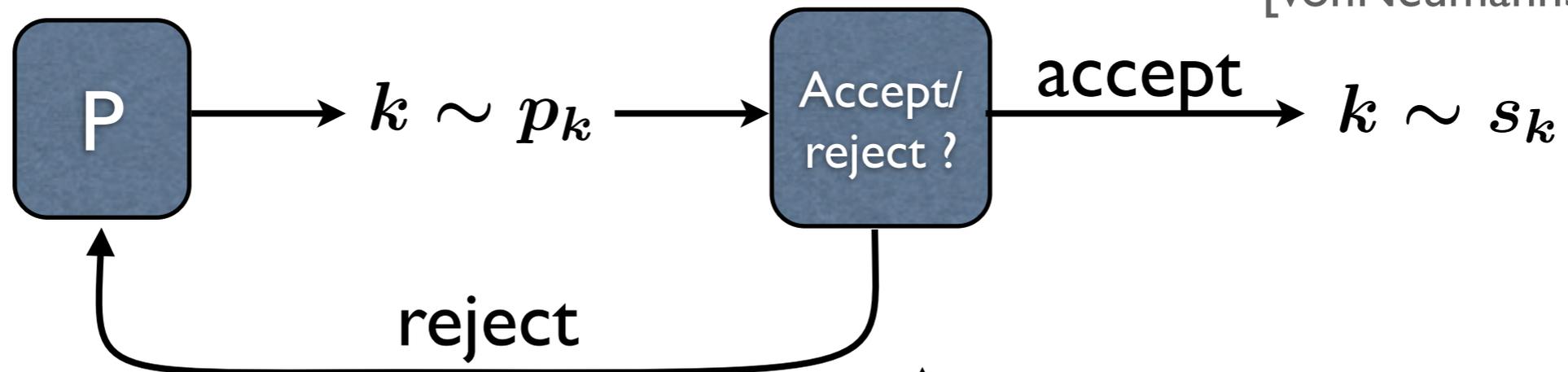
○  $\gamma = \min_k \frac{p_k}{s_k}$



○ Expected number of required samples:  $T = \frac{1}{\gamma}$

# Rejection sampling

[vonNeumann51]



○  $\Pr[\text{accept } k] = \gamma \frac{s_k}{p_k}$

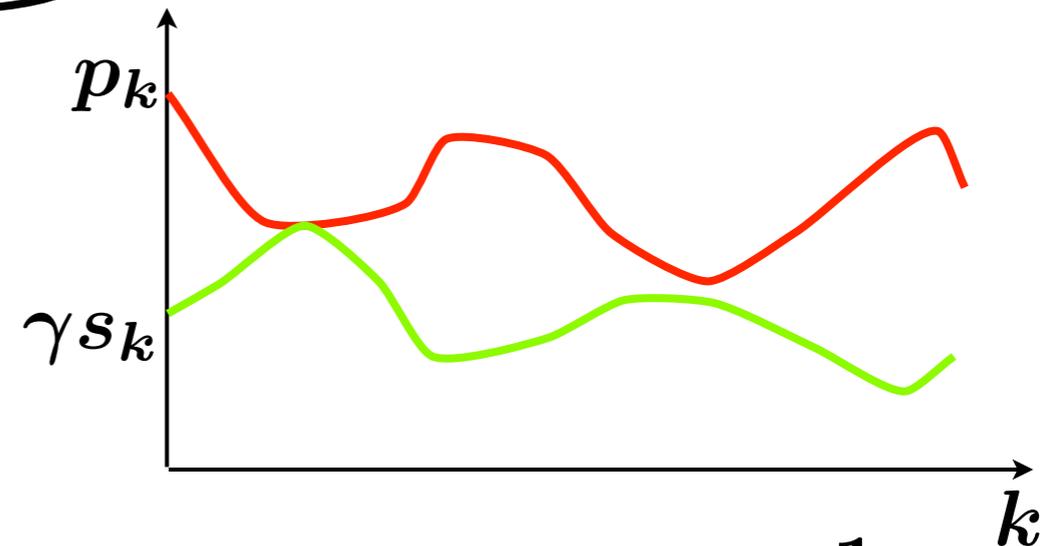
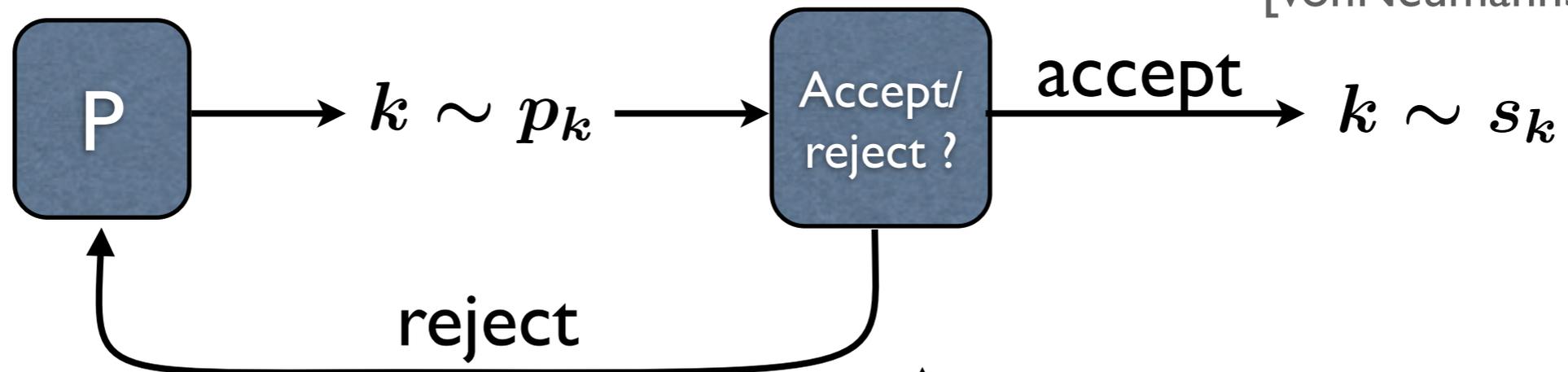
○  $\gamma = \min_k \frac{p_k}{s_k}$

○ Expected number of required samples:  $T = \frac{1}{\gamma}$

○ This is optimal [Letac75]

# Rejection sampling

[vonNeumann51]



○  $\Pr[\text{accept } k] = \gamma \frac{s_k}{p_k}$

○  $\gamma = \min_k \frac{p_k}{s_k}$

○ Expected number of required samples:  $T = \frac{1}{\gamma}$

○ This is optimal [Letac75]

○ Many applications in randomized algorithms

# Quantum resampling problem

○ Given access to a black box  $O_\xi$  preparing a state

$$|\pi^\xi\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$



# Quantum resampling problem

- Given access to a black box  $O_\xi$  preparing a state

$$|\pi^\xi\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

known amplitudes

unknown states

# Quantum resampling problem

- Given access to a black box  $O_\xi$  preparing a state

$$|\pi^\xi\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

known amplitudes

unknown states

- Prepare the state

$$|\sigma^\xi\rangle = \sum_k \sigma_k |\xi_k\rangle |k\rangle$$

# Quantum resampling problem

- Given access to a black box  $O_\xi$  preparing a state

$$|\pi^\xi\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

known amplitudes

unknown states

- Prepare the state

$$|\sigma^\xi\rangle = \sum_k \sigma_k |\xi_k\rangle |k\rangle$$

different amplitudes

same states

# Quantum resampling problem

- Given access to a black box  $O_\xi$  preparing a state

$$|\pi^\xi\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

known amplitudes

unknown states

- Prepare the state

$$|\sigma^\xi\rangle = \sum_k \sigma_k |\xi_k\rangle |k\rangle$$

different amplitudes

same states

- Question: How many calls to  $O_\xi$  are necessary?

