Oracles with Costs

Shelby Kimmel, Cedric Yen-Yu Lin, Han-Hsuan Lin

Joint Center for Quantum Information and Computer Science, University of Maryland

Center for Theoretical Physics, MIT

TQC 2015, May 22, 2015

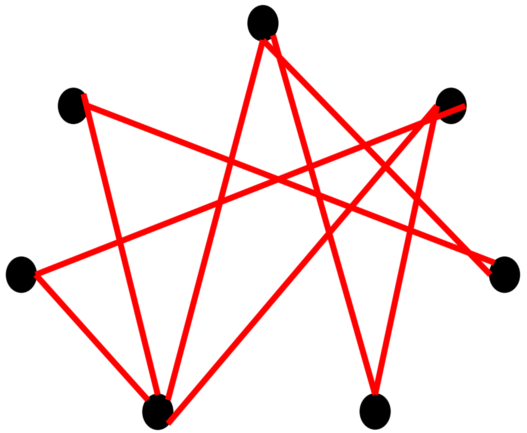
Arxiv: 1502.02174

JOINT CENTER FOR QUANTUM INFORMATION AND COMPUTER SCIENCE

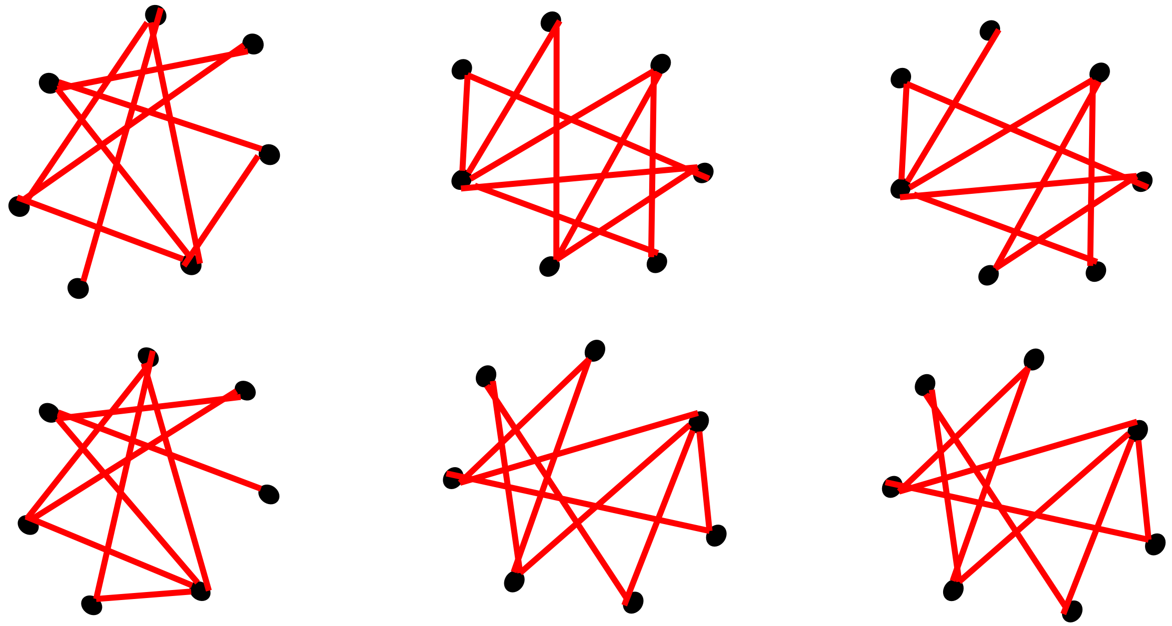
UMIACS
University of Maryland Institute for Advanced Computer Studies

Searching for an Isomorphic Graph

Is this target graph:

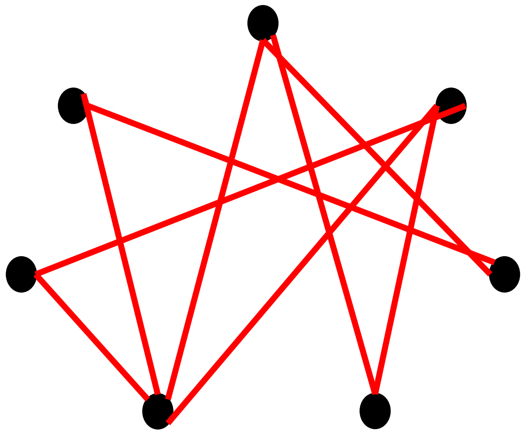


Isomorphic to any of these test graphs?

