YQuantum Writeup

We present a quantum hash function based on quantum walks on a cyclic graph of N nodes. Futhermore, we demonstrate that our hash function is deterministic, sufficiently random in the co-domain, somewhat resists preimage attacks, has few collisions, and uses fewer than 20 qubits.

This repository contains two solutions. The second one is able to hash inputs of arbitrary length, thus satisfying the requirements for the bonus problem. The bonus solution only differs in how it processes the input.

Our circuit uses all 20 qubits to ensure that the hash function may take inputs of arbitrary lengths. The bottom 4 qubits are "coins" that determine the evolution of the quantum walk. The remaining 16 qubits act as registers to record the current position of the pointer.

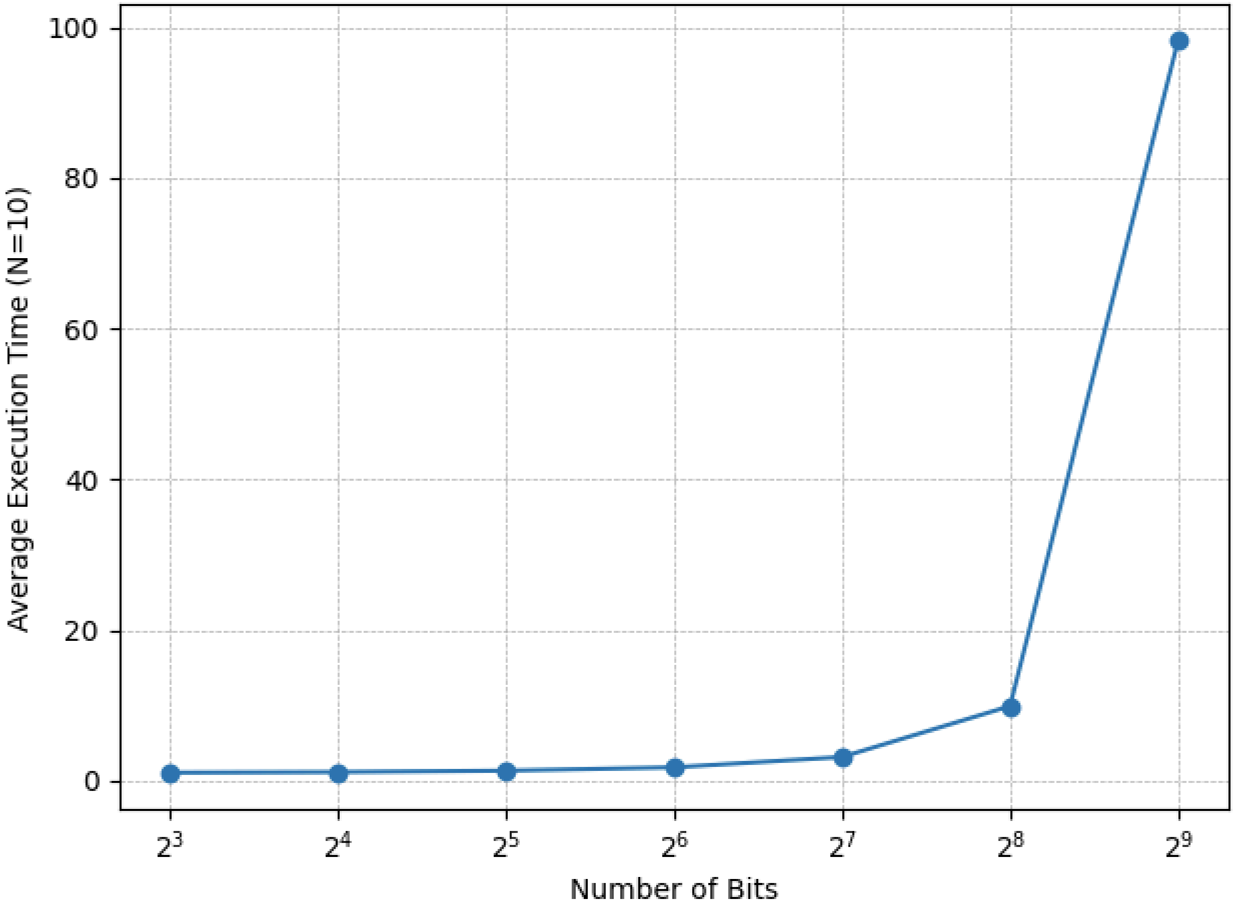
The input bitvector is padded as needed with zeros. The input is then chunked into slices of length 2. We then use Hadamard gates controlled by the classical bits in each chunk and then entangle these with the position qubits with CNOT gates.

