# Dense Quantum Hashing

A. Vasiliev

2024

# Classical Cryptographic Hashing

Hashing is a mapping that

- Creates a short presentation of classical information;
- Can be efficiently computed given classical input;
- Extracting this input from the hash is hard (one-way property);
- Makes it hard to find a collision.

Resistance of classical hashing is related to $P$ vs $NP$ problem.

# What is Quantum Hashing?

Quantum Hashing is a mapping that

- Creates a quantum state from classical information;
- Can be efficiently computed given classical input;
- Makes it impossible to extract this input from the quantum hash;
- Allows to distinguish hashes of different inputs with high probability, which also implies a reliable equality test.

Quantum Hashing can be used in quantum cryptography and communications, integrity checks, etc.

# Quantum Hashing

## Quantum Collisions

There can be no collisions in the classical sense. That is, different messages can always lead to different hashes (i.e. different quantum states). But comparison of quantum states is generally probabilistic and we need to distinguish different hashes with high probability.

A quantum collision is a situation when a procedure that tests an equality of quantum hashes outputs true, while hashes are different. Since non-orthogonal quantum states cannot be perfectly distinguished, we require them to be “nearly orthogonal”.

# Quantum Hashing

## Properties

We call a mapping $w \mapsto |\psi(w)\rangle$ $\varepsilon$-collision resistant if

$$|\langle\psi(w_1)\,|\,\psi(w_2)\rangle| < \varepsilon$$

for any pair of messages $w_1$, $w_2$, $w_1 \neq w_2$.

We call a mapping $w \mapsto |\psi(w)\rangle$ $\delta$-resistant to inversion, if the probability of correctly extracting input $w$ out of its image $|\psi(w)\rangle$ is bounded by $\delta$.

We note that resistance to inversion is given by the fundamental results from quantum information theory. In particular, we have that $\delta \leq d/q$, where $q$ is the dimension of the input space, and $d$ is the size of the quantum state space.

# Quantum Hashing

Let $S = \{s_0, s_1, ..., s_{d-1}\} \subseteq \mathbb{Z}_q$ is an $\varepsilon$-biased set, i.e.

$$bias(S) = \max_{x \neq 0} \frac{1}{d} \left| \sum_{j=0}^{d-1} e^{i2\pi s_j x/q} \right| < \varepsilon$$

# Quantum Hashing

Let $S = \{s_0, s_1, ..., s_{d-1}\} \subseteq \mathbb{Z}_q$ is an $\varepsilon$-biased set, i.e.

$$bias(S) = \max_{x \neq 0} \frac{1}{d} \left| \sum_{j=0}^{d-1} e^{i2\pi s_j x/q} \right| < \varepsilon$$

We define a quantum hash function $\psi_S : \mathbb{Z}_q \to \mathcal{H}^d$ as following:

$$|\psi_S\rangle = \frac{1}{\sqrt{d}} \sum_{j=0}^{d-1} e^{i2\pi s_j x/q} |j\rangle.$$

# Quantum Hashing

## Properties

The collision probability for a function $\psi_S(w)$ is defined by

$$\max_{w_1 \neq w_2} |\langle \psi_S(w_1) \,|\, \psi_S(w_2) \rangle| = \max_{x=(w_2 - w_1)\neq 0} |\langle \psi_S(0) \,|\, \psi_S(x) \rangle| =$$

$$= \max_{x \neq 0} \frac{1}{d} \left| \sum_{k=1}^{d} e^{i2\pi s_k x/q} \right| = bias(S) \leq \varepsilon,$$

since $S$ is an $\varepsilon$-biased set.