- Tool: Query complexity

# Classical query complexity

- Function  $f(x)$ , where  $x = (x_1, \dots, x_n)$
- Oracle  $O_x : i \rightarrow x_i$
- Goal: Compute  $f(x)$  given black-box access to  $O_x$

# Classical query complexity

- Function  $f(x)$ , where  $x = (x_1, \dots, x_n)$
- Oracle  $O_x : i \rightarrow x_i$
- Goal: Compute  $f(x)$  given black-box access to  $O_x$

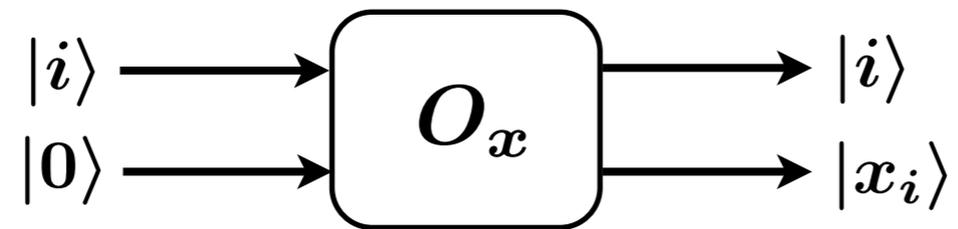
Randomized query complexity  $R_\epsilon(f)$

Minimum # calls to  $O_x$  necessary to compute  $f(x)$  with success probability  $(1 - \epsilon)$

# Quantum query complexity

Different quantum extensions:

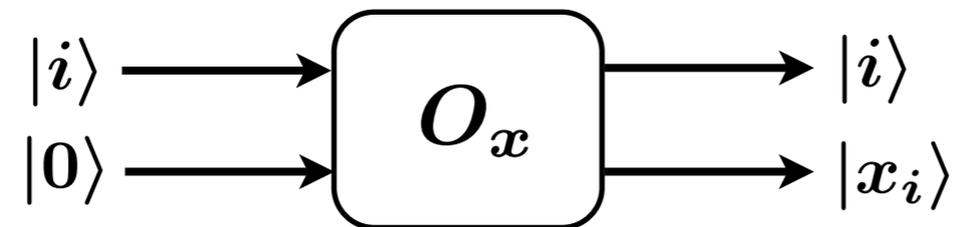
I. Can query  $O_x$  in superposition  $\Rightarrow Q_\epsilon(f) \leq R_\epsilon(f)$



# Quantum query complexity

Different quantum extensions:

1. Can query  $O_x$  in superposition  $\Rightarrow Q_\epsilon(f) \leq R_\epsilon(f)$

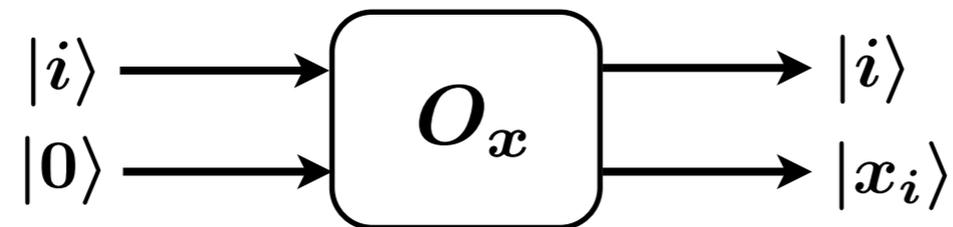


2. Instead of computing a function  $f(x)$ , generate a quantum state  $|x\rangle$

# Quantum query complexity

Different quantum extensions:

1. Can query  $O_x$  in superposition  $\Rightarrow Q_\epsilon(f) \leq R_\epsilon(f)$



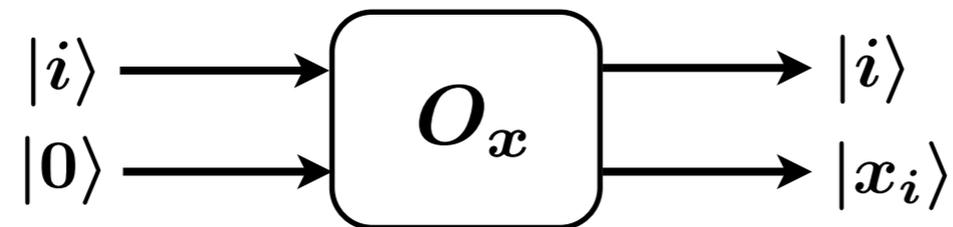
2. Instead of computing a function  $f(x)$ , generate a quantum state  $|x\rangle$

3. Oracle  $O_\xi$  is a unitary that hides the label  $\xi$  in a non-explicit way

# Quantum query complexity

Different quantum extensions:

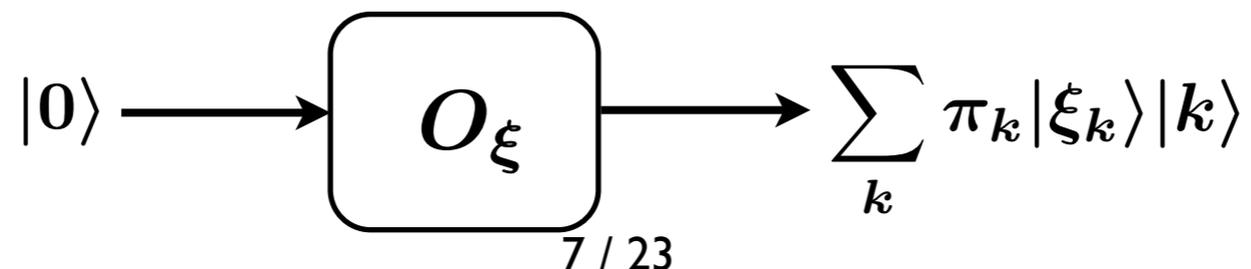
1. Can query  $O_x$  in superposition  $\Rightarrow Q_\epsilon(f) \leq R_\epsilon(f)$



2. Instead of computing a function  $f(x)$ , generate a quantum state  $|x\rangle$

3. Oracle  $O_\xi$  is a unitary that hides the label  $\xi$  in a non-explicit way

Example: Quantum resampling



# Quantum state generation

- Set of quantum states  $\mathcal{X} = \{|\xi\rangle : \xi \in \mathcal{X}\}$
- Set of oracles  $\mathcal{O} = \{O_\xi : \xi \in \mathcal{X}\}$
- Quantum state generation problem  $\mathcal{P}$  defined by  $(\mathcal{X}, \mathcal{O})$
- Goal: Generate  $|\xi\rangle$  given black-box access to  $O_\xi$

# Quantum state generation

- Set of quantum states  $= \{|\xi\rangle : \xi \in \mathcal{X}\}$
- Set of oracles  $\mathcal{O} = \{O_\xi : \xi \in \mathcal{X}\}$
- Quantum state generation problem  $\mathcal{P}$  defined by  $(\mathcal{X}, \mathcal{O})$
- Goal: Generate  $|\xi\rangle$  given black-box access to  $O_\xi$

Quantum query complexity  $Q_\epsilon(\mathcal{P})$

Minimum # calls to  $O_\xi$  necessary to generate a state  $\sqrt{1-\epsilon}|\xi\rangle|\bar{0}\rangle + \sqrt{\epsilon}|\text{error}_\xi\rangle$

work space

# Quantum rejection sampling

○ Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

# Quantum rejection sampling

○ Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow \sigma_k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow \sigma_k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

- Use control-rotation on an ancilla qubit

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow \sigma_k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

- Use control-rotation on an ancilla qubit

$$\sum_k \pi_k |\xi_k\rangle |k\rangle$$

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow \sigma_k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

- Use control-rotation on an ancilla qubit

$$\sum_k \pi_k |\xi_k\rangle |k\rangle |0\rangle$$

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow \sigma_k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi|0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

Will be chosen later

- Use control-rotation on an ancilla qubit

$$\sum_k \pi_k |\xi_k\rangle |k\rangle \widehat{\otimes} |0\rangle \rightarrow \sum_k |\xi_k\rangle |k\rangle (\sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle)$$

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi |0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

Will be chosen later

- Use control-rotation on an ancilla qubit

$$\sum_k \pi_k |\xi_k\rangle |k\rangle \widehat{\otimes} |0\rangle \rightarrow \sum_k |\xi_k\rangle |k\rangle \sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle$$

- If we measure the ancilla and obtain  $|1\rangle$  (“accept”):

$$\frac{1}{\|\vec{\alpha}\|} \sum_k \alpha_k |\xi_k\rangle |k\rangle$$

# Quantum rejection sampling

Need to change  $\pi_k \rightarrow k$ .