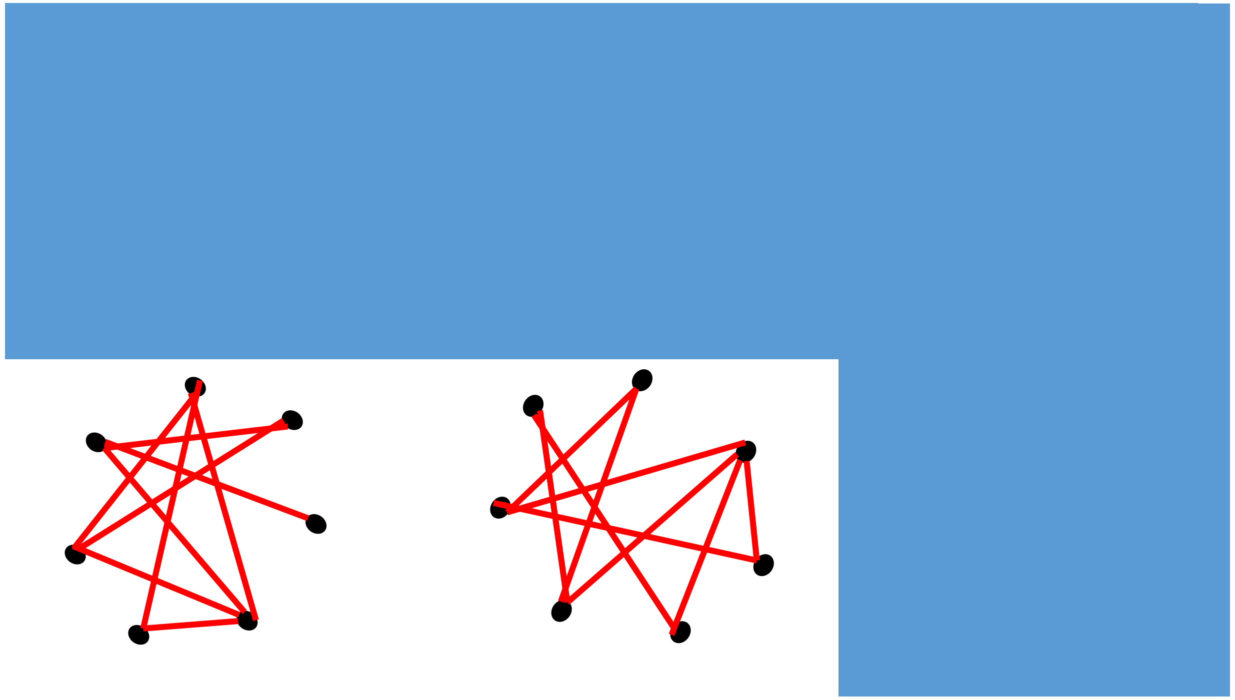


Searching for an Isomorphic Graph

Is this target graph:

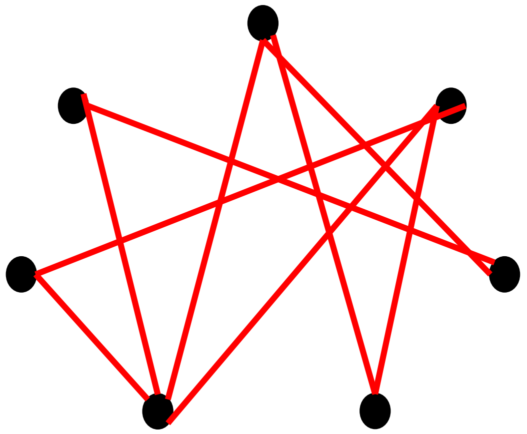


Isomorphic to any of these test graphs?

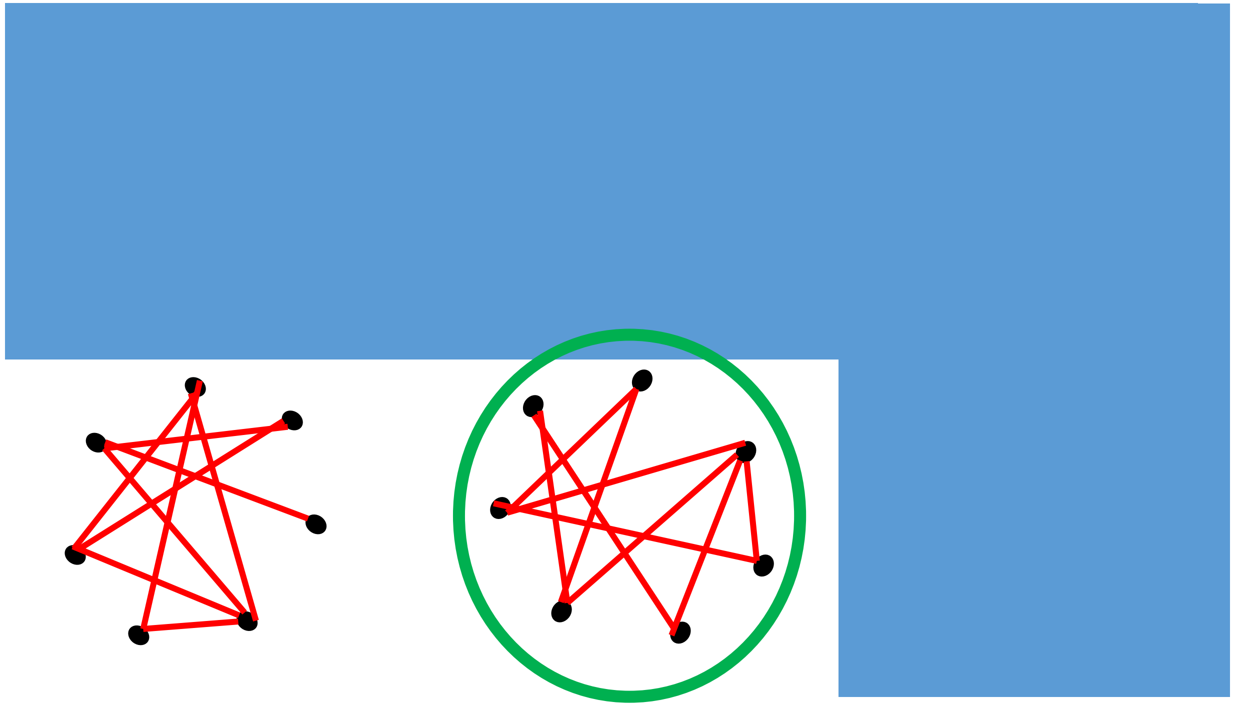


Searching for an Isomorphic Graph

Is this target graph:



Isomorphic to any of these test graphs?

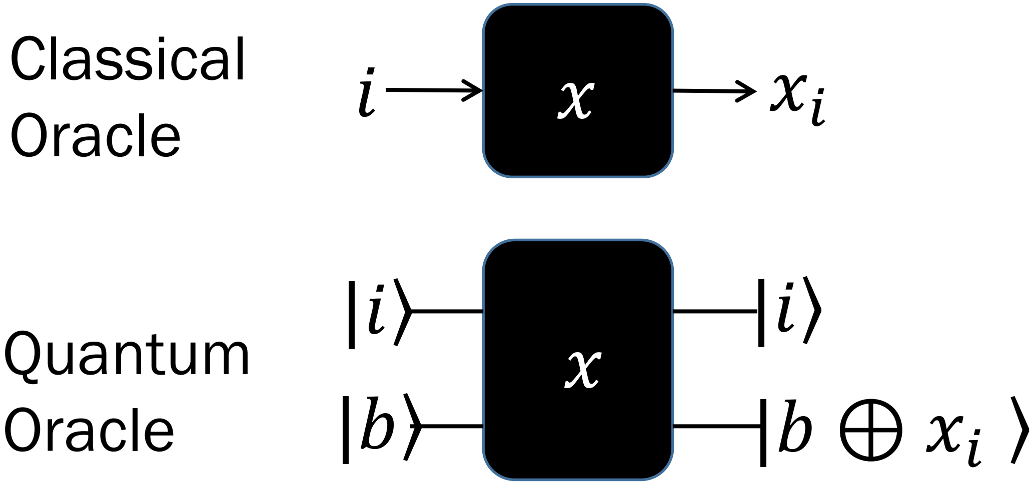


Outline

- Oracles and Oracle with Costs
- Related work
- Simple Example: Search with Two Oracles
- Open Problems