# Quantum Hashing

## Properties

The collision probability for a function $\psi_S(w)$ is defined by

$$\max_{w_1 \neq w_2} |\langle \psi_S(w_1) \,|\, \psi_S(w_2) \rangle| = \max_{x=(w_2-w_1)\neq 0} |\langle \psi_S(0) \,|\, \psi_S(x) \rangle| =$$

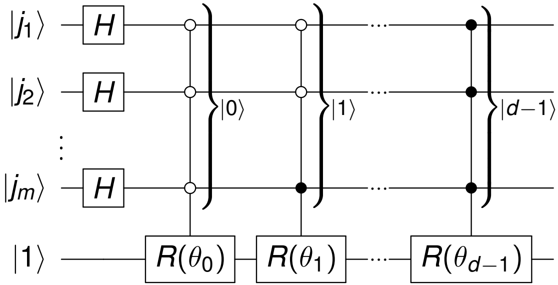
$$= \max_{x\neq 0} \frac{1}{d} \left| \sum_{k=1}^{d} e^{i2\pi s_k x/q} \right| = bias(S) \leq \varepsilon,$$

since $S$ is an $\varepsilon$-biased set.

This function is $\delta$-resistant to inversion with $\delta = \frac{d}{q}$, where $d = O\left(\frac{\log q}{\varepsilon^2}\right)$.

# Algorithm for Quantum Hashing

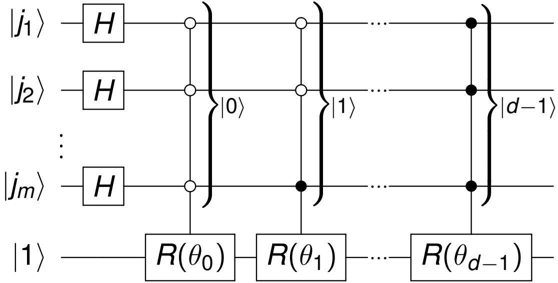
Here is the algorithm for the quantum hash function $\frac{1}{\sqrt{d}}\sum_{j=0}^{d-1} e^{i2\pi s_j x/q}|j\rangle$ on $m = \log d$ qubits (+ ancillas)



$R(\theta_j)$ is the rotation by the angle $\theta_j = \frac{4\pi s_j x}{q}$ around the $\hat{z}$ axis of the Bloch sphere.

# Algorithm for Quantum Hashing

Here is the algorithm for the quantum hash function $\frac{1}{\sqrt{d}}\sum_{j=0}^{d-1} e^{i2\pi s_j x/q}|j\rangle$ on $m = \log d$ qubits (+ ancillas)



The depth of this circuit is $O(dm) = O(\log q \log\log q)$.

# Algorithm for Quantum Hashing

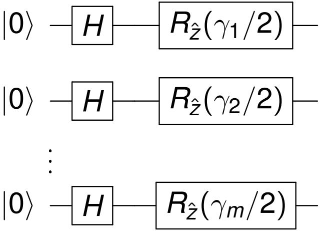
Let $B = \{b_1, b_2, \ldots, b_m\} \subseteq \mathbb{Z}_q$, and elements of $S = \{s_0, s_1, ..., s_{d-1}\} \subseteq \mathbb{Z}_q$ are given by sums of all possible subsets of $B$, i.e. for $j = j_1 j_2 \ldots j_m$

$$s_j = j_1 * b_1 + j_2 * b_2 + \cdots + j_m * b_m.$$

Remark

Here we already consider $s_0 = 0$.

# Algorithm for Quantum Hashing


$|0\rangle$
$H$
$R_{\hat{z}}(\gamma_1/2)$
$|0\rangle$
$H$
$R_{\hat{z}}(\gamma_2/2)$
$\vdots$
$|0\rangle$
$H$
$R_{\hat{z}}(\gamma_m/2)$

Algorithm of depth 2!

# Algorithm for Quantum Hashing

These equivalences are easy to verify:

$$\left[\frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{2\pi b_1 x}{q}}|1\rangle)\right] \otimes \cdots \otimes \left[\frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{2\pi b_m x}{q}}|1\rangle)\right] =$$

$$\frac{1}{\sqrt{2^m}}\left[\sum_{j_1=0}^{1} e^{i\frac{2\pi(j_1 * b_1)x}{q}}|j_1\rangle\right] \otimes \cdots \otimes \left[\sum_{j_m=0}^{1} e^{i\frac{2\pi(j_m * b_m)x}{q}}|j_m\rangle\right] =$$

$$\frac{1}{\sqrt{2^m}}\sum_{j_1=0}^{1}\cdots\sum_{j_m=0}^{1} e^{i\frac{2\pi(j_1 * b_1 + \cdots + j_m * b_m)x}{q}}|j_1\cdots j_m\rangle = \frac{1}{\sqrt{d}}\sum_{j=0}^{d-1} e^{i\frac{2\pi s_j x}{q}}|j\rangle.$$