- Use oracle  $O_\xi$  to create the original state

$$O_\xi |0\rangle = \sum_k \pi_k |\xi_k\rangle |k\rangle$$

Will be chosen later

- Use control-rotation on an ancilla qubit

$$\sum_k \pi_k |\xi_k\rangle |k\rangle \otimes \widehat{0} \rightarrow \sum_k |\xi_k\rangle |k\rangle \sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle$$

- If we measure the ancilla and obtain  $|1\rangle$  (“accept”):

$$\frac{1}{\|\vec{\alpha}\|} \sum_k \alpha_k |\xi_k\rangle |k\rangle$$

- OK if  $\vec{\alpha}$  is close to  $\vec{\sigma}$ , more precisely:

$$\frac{\vec{\sigma} \cdot \vec{\alpha}}{\|\vec{\alpha}\|} \geq \sqrt{1 - \epsilon}$$

# Optimization

$$\sum_k |\xi_k\rangle |k\rangle \sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle$$

- We measure  $|1\rangle$  (“accept”) with probability  $\|\vec{\alpha}\|^2$

# Optimization

$$\sum_k |\xi_k\rangle |k\rangle \sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle$$

- We measure  $|1\rangle$  (“accept”) with probability  $\|\vec{\alpha}\|^2$
- Naive approach: repeat  $O(1/\|\vec{\alpha}\|^2)$  times

# Optimization

$$\sum_k |\xi_k\rangle |k\rangle \sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle$$

- We measure  $|1\rangle$  (“accept”) with probability  $\|\vec{\alpha}\|^2$
- Naive approach: repeat  $O(1/\|\vec{\alpha}\|^2)$  times
- Using amplitude amplification: reduce to  $O(1/\|\vec{\alpha}\|)$

[BrassardHøyerMoscaTapp00]

# Optimization

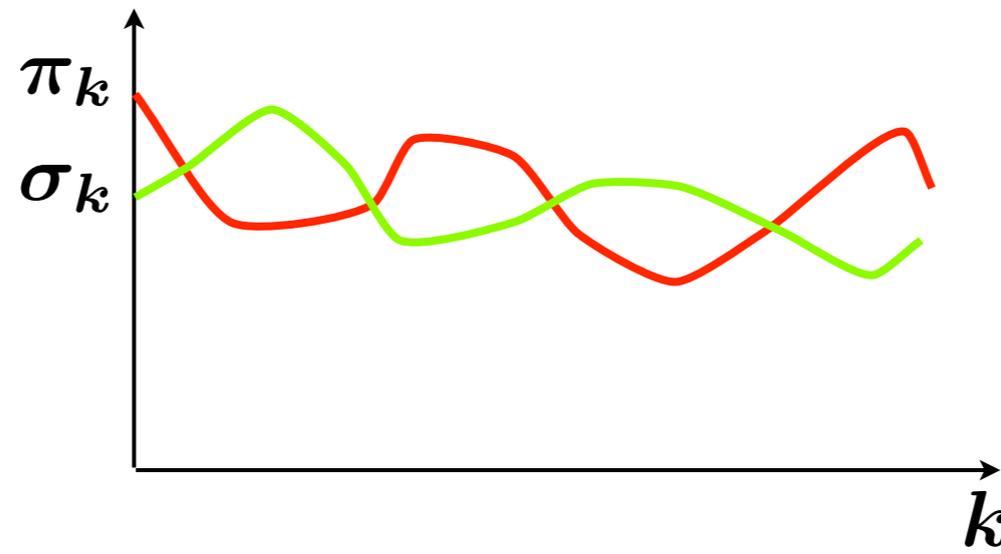
$$\sum_k |\xi_k\rangle |k\rangle \sqrt{|\pi_k|^2 - |\alpha_k|^2} |0\rangle + \alpha_k |1\rangle$$

- We measure  $|1\rangle$  (“accept”) with probability  $\|\vec{\alpha}\|^2$
- Naive approach: repeat  $O(1/\|\vec{\alpha}\|^2)$  times
- Using amplitude amplification: reduce to  $O(1/\|\vec{\alpha}\|)$   
[BrassardHøyerMoscaTapp00]
- Optimizing  $\vec{\alpha}$  : Semidefinite program

$$\text{Maximize } \|\vec{\alpha}\| \quad \text{subject to } 0 \leq \alpha_k \leq \pi_k \quad \forall k$$
$$\frac{\vec{\sigma} \cdot \vec{\alpha}}{\|\vec{\alpha}\|} \geq \sqrt{1 - \epsilon}$$