Standard Oracle Model

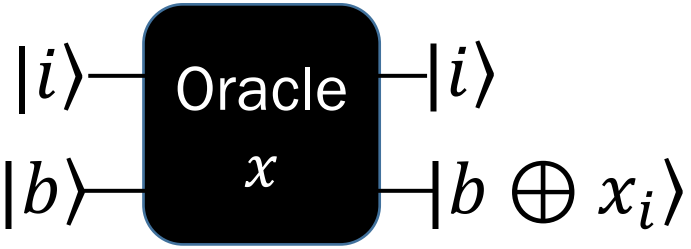
<u>Goal:</u> Evaluate a function $f(x)$ for Boolean input $x = \{x_1, x_2, \cdots, x_N\}$, given an oracle for x.



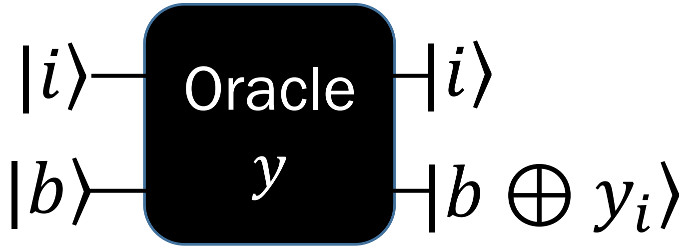
Want to minimize total uses of oracle (queries)

Oracles with Costs

Goal: Evaluate $f(x, y)$ for Boolean inputs $x = \{x_1, x_2, \cdots, x_N\}$, and $y = \{y_1, y_2, \cdots, y_N\}$ given a set of oracles for x and y



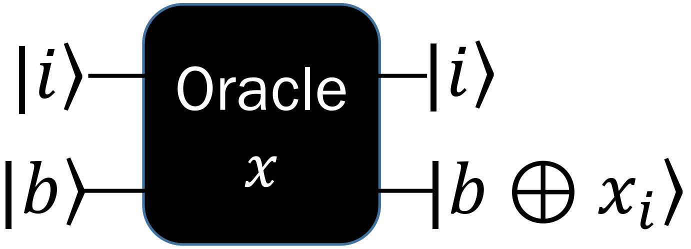
Cost c_x



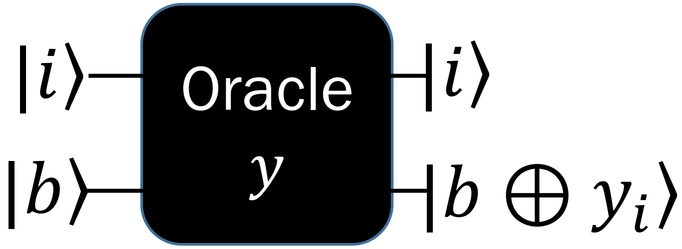
Cost c_y

Oracles with Costs

<u>Goal:</u> Evaluate $f(x, y)$ for Boolean inputs $x = \{x_1, x_2, \cdots, x_N\}$, and $y = \{y_1, y_2, \cdots, y_N\}$ given a set of oracles for x and y



Cost c_x



Cost c_y

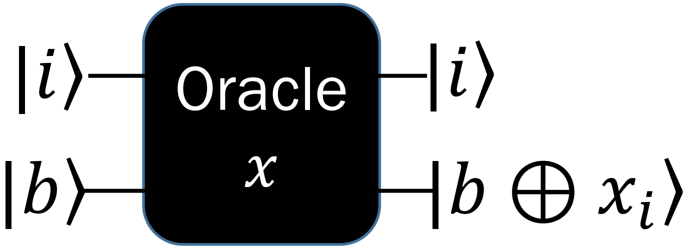
Want to minimize total cost

$$q_x c_x + q_y c_y,$$

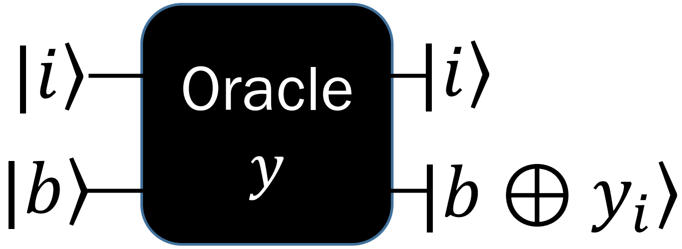
where q_i is the # of queries to Oracle i

Oracles with Costs

<u>Goal:</u> Evaluate $f(x, y)$ for Boolean inputs $x = \{x_1, x_2, \cdots, x_N\}$, and $y = \{y_1, y_2, \cdots, y_N\}$ given a set of oracles for x and y



Cost c_x



Cost c_y

Want to minimize total cost

$$q_x c_x + q_y c_y,$$

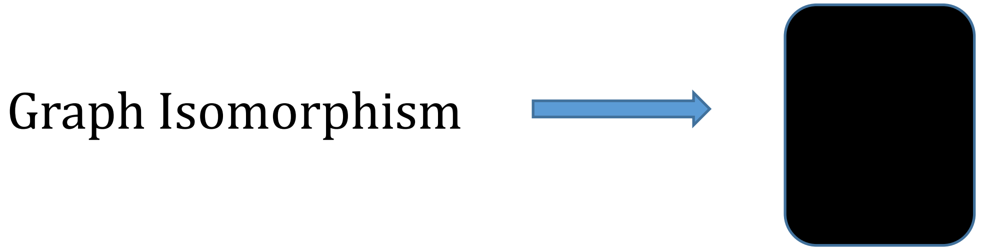
where q_i is the # of queries to Oracle i

3 Cases:

- Classical
- Oracles not allowed in superposition
- Oracles in superposition

Utility of Multiple Oracles Model

- We often have extra information (black box is not a good description).