Vasiliev A.V. Constant-Depth Algorithm for Quantum Hashing // Russian Microelectronics. - 2023. - Vol. 52, Suppl. 1. - P. S399–S402. DOI: 10.1134/S106373972360067X

# Constant-Depth Algorithm for Quantum Hashing

This corresponds to

$$|\psi_j(x)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{2\pi b_j x}{q}}|1\rangle),$$

$$|\psi(x)\rangle = |\psi_1(x)\rangle \otimes \cdots \otimes |\psi_m(x)\rangle$$

D. A. Turaykhanov, D. O. Akat'ev, A. V. Vasiliev, F. M. Ablayev, and A. A. Kalachev. Quantum hashing via single-photon states with orbital angular momentum // Physical Review A. - 2021. - Vol. 104, No. 5. - Art. No. 052606. DOI: 10.1103/PhysRevA.104.052606

# Constant-Depth Algorithm for Quantum Hashing

$$|\psi_j(x)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{2\pi b_j x}{q}}|1\rangle),$$
$$|\psi(x)\rangle = |\psi_1(x)\rangle \otimes \cdots \otimes |\psi_m(x)\rangle$$

The existence of a parameter set for such a function was considered in Vasiliev A.V. Collision Resistance of the OAM-based Quantum Hashing // Lobachevskii Journal of Mathematics. - 2023. - Vol. 44, No. 2. - P. 758–761. DOI: 10.1134/S1995080223020361

# Dense Quantum Hashing

We propose a quantum hash function that encodes classical information by the following superposition:

$$|\psi_j(x,y)\rangle = \cos\frac{\pi b_j x}{q}|0\rangle + e^{i\frac{2\pi b_j y}{q}}\sin\frac{\pi b_j x}{q}|1\rangle,$$
$$|\psi(x,y)\rangle = |\psi_1(x,y)\rangle \otimes \cdots \otimes |\psi_m(x,y)\rangle$$

The pair of arguments $(x,y)$ can be interpreted as the split of a larger input, or as a pair of an input and a key.

# Dense Quantum Hashing

We propose a quantum hash function that encodes classical information by the following superposition:

$$|\psi_j(x,y)\rangle = \cos\frac{\pi b_j x}{q}|0\rangle + e^{i\frac{2\pi b_j y}{q}} \sin\frac{\pi b_j x}{q}|1\rangle,$$

$$|\psi(x,y)\rangle = |\psi_1(x,y)\rangle \otimes \cdots \otimes |\psi_m(x,y)\rangle$$

The set of parameters $B = \{b_1, \ldots, b_m\}$ should be computed to minimize the probability of quantum collisions.

# Thank You!

# References

- Vasiliev A.V. Constant-Depth Algorithm for Quantum Hashing // Russian Microelectronics. - 2023. - Vol. 52, Suppl. 1. - P. S399–S402. DOI: 10.1134/S106373972360067X
- Turaykhanov D.A., Akat'ev D.O., Vasiliev A.V., Ablayev F.M., Kalachev A.A. Quantum hashing via single-photon states with orbital angular momentum // Phys Rev A. - 2021. - Vol. 104, Iss. 5. - Art. No. 052606.
- Vasiliev A.V. Collision Resistance of the OAM-based Quantum Hashing // Lob J Math. - 2023. - Vol. 44, No. 2. - P. 758–761.
- Ablayev F., Ablayev M., Vasiliev A. Quantum Hashing and Fingerprinting for Quantum Cryptography and Computations // Lecture Notes in Computer Science, CSR 2020. - 2020. - V. 12159. - P. 1-15
- Vasiliev, A. Quantum hashing for finite abelian groups // Lobachevskii Journal of Mathematics. - 2016. - V. 37, No. 6. - P. 751-754.