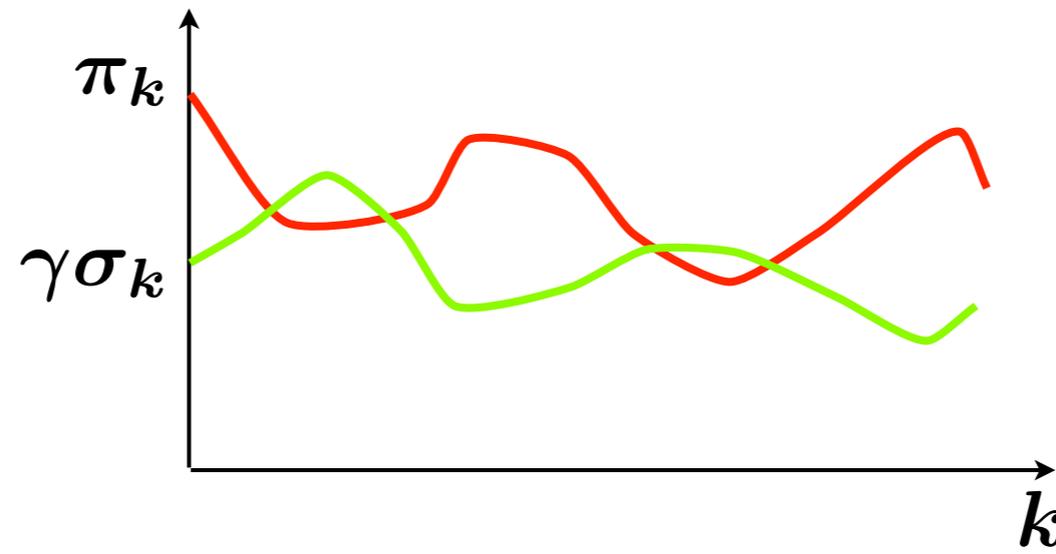
# Optimal solution

○ Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$



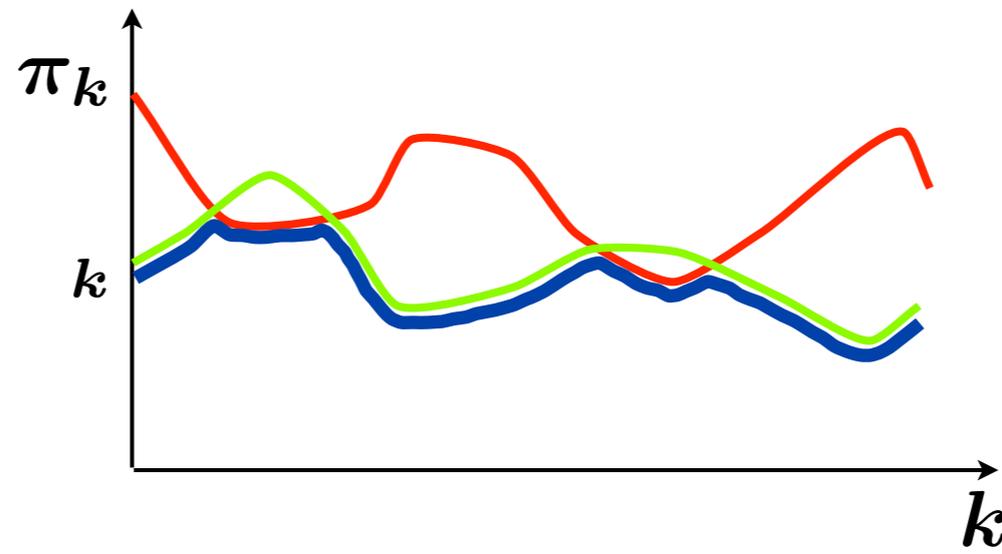
# Optimal solution

○ Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$



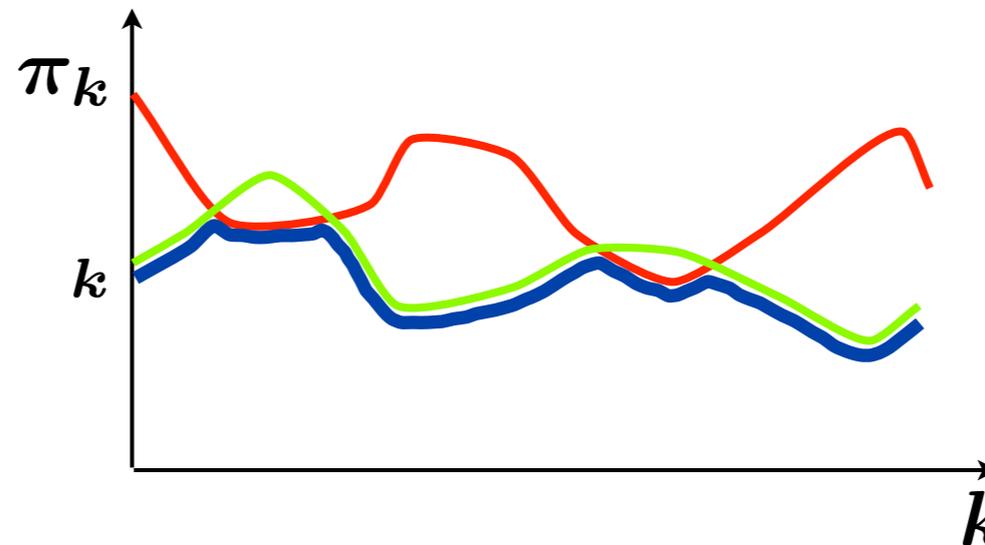
# Optimal solution

○ Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$



# Optimal solution

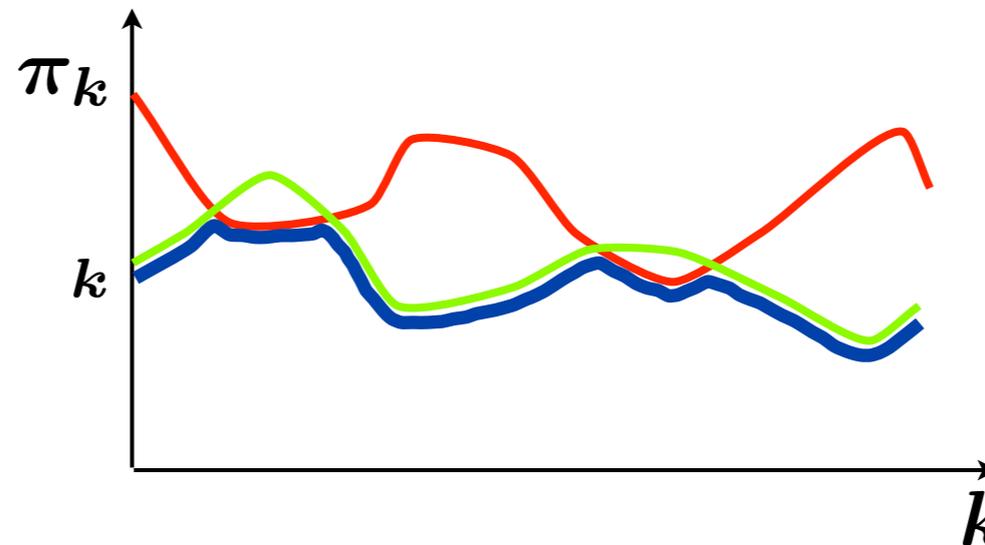
○ Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$



○ We take  $\bar{\gamma} = \max \gamma$  such that  $\frac{\vec{\sigma} \cdot \vec{\alpha}(\gamma)}{\|\vec{\alpha}(\gamma)\|} \geq \sqrt{1 - \varepsilon}$

# Optimal solution

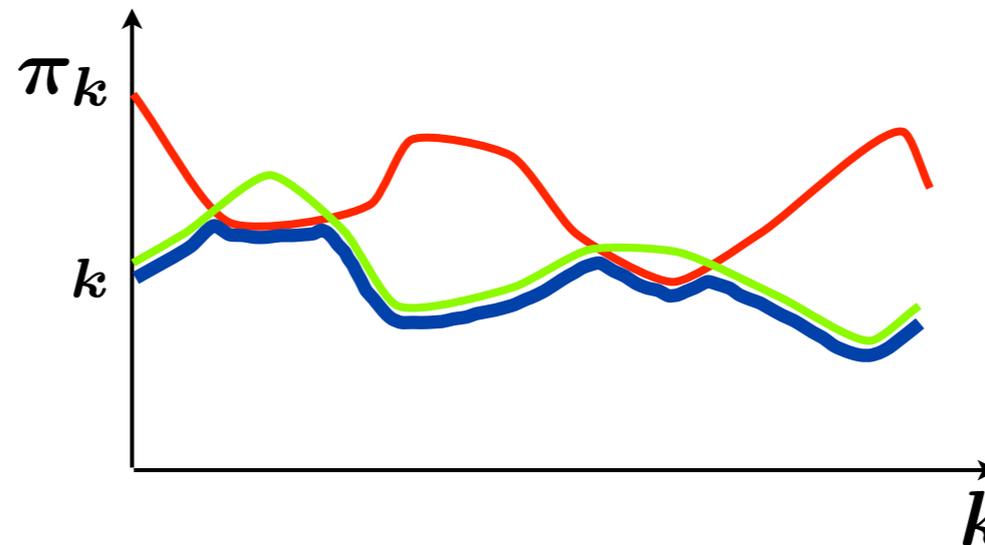
○ Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$



- We take  $\bar{\gamma} = \max \gamma$  such that  $\frac{\sum_k \sigma_k \alpha_k(\gamma)}{\|\vec{\alpha}(\gamma)\|} \geq \sqrt{1 - \epsilon}$
- We can prove that this leads to an optimal algorithm

# Optimal solution

○ Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$

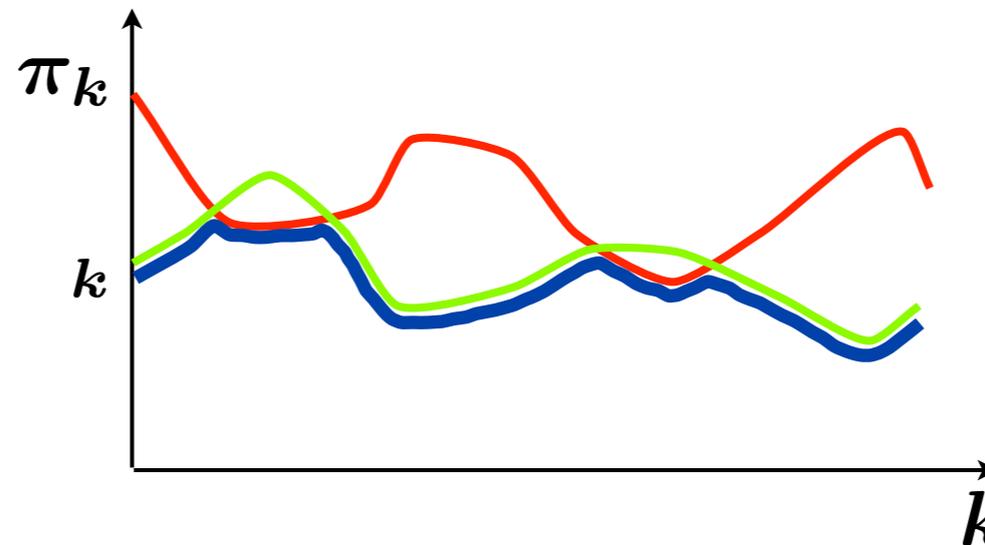


- We take  $\bar{\gamma} = \max \gamma$  such that  $\frac{\vec{\sigma} \vec{\alpha}(\gamma)}{\|\vec{\alpha}(\gamma)\|} \geq \sqrt{1 - \epsilon}$
- We can prove that this leads to an optimal algorithm

Matching lower bound uses automorphism principle with  $G = \mathbb{Z}_2^n \times U(N - 1)$

# Optimal solution

- Let  $\alpha_k(\gamma) = \min\{\pi_k, \gamma\sigma_k\}$



- We take  $\bar{\gamma} = \max \gamma$  such that  $\frac{\vec{\sigma} \vec{\alpha}(\gamma)}{\|\vec{\alpha}(\gamma)\|} \geq \sqrt{1 - \epsilon}$
- We can prove that this leads to an optimal algorithm

Matching lower bound uses automorphism principle with  $G = \mathbb{Z}_2^n \times U(N - 1)$

## Theorem

$$Q_\epsilon(\text{QSampling}_{\vec{\pi} \rightarrow \vec{\sigma}}) = \Theta(1 / \|\vec{\alpha}(\bar{\gamma})\|)$$

# Applications

- Linear system of equations [HHL09]
  - QRS was used implicitly
- Quantum Metropolis algorithm
  - Improvement on the original algorithm [TOVPV11]
- Boolean hidden shift problem
  - New algorithm!