Everything is thoroughly mixed with a round of rotations about the y-axis and futher entanglement via CNOT gates with neighbors.

Finally, we pass every qubit through a Hadamard gate and extract the state.

Performance Analysis

The performance of our solution is commensurate with expectations given the fact that our hash function can meaningfully process arbitrary amounts of data with quantum gates on simulated hardware. Numerical results are given below:



Given that the number of gates used overall increases exponentially, we can conclude that this algorithm would at least be exponential in the number of input bits.

The `quhash` algorithm if it were lazily modified to handle arbitrary powers of 2, has time complexity $O(2^N)$ on a quantum computer. The number of parameters is clearly exponential in the number of total bits, hence the cost of a the hash function also increases exponentially.

While thinking about quantum walks was useful in creating our solution, reasoning about its properties is best done with a more general view of one-way functions. Suppose we have a function $\psi : \{0,1\}^n \to (\mathcal{H}^2)^{\otimes s}$ of the form

$$\psi : u \mapsto \cos\left(\frac{2\pi u}{2^n}\right)|0\rangle + \sin\left(\frac{2\pi u}{2^n}\right)|1\rangle$$

This actually serves as a prototype for many quantum hashes in general, including ours. In particular, those hashes that look like successive multiplications by Fourier coefficients correspond exactly to the the class of one-way functions with coefficients in $\mathbb{Z}_{2^n}$. Another common choice for an abelian group includes $\mathbb{Z}_n^2$. Proving that one-way functions of a certain mathematical form are always collision and preimage resistant will suffice to prove it in our case.

Proof

General pre-image resistance follows from the fact that in general, we are projecting information onto a space far smaller than the domain. This is thus a routine application of Holevo's theorem, and we find that quantum information forbids extracting more than 20 bits from our 20 qubits.

