How to improve the price-performance ratio of quantum collision search D. J. Bernstein University of Illinois at Chicago

NSF ITR-0716498

Warning: Complexity estimates in this talk are approximate; small factors are suppressed. What is the fastest algorithm that, given s, finds collision in $x \mapsto MD5(s, x)$?

i.e. finds (x, x') with $x \neq x'$ and MD5(s, x) = MD5(s, x')?

Now have a very fast algorithm, leading to many attacks. MD5 is thoroughly broken. What is the fastest algorithm that, given s, finds collision in $x \mapsto MD5(s, x)$?

i.e. finds (x, x') with $x \neq x'$ and MD5(s, x) = MD5(s, x')?

Now have a very fast algorithm, leading to many attacks. MD5 is thoroughly broken.

Surprised by the collisions? Fact: By 1996, a few years after the introduction of MD5, Preneel, Dobbertin, et al. were calling for MD5 to be scrapped. What is the fastest algorithm that, given s, finds collision in $x \mapsto SHA-256(s, x)$?

SHA-256 is an NSA design. Seems much better than MD5, but confidence isn't high.

Ongoing SHA-3 competition will lead to much higher public confidence in SHA-3.

But should SHA-3 produce 256-bit output? 512-bit output? How do quantum computers affect the answer?

Guessing a collision

For *any* classical circuit *H* producing *b*-bit output:

Generate random (b+1)-bit strings x, x'. Chance $\geq 1/2^{b+1}$ that (x, x') is a collision in H, i.e., $x \neq x'$ and H(x) = H(x'). Otherwise try again.

Good chance of success within 2^b evaluations of H.

1996 Grover, 1997 Grover:

Take classical circuit *F* using *f* bit operations to produce 1-bit output from *b*-bit input.

Explicit construction of quantum circuit G(F)using $2^{b/2}f$ qubit operations to compute a root of Fwith high probability if F has a unique root.

1996 Boyer–Brassard–Høyer– Tapp, generalizing Grover: $2^{(b-u)/2}f$ qubit operations to find some root of F with high probability if there are $\approx 2^u$ roots. Can easily use for collisions: Given classical circuit Husing h bit operations, define F(x, x') as 0

iff (x, x') is a collision in H.

Obtain some collision with high probability using $2^{b/2}h$ qubit operations.

Table lookups

Another classical approach:

Generate many random inputs x_1, x_2, \ldots, x_M ; e.g. $M = 2^{b/2}$.

Compute and sort M pairs $(H(x_1), x_1), (H(x_2), x_2), \ldots,$ $(H(x_M), x_M)$ in lex order.

Generate many random inputs y_1, y_2, \ldots, y_N ; e.g. $N = 2^{b/2}$. After generating y_j , check for $H(y_j)$ in sorted list. Same effect as searching all MN pairs (x_i, y_j) .

For
$$M = N = 2^{b/2}$$
,

good chance of success. Only $2^{b/2}$ evaluations of H.

Define F(y) as 0 iff there is a collision among $(x_1, y), (x_2, y), \dots, (x_M, y).$ This algorithm is finding root of F by classical search.

1998 Brassard–Høyer–Tapp: Instead use quantum search; e.g., $2^{b/3}h$ qubit operations if $M = 2^{b/3}$. 2003 Grover-Rudolph,

"How significant are the known collision and element distinctness quantum algorithms?":

Brassard–Høyer–Tapp algorithm uses $\approx 2^{b/3}$ qubits!

With such a huge machine, can simply run $2^{b/3}$ parallel quantum searches for collisions (x, x').

High probability of success within time $2^{b/3}h$.

What if our quantum circuit has only $2^{b/5}$ qubits?

Again Grover–Rudolph, mindless parallelism: high probability of success within time $2^{2b/5}h$.

Grover–Rudolph advantage: no need for communication across the parallel searches.

Brassard–Høyer–Tapp needs huge RAM lookups using quantum indices. How expensive is this? Realistic model of computation developed thirty years ago:

A circuit is a 2-dimensional mesh of small parallel gates. Have fast communication between neighboring gates. Try to optimize time *T* as function of area *A*.

See, e.g., 1981 Brent–Kung for definition of model and proof that optimal circuits for length-N convolution have A = N and $T = N^{1/2}$. Can model *quantum* circuits in the same way to understand speedups from parallelism, slowdowns from communication.

Have a 2-dimensional mesh of small parallel quantum gates. Try to optimize time *T* as function of area *A*.

(Warning: Model is optimistic about quantum computation. Assumes that quantum-computer scalability problems are solved without poly slowdowns.) e.g. area $2^{b/5}$:

Have $2^{b/10} \times 2^{b/10}$ mesh of small quantum gates all operating in parallel.