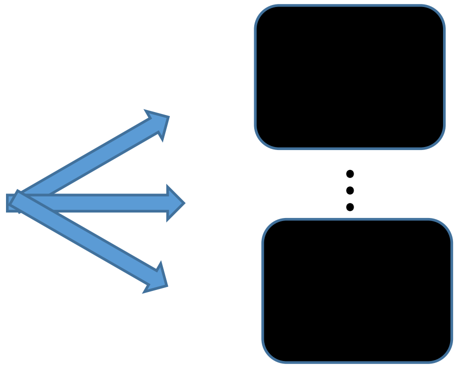


- In the real world, oracles take time to implement
- Can apply oracle tool box: algorithms, lower bounding techniques, etc

Utility of Multiple Oracles Model

- We often have extra information (black box is not a good description).

Graph Isomorphism



- In the real world, oracles take time to implement
- Can apply oracle tool box: algorithms, lower bounding techniques, etc

Related Work

- Ambainis '10: One oracle, querying different i requires different times
  - E.g. To query x_1 takes time 1, but to query x_2 takes time 2

- Montanaro '09: Searching with additional information.

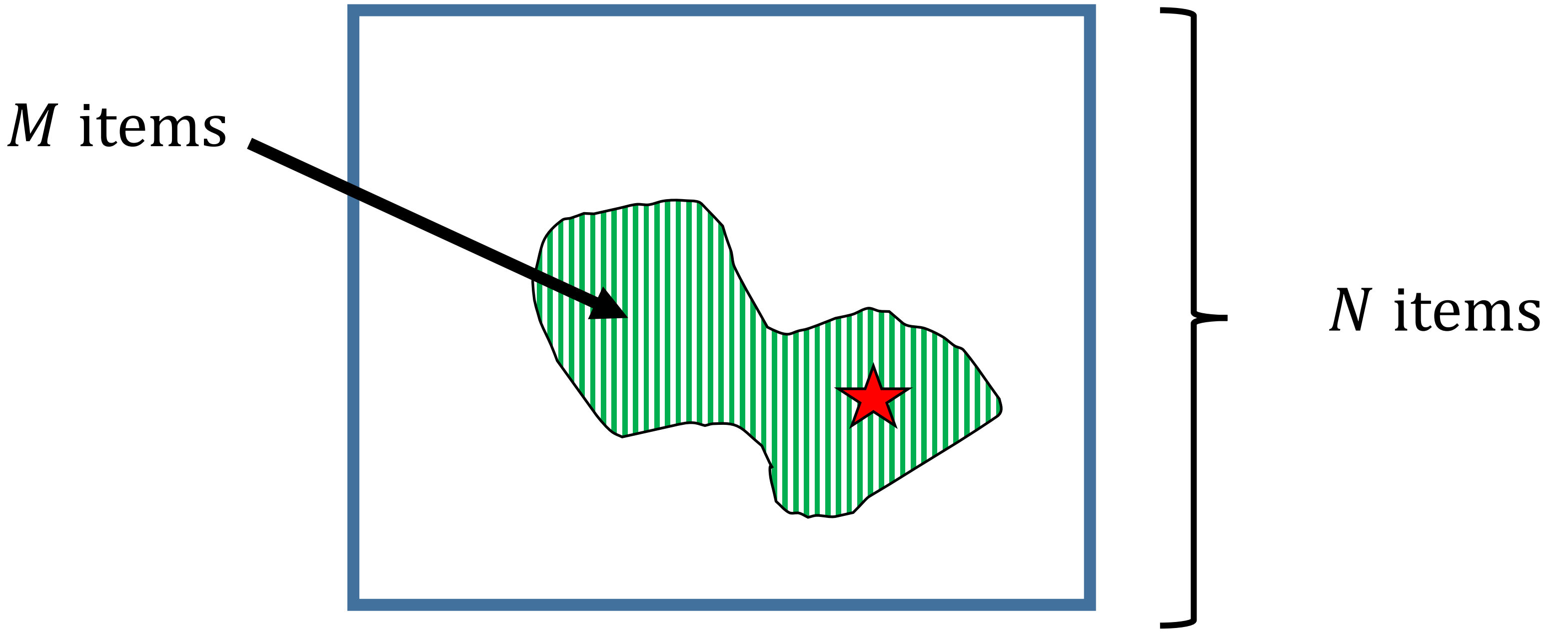
  E.g. Told that x_1=1 is more likely than x_2=1

- Cerf et al. '00: Use multiple tests to speed up evaluation of satisfiability problems.
  - No cost, No lower bounds, Need certain structure.

Searching with Two Oracles

Can ask ★ Oracle, "Is the i^{th} item starred?"
Can ask ◍ Oracle, "Is the i^{th} item striped?"