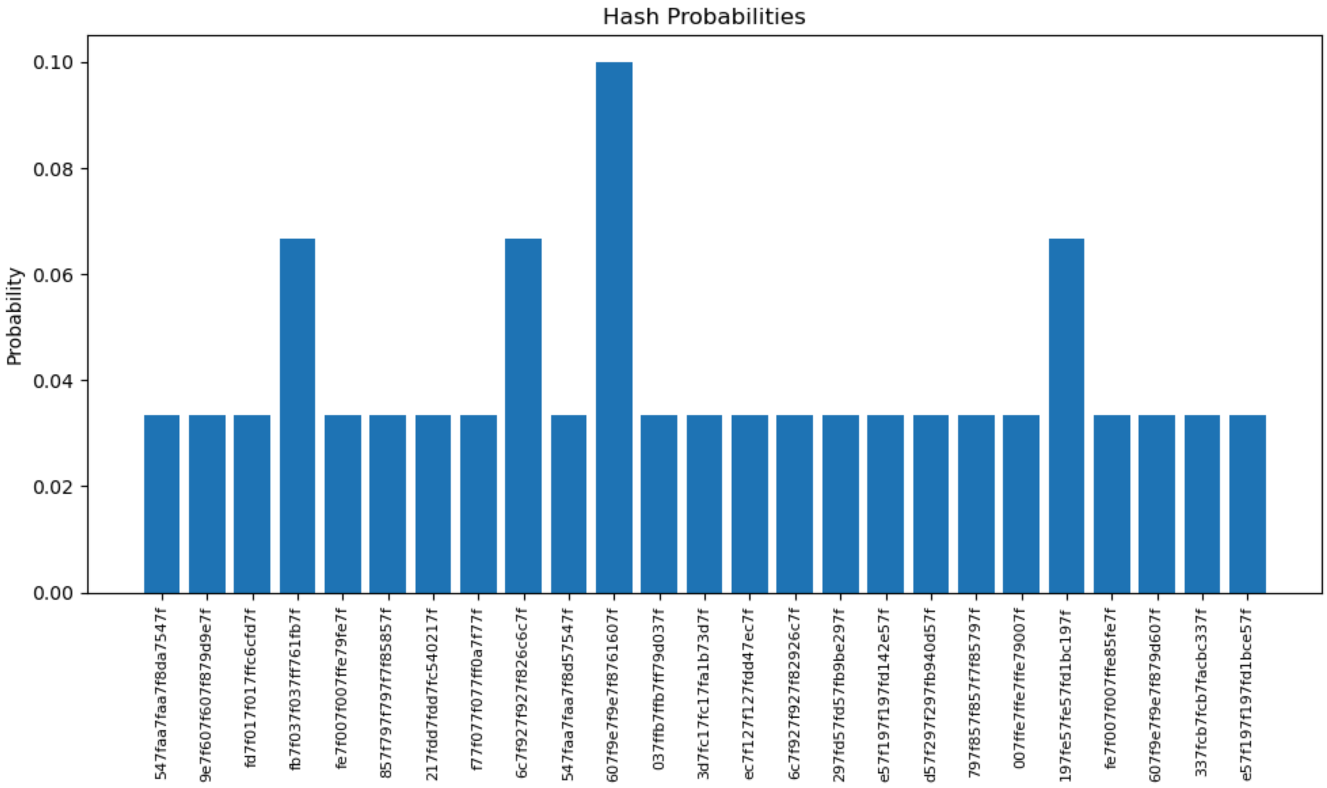
For collision resistance, we can observe that the general quantum hash function

$$|\psi_S(w)\rangle = \frac{1}{\sqrt{|S|}} \sum_{x \in S} \chi_w(x) |x\rangle,$$

where χ_w is the character of some abelian group and $S \subseteq G$, is collision resistant precisely when $\forall w, \frac{1}{|S|} |\sum_{x \in S} \chi_w(x)|$ is bounded by a small ε. We find that the quantity

$$|\langle \psi_S(w_1) | \psi_S(w_2) \rangle| = \frac{1}{|S|} \left| \sum_{x \in S} \chi_{w_1}^* \chi_{w_2} \right|,$$

which intuitively measures the degree to which different states can potentially give the same hash, is bounded by the same ε because the product of two distict non-trivial characters is a non-trivial character. When $G = \mathbb{Z}_{2^n}$, $\chi_w(x) = e^{\frac{2\pi wx}{2^n}}$, and so the expected ε is small by the basic theory of the Fourier transform. Furthermore, S can be chosen in such a way that ε is minimal, allowing for novel optimization techniques.



Analysis of `quhash`

The `quhash` algorithm works creating a parametrized circuit and using values derived from the hash for the parameters. Due to the nature of how `quhash` is parametrized, it does not scale to inputs of arbitrary lengths. In fact, alternative parametrizations based on "controlling" CRY/CRX gates with a classical bit would have scaled better since adding a new bit would then mean adding a new layer to the circuit. This analysis will treat the complexity, pre-image resistance, and collision resistance of `quhash`.

Pre-Image Resistance

Quantum information theory informs us that a bitvector hashed in a single pass though a quantum circuit is protected by Holevo's theorem, which essentially boils down to the fact that one can only extract n classical bits from n qubits. The domain is far larger than the co-domain, so this aspect of `quhash`'s security is already grounded in a mountain of literature.

Collision Resistance

We say a quantum hash function is collision resistant when the state when two distinct inputs will produce two states with a small inner product. Since `quhash` is just a bunch of rotations parametrized by integer values, the arguments used for our function apply in the same way.