Size- $2^{b/5}$ table lookup using quantum index can be handled in time $2^{b/10}$.

Brassard–Høyer–Tapp takes total time $2^{b/2}$. Grover–Rudolph is faster (despite having more "queries"): total time $2^{2b/5}$.

Parallel tables

Generate x_1, x_2, \ldots, x_M . Compute $H(x_1), H(x_2), \ldots, H(x_M)$.

Generate $y_1, y_2, ..., y_M$. Compute $H(y_1), H(y_2), ..., H(y_M)$.

Sort all hash outputs to easily find collisions. Repeat $2^b/M^2$ times; high probability of success. Mesh-sorting algorithms (e.g., 1987 Schimmler) sort these hash outputs in time $M^{1/2}$ on classical circuit of area M.

Computation of hash outputs takes time h; negligible if M is large. Total time $2^b/M^{3/2}$.

e.g. area $2^{b/5}$, time $2^{7b/10}$.

Now Grover-ize this algorithm. Define $F(x_1, \ldots, x_M, y_1, \ldots, y_M)$ as 0 iff some (x_i, y_j) is a collision in H. Original algorithm used mesh-sorting circuit for Fof size M taking time $M^{1/2}$. Convert circuit into quantum mesh-sorting circuit of size M taking time $M^{1/2}$.

Find root of F using $2^{b/2}/M$ evaluations of F on quantum superpositions. Total time $2^{b/2}/M^{1/2}$.

e.g. area $2^{b/5}$, time $2^{2b/5}$.

Would beat Grover–Rudolph in a three-dimensional model of parallel quantum computation, or in a naive parallel model without communication delays. Faster; maybe optimal?

Do better by *iterating* H.

Choose a (b + 1)-bit string x_0 . Compute *b*-bit string $H(x_0)$; (b + 1)-bit string $x_1 = \pi(H(x_0))$ where π is a padding function; *b*-bit string $H(x_1)$; (b + 1)-bit string $x_2 = \pi(H(x_1))$; *b*-bit string $H(x_2)$; etc.

Proving time estimates here needs good π randomization, but experiments show simple π working for every interesting H. After $2^{b/2}$ steps, expect to find a "distinguished point": a string x_i whose first b/2 bits are all 0.

Choose another string y_0 , iterate in the same way until a distinguished point.

$$2^b$$
 pairs (x_i, y_j) ,

so expect some collision.

If there *is* a collision then the distinguished points are the same. Seeing this quickly reveals the collision. More generally, redefine "distinguished point" as having $b/2 - \lceil \lg M \rceil$ bits 0.

Build *M* parallel iterating units from *M* different strings. Expect time $2^{b/2}/M$ to find *M* distinguished points.

Good chance of collision. Easily find collision by sorting distinguished points. Summary:

area M, conj. time $2^{b/2}/M$. e.g. area $2^{b/5}$, conj. time $2^{3b/10}$.

Analogous quantum circuit: area M, conj. time $2^{b/2}/M$. e.g. area $2^{b/5}$, conj. time $2^{3b/10}$. Quantum-search speedup matches iteration speedup!

Compare to Grover–Rudolph: area $2^{b/5}$, time $2^{2b/5}$.

Or Brassard–Høyer–Tapp: area $2^{b/5}$, time $2^{b/2}$.

Concretely: b = 500.

Brassard–Høyer–Tapp, quantum: area 2^{100} , time 2^{250} .

Grover–Rudolph, quantum: area 2^{100} , time 2^{200} .

Iteration, quantum or classical: area 2^{100} , conj. time 2^{150} .

 $T = 2^{b/2}/A$ is optimal for generic classical algorithms. Conjecture: also for quantum. Naive free-communication model:

Brassard–Høyer–Tapp, quantum: area 2^{100} , time 2^{200} .

Grover–Rudolph, quantum: area 2^{100} , time 2^{200} .

Parallel tables (new), quantum: area 2^{100} , time 2^{150} .

Iteration, quantum or classical: area 2^{100} , conj. time 2^{150} .

Important notes:

 Optimal quantum computers seem to be classical computers!
 Clear quantum impact upon factorization, preimages, et al.
 but not upon collisions. Important notes:

 Optimal quantum computers seem to be classical computers!
 Clear quantum impact upon factorization, preimages, et al.
 but not upon collisions.

2. This algorithm isn't new.
M = 1: 1975 Pollard.
General case: famous
1994 van Oorschot-Wiener
paper, four years before
1998 Brassard-Høyer-Tapp.