Promised: The starred item is also striped



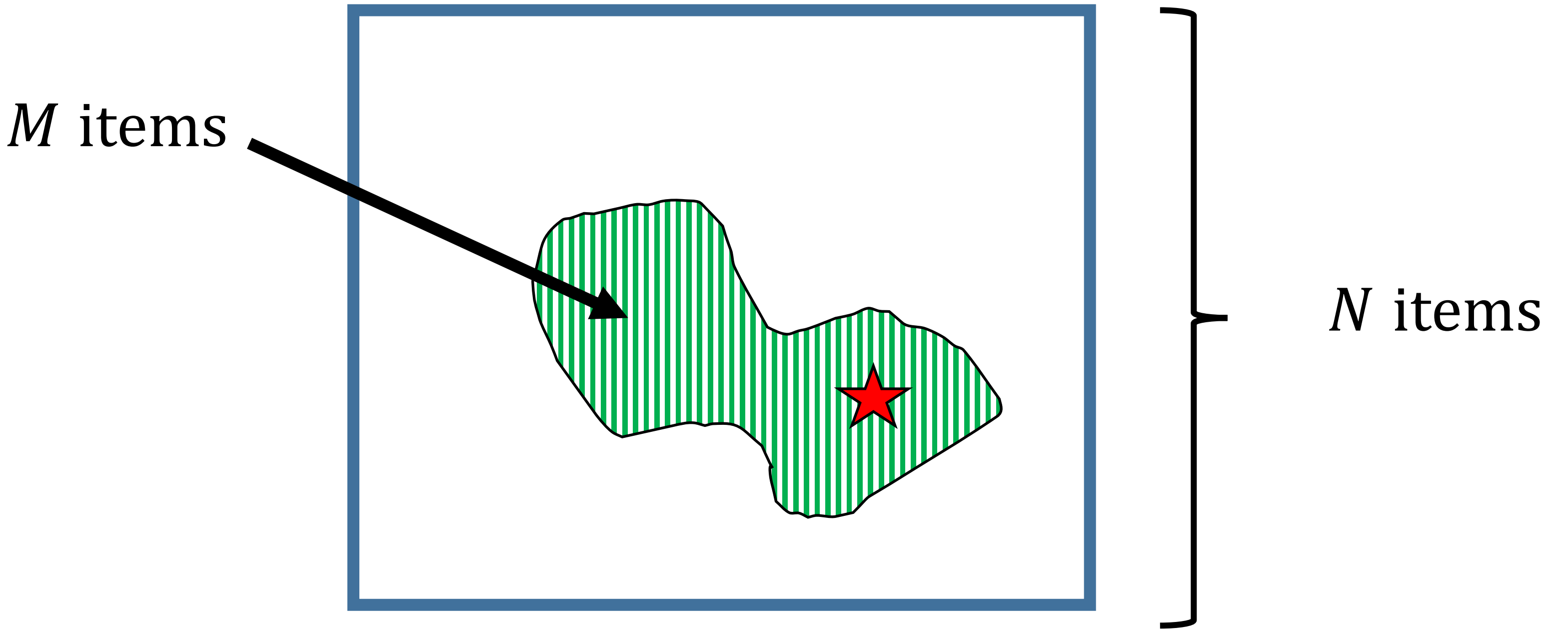
Searching with Two Oracles

Can ask ★ **Oracle**, “Is the i^{th} item starred?” **with cost $c_\star$**

Can ask **Oracle**, “Is the i^{th} item striped?” **with cost $c_\parallel$**

$c_\star > c_\parallel$

Promised: The starred item is also striped



Searching with Two Oracles

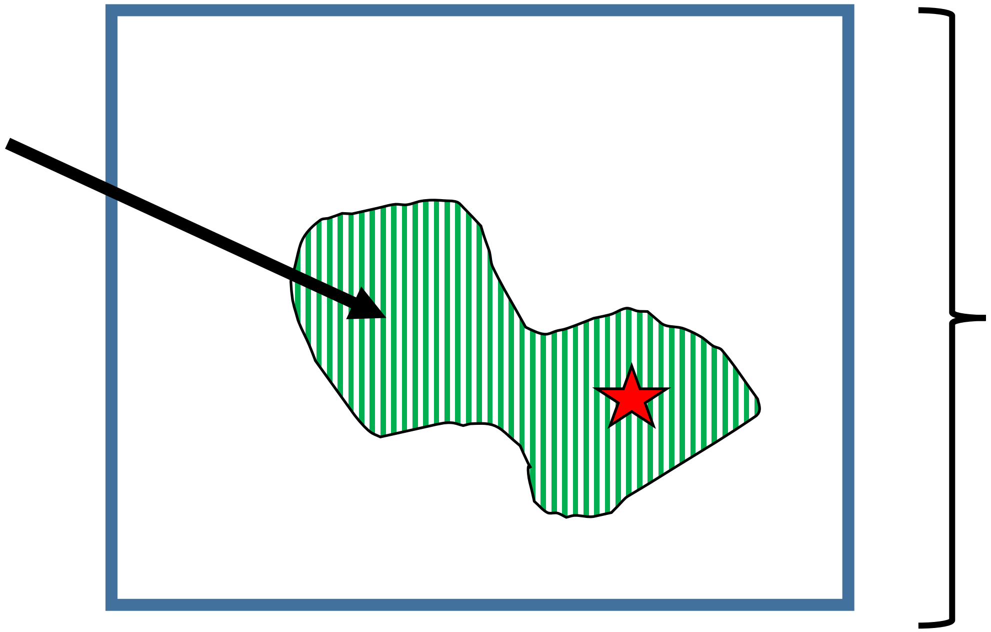
Can ask $\star$ **Oracle**, "Is the i^{th} item starred?" **with cost** $c_\star$

Can ask **Oracle**, "Is the i^{th} item striped?" **with cost** $c_\|$

$c_\star > c_\|$

Promised: The starred item is also striped

M items



N items

Can you find the starred item at lower cost using **Oracle**?

...

Searching with Two Oracles

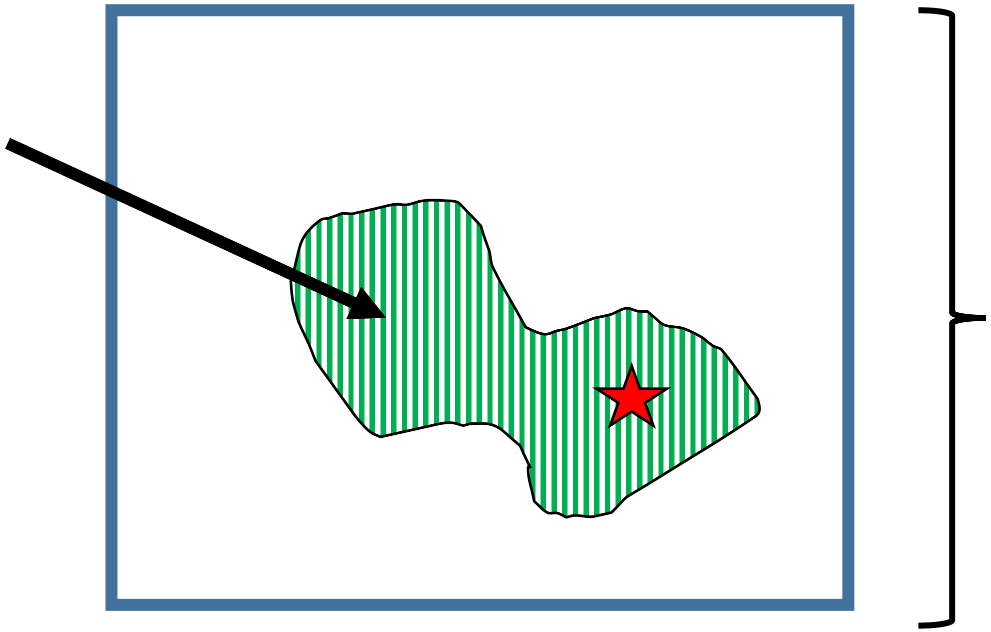
Can ask Oracle, “Is the i^{th} item starred?” **with cost** $c_\star$