# Small-biased sets

Lets define the notion of the bias for the subset $S = \{s_0, \ldots, s_{d-1}\} \subseteq \mathbb{Z}_q$:

$$bias(S,x) = \frac{1}{d}\left|\sum_{j=0}^{d-1} e^{i2\pi s_j x/q}\right|$$

$$bias(S) = \max_{x \neq 0} bias(S,x)$$

The set $S$ is called $\varepsilon$-biased, if

$$bias(S) \leq \varepsilon.$$

# Small-biased sets

We note the following equivalence between $\varepsilon$-biased sets.

Let $S = \{s_0, \ldots, s_{d-1}\}$ and $S' = \{0, (s_1 - s_0), \ldots, (s_{d-1} - s_0)\}$. Then for each $x \in \mathbb{Z}_q$ $bias(S, x) = bias(S', x)$, i.e. the set $S$ is equivalent (in terms of its bias) to $S'$.

$$\frac{1}{d}\left|\sum_{k=0}^{d-1} e^{i2\pi s_k x/q}\right| = \frac{1}{d}\left|e^{i\frac{2\pi s_1 x}{q}}\right|\left|\sum_{k=0}^{d-1} e^{i2\pi(s_k - s_0)x/q}\right| = \frac{1}{d}\left|\sum_{k=0}^{d-1} e^{i2\pi(s_k - s_1)x/q}\right|$$

$$bias(S, x) = \frac{1}{d}\left|\sum_{k=0}^{d-1} e^{i2\pi s_k x/q}\right| = \frac{1}{d}\left|\sum_{k=0}^{d-1} e^{i2\pi(s_k - s_0)x/q}\right| = bias(S', x)$$

# Small-biased sets

We note the following equivalence between $\varepsilon$-biased sets.
Let $S = \{s_0, \ldots, s_{d-1}\}$ and $S' = \{0, (s_1 - s_0), \ldots, (s_{d-1} - s_0)\}$. Then for each $x \in \mathbb{Z}_q$ $bias(S, x) = bias(S', x)$, i.e. the set $S$ is equivalent (in terms of its bias) to $S'$.

Thus, without loss of generality we may consider $s_0$ to be equal 0.

# The existence of Small-biased sets

It is known that an $\varepsilon$-biased set $S \subseteq \mathbb{Z}_q$ of size $O\left(\frac{\log q}{\varepsilon^2}\right)$ exists.

# The existence of Small-biased sets

It is known that an $\varepsilon$-biased set $S \subseteq \mathbb{Z}_q$ of size $O\left(\frac{\log q}{\varepsilon^2}\right)$ exists.

There are explicit constructions of such sets of size polynomial in $\log q$ and $1/\varepsilon$.

# Algorithm for Quantum Hashing

$$\left|\psi_j(x)\right\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\frac{2\pi b_j x}{q}}|1\rangle),$$
$$|\psi(x)\rangle = |\psi_1(x)\rangle \otimes \cdots \otimes |\psi_m(x)\rangle$$

Lemma

There exists $B = \{b_1, \ldots, b_m\} \subseteq \mathbb{Z}_q$ of size $m = \lceil 16 \ln q \rceil$, such that for each pair $x_1 \neq x_2$ we can distinguish $\left|\psi_j(x_1)\right\rangle$ and $\left|\psi_j(x_2)\right\rangle$ with error probability bounded by $1/q^4$.

# Algorithm for Quantum Hashing

Lemma

There exists $B = \{b_1, \ldots, b_m\} \subseteq \mathbb{Z}_q$ of size $m = \lceil 16 \ln q \rceil$, such that for each pair $x_1 \neq x_2$ we can distinguish $\left|\psi_j(x_1)\right\rangle$ and $\left|\psi_j(x_2)\right\rangle$ with error probability bounded by $1/q^4$.

This set generates the $\varepsilon$-biased set $S = \{s_0, \ldots, s_{d-1}\}$ of size $O(q)$ with $\varepsilon = 1/q^4$.

The corresponding algorithm for computing quantum hash function would be based on the Hilbert space of dimension $O(q)$ and would have depth $O(1)$.