# Linear system of equations

[HHL09]

Setup:

○ Invertible  $d \times d$  matrix  $A$

○ Vector  $|b\rangle \in \mathbb{C}^d$

can be assumed Hermitian

# Linear system of equations

[HHL09]

Setup:

○ Invertible  $d \times d$  matrix  $A$

○ Vector  $|b\rangle \in \mathbb{C}^d$

can be assumed Hermitian

Quantum linear equations problem

Prepare the state  $|x\rangle$  such that

$$A|x\rangle = |b\rangle$$

# Linear system of equations

[HHL09]

Setup:

○ Invertible  $d \times d$  matrix  $A$

○ Vector  $|b\rangle \in \mathbb{C}^d$

can be assumed Hermitian

Quantum linear equations problem

Prepare the state  $|x\rangle$  such that

$$A|x\rangle = |b\rangle$$

Main idea: use quantum phase estimation (QPE) [Kitaev95,CEMM97]

+ quantum rejection sampling (QRS)

# Algorithm

[HarrowHassidimLloyd09]

○ Let  $|b\rangle = \sum_k b_k |k\rangle$ , where

- $|k\rangle$  are the eigenstates of  $A$
- $\lambda_k$  are the corresponding eigenvalues

$$\begin{aligned} A|x\rangle &= |b\rangle \\ &\Leftrightarrow \\ |x\rangle &= A^{-1}|b\rangle \end{aligned}$$



# Algorithm

[HarrowHassidimLloyd09]

○ Let  $|b\rangle = \sum_k b_k |k\rangle$ , where

□  $|k\rangle$  are the eigenstates of  $A$

□  $\lambda_k$  are the corresponding eigenvalues

$$\begin{aligned} A|x\rangle &= |b\rangle \\ &\Leftrightarrow \\ |x\rangle &= A^{-1}|b\rangle \end{aligned}$$

○ Use QPE to prepare  $|b\rangle = \sum_k b_k |k\rangle |\lambda_k\rangle$



# Algorithm

[HarrowHassidimLloyd09]

○ Let  $|b\rangle = \sum_k b_k |k\rangle$ , where

□  $|k\rangle$  are the eigenstates of  $A$

□  $\lambda_k$  are the corresponding eigenvalues

$$\begin{aligned} A|x\rangle &= |b\rangle \\ \Leftrightarrow \\ |x\rangle &= A^{-1}|b\rangle \end{aligned}$$

○ Use QPE to prepare  $|b\rangle = \sum_k b_k |k\rangle |k\rangle$

○ Use QRS to get  $\sum_k b_k \lambda_k^{-1} |k\rangle |\lambda_k\rangle$

□ Known amplitude (ratios):  $\lambda_k^{-1}$

□ Unknown states:  $|k\rangle$



# Algorithm

[HarrowHassidimLloyd09]

○ Let  $|b\rangle = \sum_k b_k |k\rangle$ , where

□  $|k\rangle$  are the eigenstates of  $A$

□  $\lambda_k$  are the corresponding eigenvalues

$$\begin{aligned} A|x\rangle &= |b\rangle \\ \Leftrightarrow \\ |x\rangle &= A^{-1}|b\rangle \end{aligned}$$

○ Use QPE to prepare  $|b\rangle = \sum_k b_k |k\rangle |k\rangle$

○ Use QRS to get  $\sum_k b_k \lambda_k^{-1} |k\rangle |\lambda_k\rangle$

□ Known amplitude (ratios):  $\lambda_k^{-1}$

□ Unknown states:  $|k\rangle$

○ Undo phase estimation to obtain

$$|x\rangle = \sum_k b_k \lambda_k^{-1} |k\rangle = A^{-1}|b\rangle$$



# Quantum Metropolis algorithm

Setup:

○ Hamiltonian  $H$

□ Eigenstates  $|k\rangle$

□ Eigenenergies  $E_k$

○ Inverse temperature  $\beta$

# Quantum Metropolis algorithm

Setup:

○ Hamiltonian  $H$

□ Eigenstates  $|k\rangle$

□ Eigenenergies  $E_k$

○ Inverse temperature  $\beta$

Metropolis sampling problem

Prepare the thermal state  $\sum_k p_k |k\rangle\langle k|$ ,  
where  $p_k \sim \exp(-\beta E_k)$  is the Gibbs  
distribution

# Classical solution

[MRRTT53]

○ If  $H$  is diagonal (=classical)

□ Eigenstates  $|k\rangle$  are known

□ Eigenenergy  $E_k$  can be efficiently computed from  $|k\rangle$

# Classical solution

[MRRTT53]

○ If  $H$  is diagonal (=classical)

□ Eigenstates  $|k\rangle$  are known

□ Eigenenergy  $E_k$  can be efficiently computed from  $|k\rangle$

○ Start from a random  $|k\rangle$

# Classical solution

[MRRTT53]

○ If  $H$  is diagonal (=classical)

□ Eigenstates  $|k\rangle$  are known

□ Eigenenergy  $E_k$  can be efficiently computed from  $|k\rangle$

○ Start from a random  $|k\rangle$

e.g., spin flip

○ Apply a “kick” to get another  $|l\rangle$

# Classical solution

[MRRTT53]

## ○ If $H$ is diagonal (=classical)

- Eigenstates  $|k\rangle$  are known
- Eigenenergy  $E_k$  can be efficiently computed from  $|k\rangle$

## ○ Start from a random $|k\rangle$

e.g., spin flip

## ○ Apply a “kick” to get another $|l\rangle$

## ○ Compute the energies $E_k$ and $E_l$

- If  $E_l \leq E_k$ , accept the move
- If  $E_l > E_k$ , accept only with probability  $\exp(\beta(E_k - E_l))$

# Classical solution

[MRRTT53]

○ If  $H$  is diagonal (=classical)

□ Eigenstates  $|k\rangle$  are known

□ Eigenenergy  $E_k$  can be efficiently computed from  $|k\rangle$

○ Start from a random  $|k\rangle$

e.g., spin flip

○ Apply a “kick” to get another  $|l\rangle$

○ Compute the energies  $E_k$  and  $E_l$

□ If  $E_l \leq E_k$ , accept the move

□ If  $E_l > E_k$ , accept only with probability  $\exp(\beta(E_k - E_l))$

○ Repeat

# Quantum Metropolis algorithm

○ If  $H$  is not diagonal (=quantum)

- Eigenstates  $|k\rangle$  and eigenenergies  $E_k$  are not known to start with
- But: we can project onto the  $|k\rangle$ -basis and get the corresponding  $E_k$  by using quantum phase estimation (QPE).

# Quantum Metropolis algorithm

○ If  $H$  is not diagonal (=quantum)

- Eigenstates  $|k\rangle$  and eigenenergies  $E_k$  are not known to start with
- But: we can project onto the  $|k\rangle$ -basis and get the corresponding  $E_k$  by using quantum phase estimation (QPE).