Can ask Oracle, “Is the i^{th} item striped?” **with cost** $c_\parallel$

$c_\star > c_\parallel$

Promised: The starred item is also striped

M items



N items

Can you find the starred item at lower cost using Oracle?
… **YES**

Searching with Two Oracles

Can ask ★ **Oracle**, "Is the i^{th} item starred?" **with cost $c_\star$**
Can ask **Oracle**, "Is the i^{th} item striped?" **with cost $c_\parallel$**

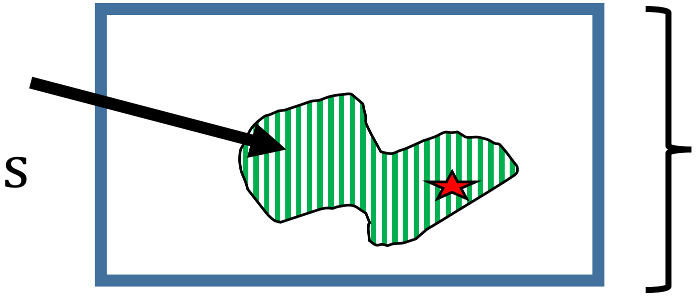
$c_\star > c_\parallel$

Promised: The starred item is also striped

Classical $= \theta(\min\{c_\star N, \quad c_\parallel N + c_\star M\})$

Sometimes best to check all N items using ★ **Oracle**

M items



N items

Searching with Two Oracles

Can ask ★ **Oracle**, "Is the i^{th} item starred?" **with cost $c_\star$**

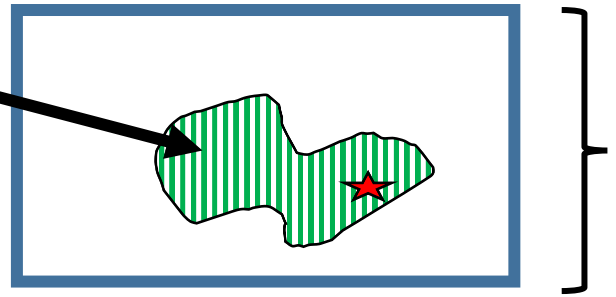
Can ask **Oracle**, "Is the i^{th} item striped?" **with cost $c_\parallel$**

$c_\star > c_\parallel$

Promised: The starred item is also striped

Classical $= \theta(\min\{c_\star N, \quad c_\parallel N + c_\star M\})$

M items



N items

Sometimes best to check all N items using ★ **Oracle**

Find all striped items by checking all N items using **Oracle**. Then use ★ **Oracle** to check all M striped items.

Searching with Two Oracles

Can ask ★ **Oracle**, "Is the i^{th} item starred?" **with cost $c_\star$**

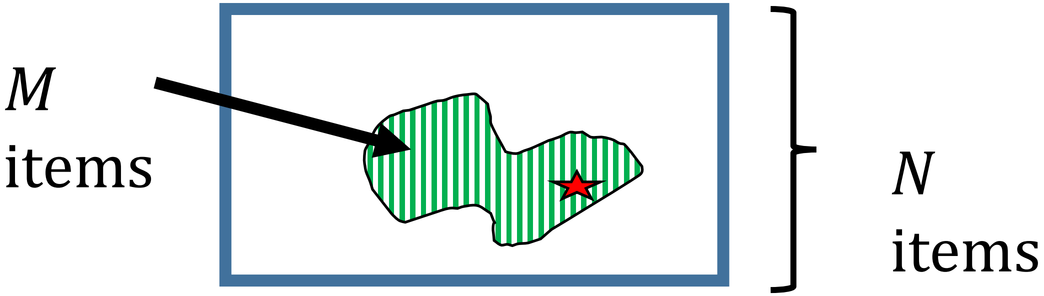
Can ask **Oracle**, "Is the i^{th} item striped?" **with cost $c_\parallel$**

$c_\star > c_\parallel$

Promised: The starred item is also striped

Classical $= \theta(\min\{c_\star N, \quad c_\parallel N + c_\star M\})$

Quantum $= \theta(\min\{c_\star \sqrt{N}, \ c_\parallel \sqrt{N} + c_\star \sqrt{M}\})$



Perform Grover search using ★ **Oracle**

Searching with Two Oracles

Can ask **Oracle**, "Is the i^{th} item starred?" **with cost** $c_\star$

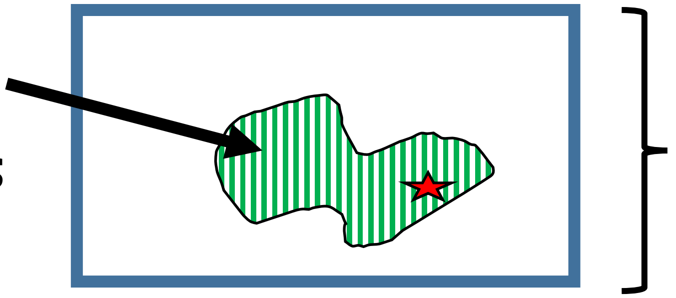
Can ask **Oracle**, "Is the i^{th} item striped?" **with cost** $c_\parallel$

$c_\star > c_\parallel$

Promised: The starred item is also striped

$$\text{Classical} = \theta(\min\{c_\star N,\quad c_\parallel N + c_\star M\})$$
$$\text{Quantum} = \theta\left(\min\left\{c_\star\sqrt{N},\ c_\parallel\sqrt{N} + c_\star\sqrt{M}\right\}\right)$$

M items



N items

Perform Grover search using **Oracle**

Do amplitude amplification

Searching with Two Oracles

Can ask **Oracle**, "Is the i^{th} item starred?" **with cost $c_\star$**

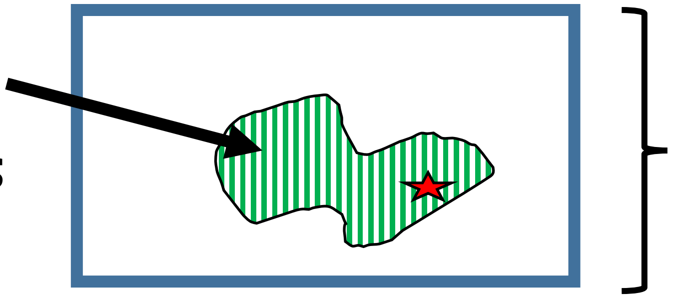
Can ask **Oracle**, "Is the i^{th} item striped?" **with cost $c_\parallel$**

$c_\star > c_\parallel$

Promised: The starred item is also striped

$$\text{Classical} = \theta(\min\{c_\star N,\quad c_\parallel N + c_\star M\})$$
$$\text{Quantum} = \theta\left(\min\left\{c_\star\sqrt{N},\ c_\parallel \sqrt{N} + c_\star\sqrt{M}\right\}\right)$$

M items



N items

Observations:

- Bounds are same whether oracles used in superposition or not.
- Amplitude amplification is optimal

Lower Bounds for Search with Two Oracles

Almost any technique works to give asymptotically tight lower bound:

- Reduction to search
- Adversary method
- Variable times lower bounding method of Ambainis.

Lower Bounds for Search with Two Oracles

Almost any technique works to give asymptotically tight lower bound:

- Reduction to search
- Adversary method
- Variable times lower bounding method of Ambainis.

Is it possible to prove exact optimality?

- Grover's algorithm is exactly optimal.[Zalka '99]
- We use amplitude amplification, a variant of Grover's algorithm.

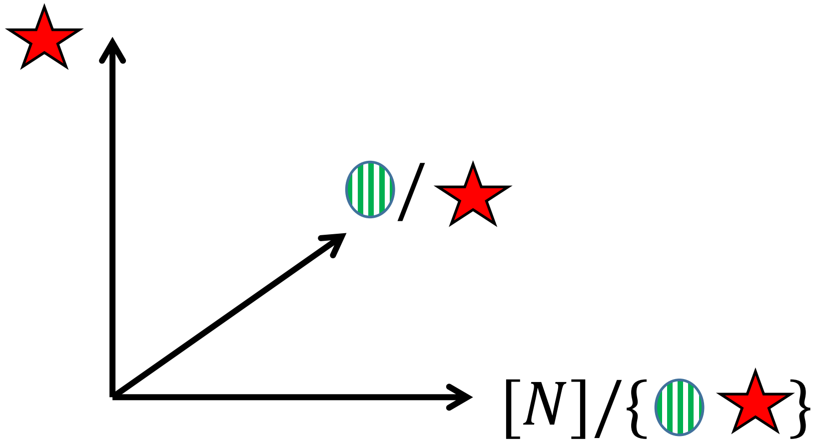
Exact Bounds for Search with Two Oracles

Special case:

- Start in equal superposition state:

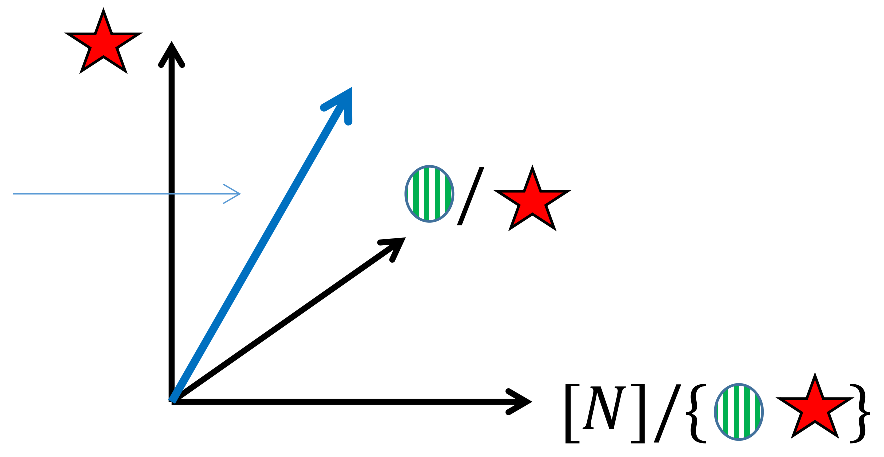
$$\frac{1}{\sqrt{N}}\sum_{i=1}^{N}|i\rangle$$

- Can only apply oracles and G: reflection about equal superposition state



Exact Bounds for Search with Two Oracles

Label position of state of system at any point in the algorithm using (shifted) polar coordinates: θ, ϕ

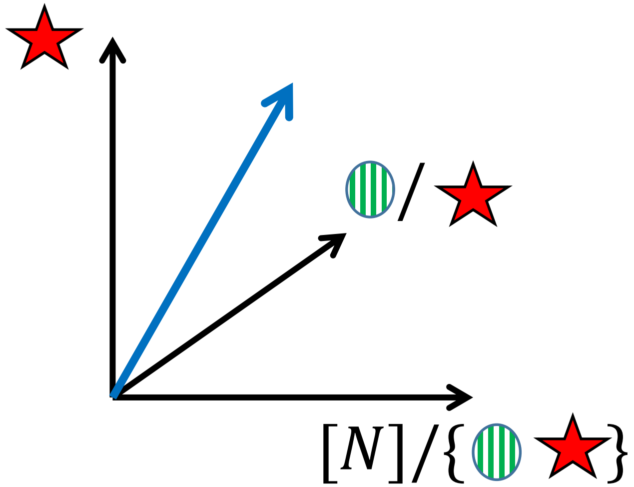


$$H(\theta,\phi) = \theta - k(N, M, \boldsymbol{c}_*, \boldsymbol{c}_{||}) \times \min_{l\in\mathbb{Z}} |\phi + 2l\pi - \pi/2|$$

Initially, progress function is close 0, at end should be close to $\frac{\pi}{2}$.

- G does not change progress function
- Oracles can increase progress function

Exact Bounds for Search with Two Oracles



Algorithm that succeeds with probability at least $1-\epsilon$:*

Cost $\geq\ c_{||}\sqrt{N}\arcsin(\sqrt{1-\epsilon})\sec(\phi_0+\sqrt{M/N})/2$

Where $\phi_0 = \max\begin{cases} 0 \\ \phi : \tan(\phi+\sqrt{M/N}) = \phi + c_*/c_{||}\sqrt{M/N} \end{cases}$

Same as optimal amplitude amplification algorithm!

$*$ In the limit of $C(c_{||}, c_*, M, N) = \frac{c_{||}\sqrt{N}}{c_*\sqrt{\epsilon}2Mcos(\phi_0+\sqrt{M/N})} \to 0$

Directions for Future Work

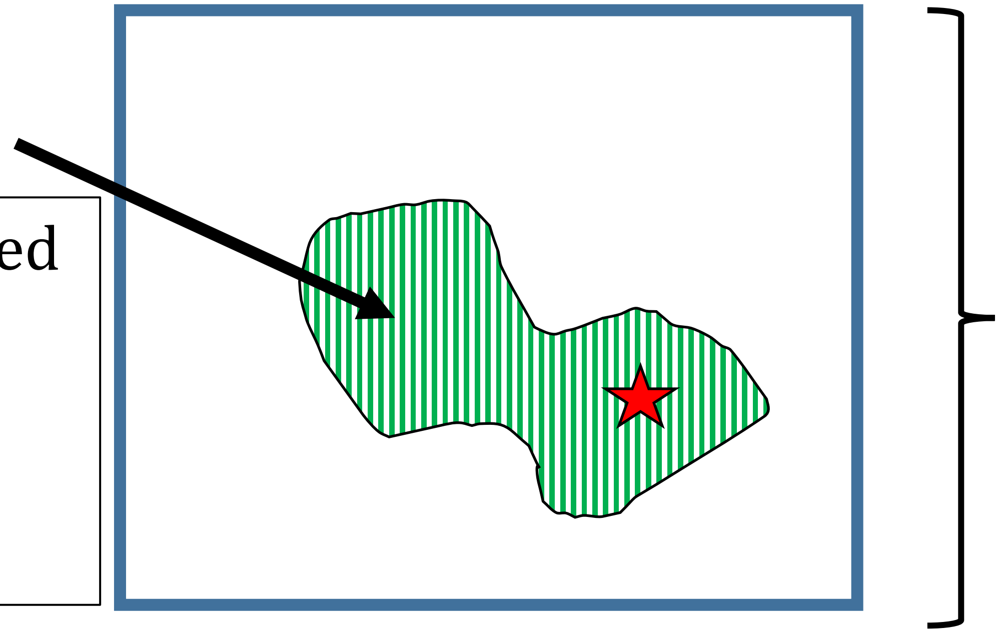
- Create exactly tight bounds for searching with two oracles?
- Prove asymptotic optimality for analogous problem with $\log N$ multiply nested oracles?
- Create a general framework for understanding oracles with costs, like the general adversary bound?
- Are there other problems (besides search) where introducing additional oracles makes sense?

Arxiv: 1502.02174

Searching with Two Oracles

Can ask Oracle, “Is the i^{th} item starred?”
Can ask Oracle, “Is the i^{th} item striped?”

Promised: The starred item is also striped



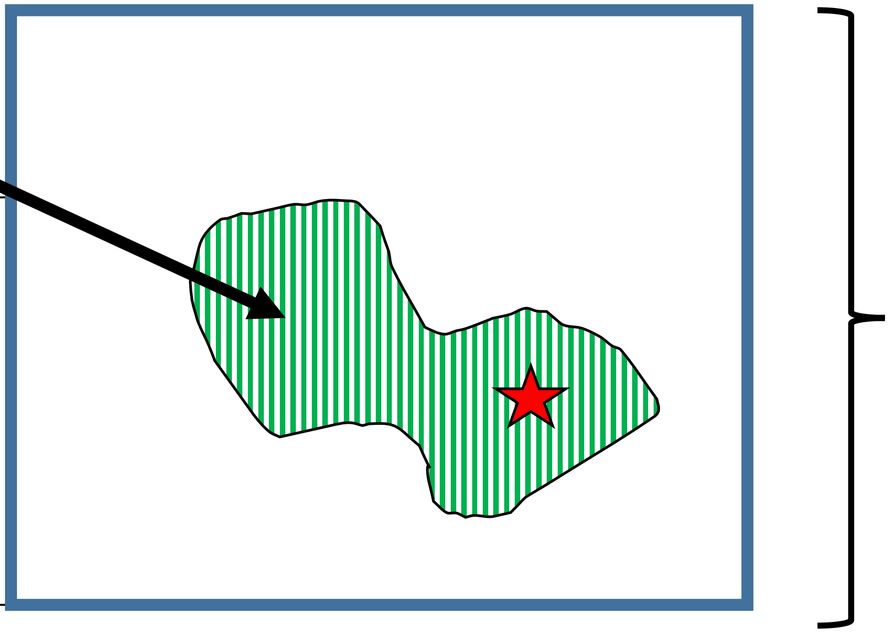
Can you find the starred item with fewer total oracle queries using Oracle?

…

Searching with Two Oracles

Can ask Oracle, “Is the i^{th} item starred?”
Can ask Oracle, “Is the i^{th} item striped?”

Promised: The starred item is also striped



Can you find the starred item with fewer total oracle queries using Oracle?
… **NO**