The Collision Problem

Andreas Klappenecker

Texas A&M University

The Collision Problem

Suppose that $f\colon \{1, 2, \ldots, n\} \to S$ is a black-box function that is promised to be either one-to-one or two-to-one.

Goal: Determine whether f is one-to-one or two-to-one.

In other words, we would like to know whether there exists a pair of distinct numbers x and y in $\{1, 2, \ldots, n\}$ such that $f(x) = f(y)$.

1. The codomain S of the function f must have at least n elements, since there cannot exist one-to-one functions on $\{1, 2, \ldots, n\}$ when S has fewer than n elements.
2. The number of elements n must be even, since there cannot exist two-to-one functions when n is odd.

Example

86, 83, 43, 78, 38, 89, 92, 70, 99, 96, 77, 28, 11, 71, 29, 85, 65, 41, 99, 67, 28, 61, 55, 62, 50, 56, 41, 71, 76, 34, 75, 94, 21, 13, 70, 67, 31, 44, 81, 21, 53, 22, 98, 93, 93, 52, 57, 31, 82, 44, 36, 27, 17, 59, 58, 23, 56, 59, 40, 18, 68, 39, 96, 55, 10, 12, 66, 40, 72, 90, 30, 52, 60, 17, 54, 91, 73, 15, 51, 90, 24, 42, 14, 33, 84, 69, 81, 34, 50, 79, 32, 94, 57, 63, 48, 80, 25, 11, 36, 14, 47, 68, 73, 85, 15, 79, 66, 82, 54, 95, 88, 12, 61, 38, 88, 87, 77, 89, 62, 23, 72, 18, 46, 20, 37, 10, 60, 58, 48, 45, 30, 16, 97, 75, 64, 92, 63, 32, 20, 47, 13, 53, 86, 45, 84, 51, 97, 65, 35, 49, 25, 80, 64, 26, 22, 76, 46, 27, 29, 39, 69, 16, 19, 33, 87, 74, 19, 74, 37, 43, 91, 78, 49, 42, 35, 83, 26, 95, 98, 24

Why do we Care?

Given two graphs G and H, $V(G) = V(H) = \{1, 2, \ldots, n\}$.
Let S_n be the symmetric group of all $n!$ permutations of n elements.

Consider the set $\{\pi(G) \mid \pi \in S_n\} \cup \{\pi(H) \mid \pi \in S_n\}$. If G and H are rigid graphs, meaning that they are automorphism-free, then we get a collision if and only if the graphs G and H are isomorphic.

In fact, the function f from $S_n \times \{G, H\}$ into the set of permutations of the graphs of G and H given by

$$f(\pi, G) = \pi(G) \quad \text{and} \quad f(\pi, H) = \pi(H)$$

is two-to-one when G and H are rigid graphs, since

$$G = \pi(H) \quad \text{implies} \quad \pi'(G) = \pi'(\pi(H)) \quad \text{for all } \pi' \in S_n.$$

How Hard is the Problem? (Deterministically)

In the worst case, we need to select at least $\left\lceil \frac{n+1}{2} \right\rceil$ elements to decide with certainty whether f is one-to-one or two-to-one.

[Clearly, $n/2$ elements might not suffice, as the values could be all different, but $n/2 + 1 = \left\lceil \frac{n+1}{2} \right\rceil$ elements are enough by the pigeonhole principle.]

Thus, any deterministic algorithm needs to evaluate $\Omega(n)$ samples to solve the problem with certainty.

How Hard is the Problem? (Randomized)

Suppose that f is a two-to-one function on the domain $\{1, 2, \ldots, n\}$, where $n = 2\ell$. Then m samples will yield a collision with probability $1 - \delta$ or higher as long as $m \geqslant \sqrt{n \ln \frac{1}{\delta} + \frac{1}{4}} + \frac{1}{2}$.

Since f has ℓ different values, the probability that all m samples yield a different value is at most (recall that $1 - x \leqslant e^{-x}$)

$$\left(1 - \frac{1}{\ell}\right)\left(1 - \frac{2}{\ell}\right) \cdots \left(1 - \frac{m-1}{\ell}\right) \leqslant \prod_{k=1}^{m-1} \exp\left(-\frac{k}{\ell}\right) = \exp\left(-\frac{m^2 - m}{2\ell}\right) \leqslant \delta$$

We can determine with high probability whether f is a one-to-one or two-to-one function by evaluating the black box $\Theta(\sqrt{n})$ times using a randomized sampling algorithm (essentially Birthday paradox).

Comparison

We can determine with high probability whether f is a one-to-one or two-to-one function by evaluating the black box $\Theta(\sqrt{n})$ times using a randomized sampling algorithm (essentially Birthday paradox).

We can choose an argument x and use the black box to obtain the value $v = f(x)$. Then use $\Theta(\sqrt{n})$ calls to the black box function to find y s.t. $v = f(y)$ using Grover's algorithm.

Comparison

We can determine with high probability whether f is a one-to-one or two-to-one function by evaluating the black box $\Theta(\sqrt{n})$ times using a randomized sampling algorithm (essentially Birthday paradox).

We can choose an argument x and use the black box to obtain the value $v = f(x)$. Then use $\Theta(\sqrt{n})$ calls to the black box function to find y s.t. $v = f(y)$ using Grover's algorithm.

Both algorithms use the same order of magnitude of calls to the black box function. The space requirement for the quantum algorithm is less, but otherwise there is no benefit!

General Idea

Why don't we combine the randomized sampling with Grover search?

The BHT Algorithm

Given: A black box function $f\colon \{1, 2, \ldots, n\} \to S$ that is promised to be one-to-one or two-to-one.

1. Select a subset T of the domain of f of cardinality $m = n^{1/3}$.
2. Evaluate $f(x)$ for all x in T and check for collisions. This requires $\Theta(n^{1/3})$ black box evaluations. If there are collisions, return "two-to-one", otherwise go to the next step.
3. Select an element t in T. No element s in T satisfies $f(s) = f(t)$, since T is collision-free.
4. Use Grover to find an element s in T^c such that $f(s) = f(t)$ for some $t \in T$. If no element is found, then return "one-to-one", otherwise "two-to-one".

The Key Point of the BHT Algorithm

A Grover search on N elements with m solutions requires an expected $\Theta(\sqrt{N/m})$ black-box queries.

The Key Point of the BHT Algorithm

A Grover search on N elements with m solutions requires an expected $\Theta(\sqrt{N/m})$ black-box queries.

The search function of the collision algorithm is given by

$$g(s) = \exists t \in T : f(s) = f(t).$$

The Key Point of the BHT Algorithm

A Grover search on N elements with m solutions requires an expected $\Theta(\sqrt{N/m})$ black-box queries.

The search function of the collision algorithm is given by

$$g(s) = \exists t \in T : f(s) = f(t).$$

Overall, the expected number of black-box calls is given by

$$m + \Theta(\sqrt{(n-m)/m}) = n^{1/3} + \Theta(n^{1/3}) = \Theta(n^{1/3}),$$

when $m = n^{1/3}$.

Lower Bounds

Aaronson

Any quantum algorithm for the collision problem needs $\Omega(n^{1/5})$ queries.

Aaronson

Any quantum algorithm for the collision problem needs $\Omega(n^{1/5})$ queries.

Shi

Any quantum algorithm for the collision problem needs $\Omega(n^{1/3})$ queries.