○ Prepare a random  $|k\rangle$  using QPE (and record  $E_k$ )

# Quantum Metropolis algorithm

○ If  $H$  is not diagonal (=quantum)

□ Eigenstates  $|k\rangle$  and eigenenergies  $E_k$  are not known to start with

□ But: we can project onto the  $|k\rangle$ -basis and get the corresponding  $E_k$  by using quantum phase estimation (QPE).

○ Prepare a random  $|k\rangle$  using QPE (and record  $E_k$ )

○ Apply a “kick” (random unitary gate)

# Quantum Metropolis algorithm

- If  $H$  is not diagonal (=quantum)
  - Eigenstates  $|k\rangle$  and eigenenergies  $E_k$  are not known to start with
  - But: we can project onto the  $|k\rangle$ -basis and get the corresponding  $E_k$  by using quantum phase estimation (QPE).
- Prepare a random  $|k\rangle$  using QPE (and record  $E_k$ )
- Apply a “kick” (random unitary gate)
- Use QPE to project on another  $|l\rangle$  (and record  $E_l$ )

# Quantum Metropolis algorithm

○ If  $H$  is not diagonal (=quantum)

- Eigenstates  $|k\rangle$  and eigenenergies  $E_k$  are not known to start with
- But: we can project onto the  $|k\rangle$ -basis and get the corresponding  $E_k$  by using quantum phase estimation (QPE).

○ Prepare a random  $|k\rangle$  using QPE (and record  $E_k$ )

○ Apply a “kick” (random unitary gate)

○ Use QPE to project on another  $|l\rangle$  (and record  $E_l$ )

○ Compare the energies  $E_k$  and  $E_l$

- If  $E_l \leq E_k$ , accept the move
- If  $E_l > E_k$ , accept only with probability  $\exp(-\beta(E_l - E_k))$

# Quantum Metropolis algorithm

## ○ Problem:

- Rejected moves require to revert the state from  $|i\rangle$  to  $|k\rangle$



# Quantum Metropolis algorithm

## ○ Problem:

- Rejected moves require to revert the state from  $|l\rangle$  to  $|k\rangle$
- We cannot keep a copy of  $|k\rangle$  (requires to clone an unknown state!)



# Quantum Metropolis algorithm

## ○ Problem:

- Rejected moves require to revert the state from  $|l\rangle$  to  $|k\rangle$
- We cannot keep a copy of  $|k\rangle$  (requires to clone an unknown state!)

## ○ Two solutions:

- Temme *et al.* [TOVPV11] propose a “rewinding” technique to revert to  $|k\rangle$ , based on a series of projective measurements.



# Quantum Metropolis algorithm

## ○ Problem:

- ❑ Rejected moves require to revert the state from  $|l\rangle$  to  $|k\rangle$
- ❑ We cannot keep a copy of  $|k\rangle$  (requires to clone an unknown state!)

## ○ Two solutions:

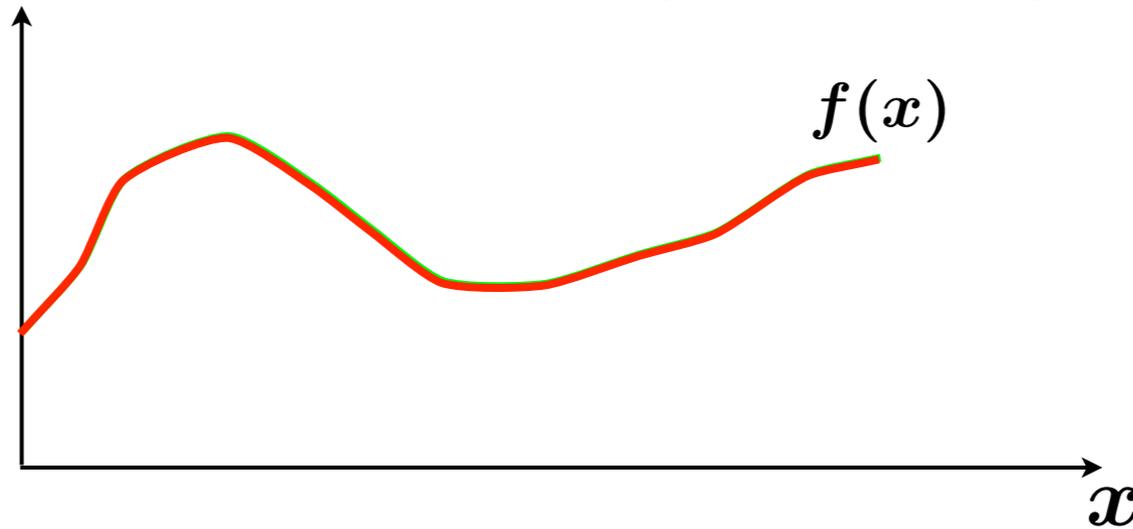
- ❑ Temme *et al.* [TOVPV11] propose a “rewinding” technique to revert to  $|k\rangle$ , based on a series of projective measurements.
- ❑ Use quantum rejection sampling! Equivalent to amplifying accepted moves, therefore avoiding having to revert moves at all.



# Boolean hidden shift

Setup:

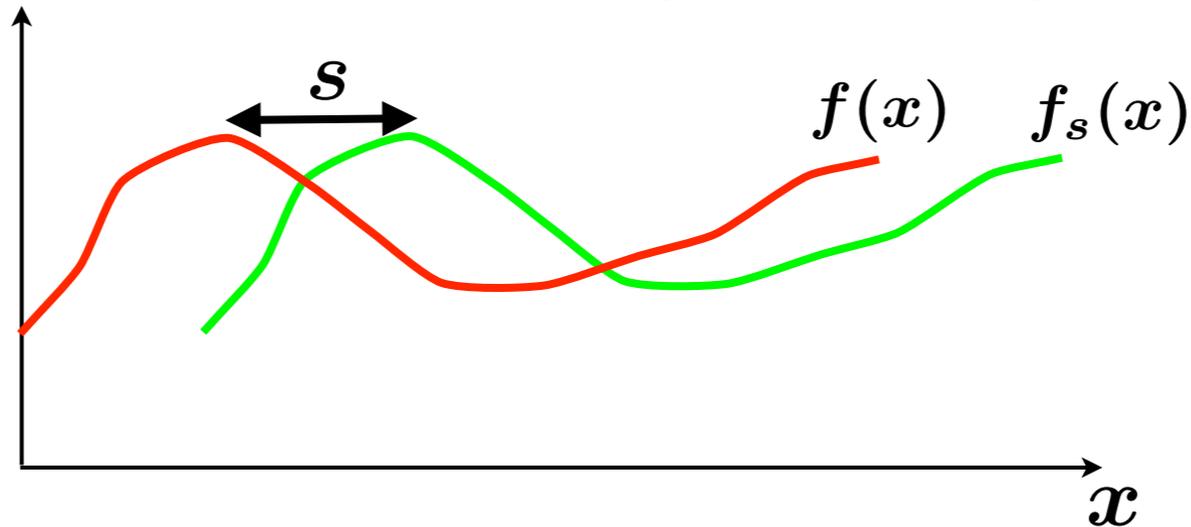
- $f(x)$ : (known) Boolean function
- $f_s(x) = f(x + s)$ , with an (unknown) shift  $s \in \{0, 1\}^n$



# Boolean hidden shift

Setup:

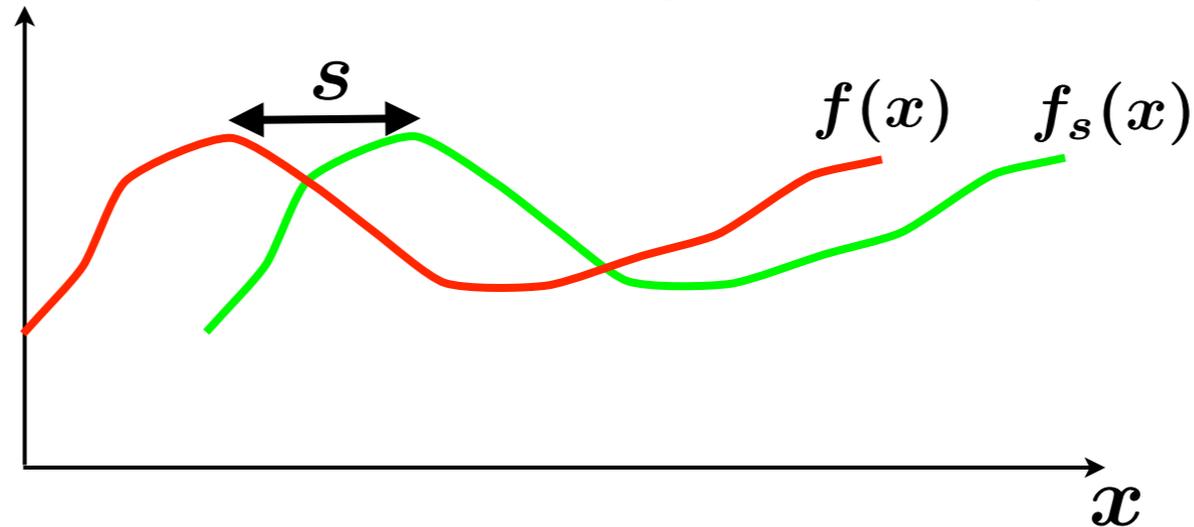
- $f(x)$ : (known) Boolean function
- $f_s(x) = f(x + s)$ , with an (unknown) shift  $s \in \{0, 1\}^n$



# Boolean hidden shift

Setup:

- $f(x)$ : (known) Boolean function
- $f_s(x) = f(x + s)$ , with an (unknown) shift  $s \in \{0, 1\}^n$

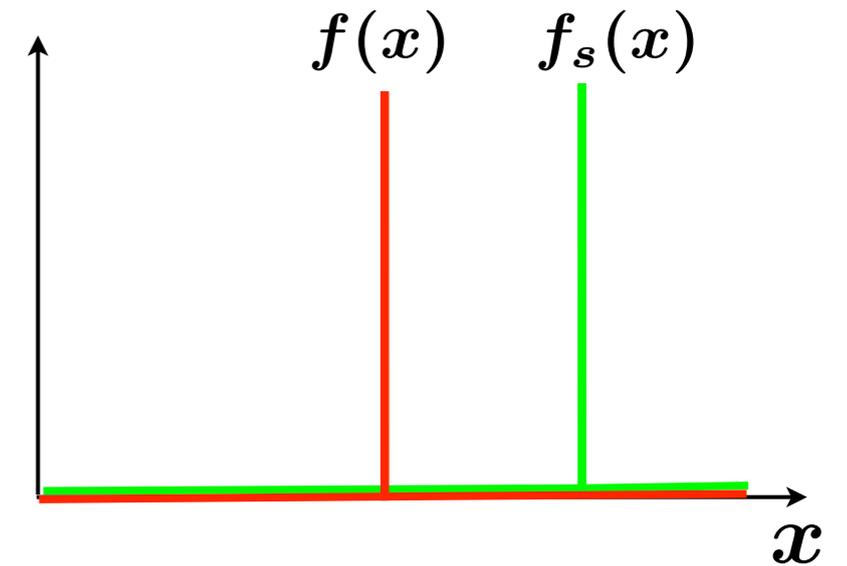


Boolean hidden shift problem

Given black-box access to  $f_s(x)$ ,  
find the hidden shift  $s$

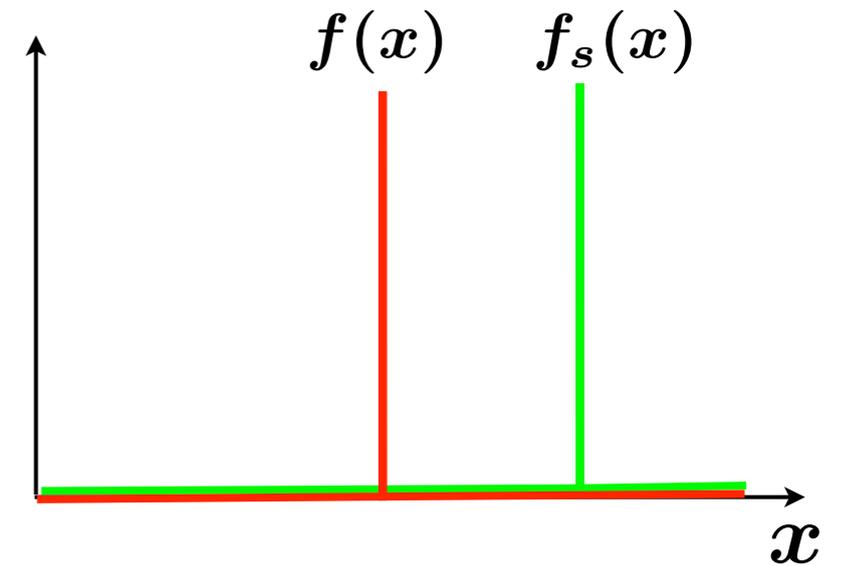
# Special cases

- Delta function  $f(x) = \delta_{xx_0}$
- = Grover's search problem
- Requires  $\Theta(\sqrt{2^n})$  queries [Grover96]



# Special cases

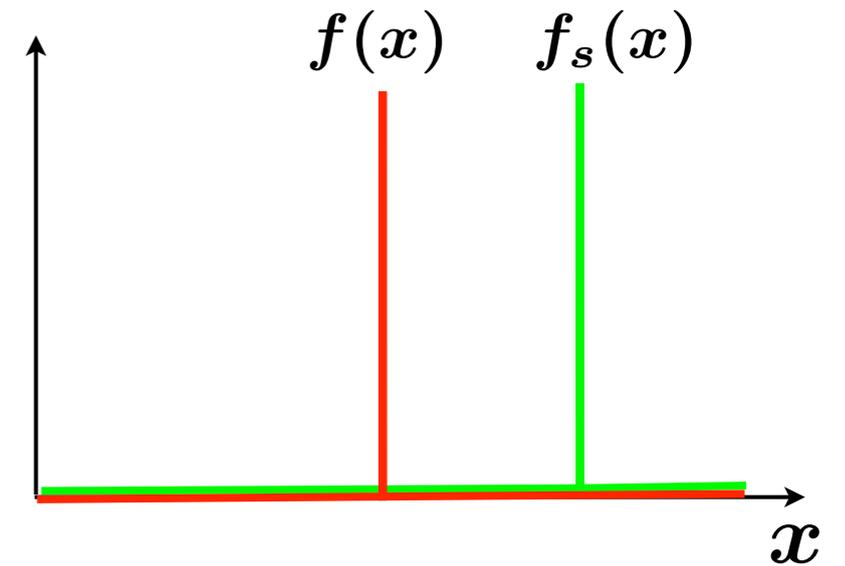
- Delta function  $f(x) = \delta_{xx_0}$ 
  - = Grover's search problem
  - Requires  $\Theta(\sqrt{2^n})$  queries [Grover96]



- Bent functions
  - = Functions with flat Fourier spectrum
  - Can be solved with 1 query! [Rötteler10]

# Special cases

- Delta function  $f(x) = \delta_{xx_0}$ 
  - = Grover's search problem
  - Requires  $\Theta(\sqrt{2^n})$  queries [Grover96]
- Bent functions
  - = Functions with flat Fourier spectrum
  - Can be solved with 1 query! [Rötteler10]
- What about other functions???

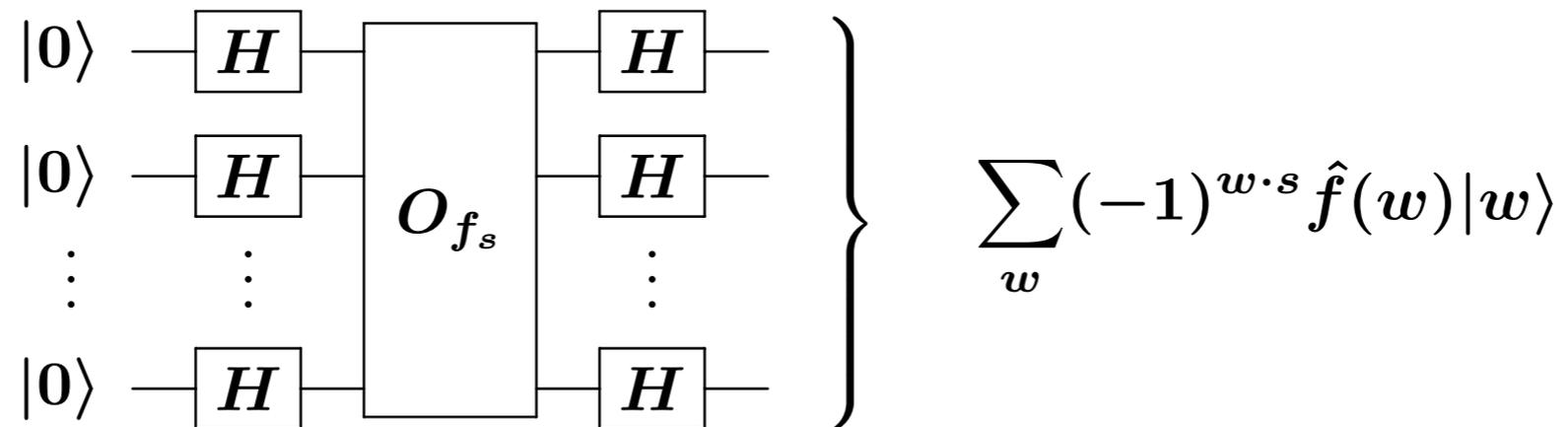


# New algorithm based on QRS

○ Use the following circuit, where

□  $H$  is the Hadamard transform

□  $O_{f_s}$  is the black box for  $f_s$ , acting as  $O_{f_s}|x\rangle = (-1)^{f_s(x)}|x\rangle$

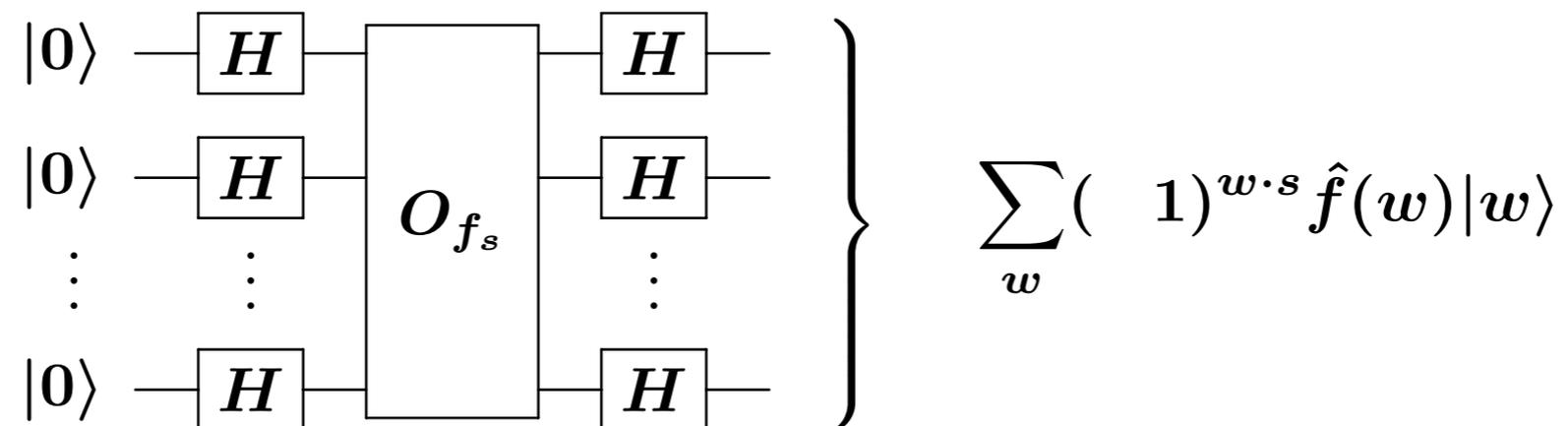


# New algorithm based on QRS

○ Use the following circuit, where

□  $H$  is the Hadamard transform

□  $O_{f_s}$  is the black box for  $f_s$ , acting as  $O_{f_s}|x\rangle = (-1)^{f_s(x)}|x\rangle$



○ Use QRS to produce the state  $\frac{1}{\sqrt{2^n}} \sum_w (-1)^{w \cdot s} |w\rangle$

□ Known amplitudes = Fourier coefficients  $\hat{f}(w)$

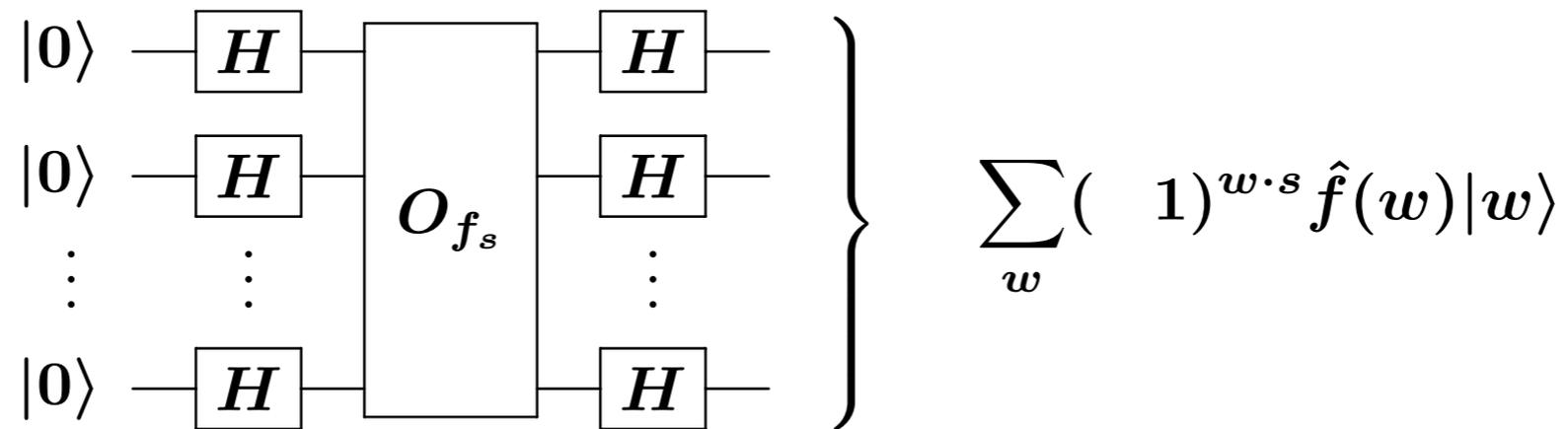
□ Unknown “states” = phases  $(-1)^{w \cdot s}$

# New algorithm based on QRS

○ Use the following circuit, where

□  $H$  is the Hadamard transform

□  $O_{f_s}$  is the black box for  $f_s$ , acting as  $O_{f_s}|x\rangle = (-1)^{f_s(x)}|x\rangle$



○ Use QRS to produce the state  $\frac{1}{\sqrt{2^n}} \sum_w (-1)^{w \cdot s} |w\rangle$

□ Known amplitudes = Fourier coefficients  $\hat{f}(w)$

□ Unknown “states” = phases  $(-1)^{w \cdot s}$

○ Use a final Fourier transform  $H^{\otimes n}$  to get  $|s\rangle$

# Wrap-up

- Rejection sampling has found many applications in classical computing

# Wrap-up

- Rejection sampling has found many applications in classical computing
- Quantum rejection sampling could be as useful for quantum computing!

# Wrap-up

- Rejection sampling has found many applications in classical computing
- Quantum rejection sampling could be as useful for quantum computing!
- Example: 3 diverse applications
  - Linear system of equations [HarrowHassidimLloyd09]
  - Quantum Metropolis algorithm
  - Boolean hidden shift problem

# Outlook

## ○ Other applications

### Amplifying QMA witnesses

[MarriottWatrous05,NagajWocjanZhang09]

### Preparing PEPS states [SchwarzTemmeVerstraeteII]

### ???

Support:



# Outlook

## ○ Other applications

### Amplifying QMA witnesses

[MarriottWatrous05,NagajWocjanZhang09]

### Preparing PEPS states [SchwarzTemmeVerstraeteII]

### ???

## ○ Adversary method for this extended model of quantum query complexity?

### Non-trivial error dependence

### Infinite-size adversary matrices

Support:

