Journal of Engineering and Applied Sciences 12 (4): 877-880, 2017
ISSN: 1816-949X

Minimizing Collisions for Quantum Hashing

Alexander Vasiliev, Marat Latypov and Mansur Ziatdinov
Kazan Federal University, Kremlyovskaya St. 18, 420008 Kazan, Russian Federation

Abstract: Hashing is a widely used technique in computer science. The recently proposed quantum hashing has also proved its usefulness in a number of applications. The key property of both classical and quantum hashing is the ability to withstand collisions however, the notion of collision itself is different in the classical and quantum setting. In this study we analyze the set of numeric parameters that determine the probability of quantum collisions for the quantum hashing. Although, there is a general method of obtaining good hashing parameters, it makes sense for comparatively large inputs. That is why we construct different methods to complement the general one. We present two explicit optimization algorithms for computation of quantum hashing parameters: one is based on the genetic approach and the other uses the annealing simulation. The solution to the considered optimization problem can be used for the variety of quantum hash functions and also provides a solution to the general problem of constructing sets of pairwise distinguishable states in low-dimensional spaces.

Key words: Quantum computation, quantum information, quantum hashing, genetic algorithm, annealing simulation algorithm

INTRODUCTION

Hashing is a well-known technique, widely used in computer science. Following the ideas and properties of the cryptographic hashing, we have proposed its quantum analogue in (Ablayev and Vasiliev, 2014). Just like in classical case it can find applications in different communication scenarios including quantum digital signature protocol from (Gottesman and Chuang, 2001) and quantum communication protocols (e.g. in the one-way quantum communication model and simultaneous message passing model (Vasiliev, 2015). It has also proved useful for constructing efficient quantum algorithms (Ablayev and Vasiliev, 2014). This quantum hash function was generalized in (Ablayev and Ablayev, 2014) by proposing a method for constructing new quantum hash functions from a specific family of functions and an arbitrary universal hash family.

The key property of both classical and quantum hashing is the collision resistance. Ablayev and Vasiliev (2014), we have discussed the notion of quantum collision. The reason why we have defined it is the observation that in quantum hashing there might be no collisions in the classical sense: since quantum hashes are quantum states they can store arbitrary amount of data and can be different for unequal messages. But the procedure of comparing those quantum states implies measurement which can lead to collision-type errors.

We have defined a quantum collision to be a situation when a procedure that tests an equality of quantum hashes outputs true, while hashes are different. This procedure can be a well-known SWAP-test (Buhrman *et al*., 2001) or something that is adapted for specific hash function. Anyway, it deals with the notion of distinguishability of quantum states. And since non-orthogonal quantum states cannot be perfectly distin guished we require that the pairwise inner products of the quantum hashes for different inputs are bounded.

A quantum hash function can be perfect, i.e., having no quantum collisions at all. However, this would mean the absence of the other important property of the cryptographic hashing-the pre-image resistance. The trade-off between these 2 properties and the construction of a "balanced" quantum hash function were discussed in (Ablayev and Marat, 2015).

Here we concentrate on the problem of minimizing the probability of collisions for the quantum hash function from (Ablayev and Vasiliev, 2014). The underlying numeric optimization problem is the same for all further generalizations of that function from (Ablayev and Ablayev, 2014; Buhrman *et al*., 2001; Ablayev and Marat, 2015) and thus our solution can be used for the variety of quantum hash functions.

As noted above the probability of quantum collisions is determined by the value of the inner product of quantum hashes for different inputs. Therefore, the

Corresponding Author: Alexander Vasiliev, Kazan Federal University, Kremlyovskaya St. 18, 420008 Kazan, Russian Federation

solution to the problem of minimizing quantum collisions with additional requirement of small quantum hash size (and thus pre-image resistance) would also solve the problem of constructing sets of pairwise-distinguishable states in low-dimensional spaces (Buhrman *et al*., 2001).

Preliminaries: In this study, we recall a quantum hashing function from (Ablayev and Vasiliev, 2014). Let $q = 2^n$ and $B = \{b_1, b_2, ..., b_d\} \subset Z_q$. We define a quantum hash function $\psi_{q,B}$: $\{0, 1\}^n \to (H^2)^{\otimes(\log d+1)}$ as follows. For an input $x \in \{0,1\}^n$, we let:

$$\left|\psi_{q,B}(x)\right\rangle = \frac{1}{\sqrt{d}}\sum_{i=1}^{d}\left|i\right\rangle\left(\text{Cose}\frac{2\pi b_i x}{q}\left|0\right\rangle + \sin\frac{2\pi b_i x}{q}\left|1\right\rangle\right).$$

It follows from this definition that the quantum hash $|\psi_{q,B}(x)\rangle$ of an n-bit bit string x consists of log d+1 qubits. We have shown in (Ablayev and Vasiliev, 2014) that d can be of order O(n) without losing the quality of hashing. The set $B = \{b_1, b_2, ..., b_d\}$ of hashing parameters not only determines the size of the hash but also gives the function $\psi_{q,B}$ an ability to withstand collisions, i.e., to distinguish different hashes with bounded error probability. We have called this property δ-resistance. Formally, for $\delta \in (0, 1)$, we call a function ψ: $x \to (H^2)^{\otimes s}$ δ-resistant if for any pair w, w′ of different inputs:

$$\left|\left\langle\psi(w)\middle|\psi(w')\right\rangle\right| \le \delta$$

The value of q for the hash function $\psi_{q,B}$ entirely depends on q (which is fixed here by the size of the input) and the set B, i.e., $\delta = \delta(q, B)$. Ablayev and Vasiliev (2014) we have used a construction for this set of polylogarithmic size (in n) based on (Razborov *et al*., 1993). We have also proved the following result.

Theorem: For arbitrary $\delta \in (0, 1)$ there exists a set $B = \{b_1, b_2, ..., b_d\}$ of size $d = \lceil (2/\delta^2) \ln 2q \rceil$ such that the quantum hash function $\psi_{q,B}$ is δ-resistant. In other words, for arbitrary $\delta \in (0, 1)$ it is possible construct a δ-resistant quantum hash function $\psi_{q,B}$ that would produce a log d+1 = O(log log q) = O(log n)-qubit hash out of n-bit input.

MATERIALS AND METHODS

Optimization problem: It can be easily seen that for the function $\psi_{q,B}(x)$ we have:

$$\left|\left\langle\psi_{q,B}(w)\middle|\psi_{q,B}(w)\right\rangle\right| = \left|\frac{1}{d}\sum_{j=1}^{d}\text{Cos}\frac{2\pi b_j(w-w')}{q}\right|$$

and we want it to be less than some δ for any value of (w-w′) except for 0. Thus, the optimization problem that arouses here is the following. For a fixed q minimize the target function:

$$\delta(q,B) = \max_{x\neq 0}\left|\frac{1}{d}\sum_{j=1}^{d}\cos\frac{2\pi b_j(w-w')}{q}\right|$$

over all $B = \{b_1, b_2, ..., b_d\} \subset zq$. The best possible solution exists for B = zq , since if $x \neq 0$

$$\delta(q,B) = \left|\frac{1}{q}\sum_{b\in z_q}\cos\frac{2\pi bx}{q}\right| = \left|\frac{1}{q}\text{Re}\left(\sum_{b\in z_q}e^{-i\frac{2\pi bx}{q}}\right)\right|$$

However, this would mean that the size of the hash is even larger than the input and hashing loses one of its important properties. So we require that d should be much smaller than q (preferably, d = O(log q) and we actually solve the above problem several times for increasing d until it gives us the set B with desired value of δ (q, B).

Genetic algorithm: The idea of genetic algorithm was proposed by Holland (1975) for the investigation of natural adaptation but it was later used for solving computational search problems. It is based on the evolutionary theory by Charles Darwin and uses biological terms to describe the algorithm. When applied to our optimization problem we define them as follows:

- Chromosome is a set of numeric parameters B
- Individual is a solution to the search problem and is described by a set of properties (its chromosomes)
- Since our individuals have a single chromosome, we will use both terms interchangeably
- Gene is an element of the set B
- Locus is a position of a gene on the chromosome
- Allele is a set of sequential genes on the chromosome
- Population is a fixed number of individuals
- Fitness describes the quality of the solution (individual). In our case fitness is given by the function δ (q, B) and our aim is to minimize it

To start the algorithm we randomly generate a family of sets (a population); all sets of genes are kept sorted to ease operations with them. The size of the population is equal to |B| but no more than 100. Such a small number of individuals is chosen to reduce the running time of the algorithm. Then the population is evolved as following.

First of all, the new individuals are created from existing ones using the crossover process. It is performed by choosing 2 random locations in chromosomes and exchanging alleles between these locations. That is 2 individuals are created after the crossover of two parent chromosomes. After the crossover both children can mutate with given probability.

Mutation is a probabilistic process of changing some genes of an individual. The probability of mutation for each individual is set to 0.25. The probability of mutation of a certain gene is set to 0.1. All genes have equal probability of mutation. The mutation of a gene is described by the following equation:

$$\begin{cases} y_i' = y_i + (\max - y_i)\left(1 - r(1 - \frac{t}{T})^b\right), \text{ if } p = 0, \\ y_i - (y_i - \min)\left(1 - r(1 - \frac{t}{T})^b\right), \text{ if } p = 1 \end{cases}$$

Where:

$r \in [0, 1]$ = A random number
T = A number of generation
p = A total number of generations
p = Picked randomly from {0, 1}
b = An adjustable parameter which we have set to 1, max = q/2 and min = 0

Finally, for all individuals the fitness function is evaluated and distinct individuals with the best results give the next generation. The evolution process repeats the given number of iterations we have set the maximal number of generations to 100.

RESULTS AND DISCUSSION

Simulated annealing: We have also developed a simulated annealing algorithm to compute the set B. This algorithm is a metaheuristic search algorithm and it is described in (Kirkpatrick *et al*., 1983). We used concurrent-sa library for Haskell language for general procedure of simulated annealing. Simulated annealing is inspired by a physical process of melting some substance and then lowering the temperature slowly. This process allows the substance to get to optimal state (i.e., state with the lowest energy).

To apply this algorithm to our problem we need to define what is a state what is the energy of a state and how the state changes. We define the state as some set B with elements from z_q. We define the energy of a set B as the value of δ (q, B). Set B changes into the other (neighbor) set if these sets differ only in one element. So, we start by generating a family of random states then we change them according to current temperature. This temperature slowly decreases. After sufficient time population will have sets with low δ (q). The change of temperature, required number of iterations and concurrency are handled by concurrent-sa library. In our program we wrote only the size of the family, the function for changing states and the function for generating random sets.

Summary: To summarize we have analyzed the problem of quantum collisions for the quantum hash function and proposed two explicit optimization algorithms for constructing sets of numeric parameters that minimize the probability of collisions.

CONCLUSION

The algorithms presented in this study can be used as a part of the following strategy of minimizing quantum collisions. If the size of the input is small we can use a brute-force algorithm to find the best possible set of parameters for quantum hashing. Otherwise, if the size of the input is large a constructive algorithm can give asymptotically good results. But in the case of moderate-sized inputs both of these algorithms fail: the first one because of the time complexity and the other because of the oversized quantum hash. In this case our algorithms can be used to construct a balanced quantum hash function which is both compact and collision resistant.

ACKNOWLEDGEMENTS

The research is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University. Research was in part supported by the Russian foundation for basic research (under the grants 14-07-00878, 14-07-00557, 15-37-21160).

REFERENCES

Ablayev, F. and A. Marat, 2015. On the concept of cryptographic quantum hashing. MSc Thesis, Cornell University, Ithaca, New York.

Ablayev, F. and A. Vasiliev, 2014. Computing Boolean Functions via Quantum Hashing. In: Computing with New Resources, Cristian, S.C., F. Rusins and K. Iwama (Eds.). Springer, Switzerland, Europe, ISBN:978-3-319-13349-2, pp: 149-160.

Ablayev, F. and M. Ablayev, 2014. Quantum Hashing via E-Universal Hashing Constructions and Freivalds Fingerprinting Schemas. Proceedings of the 16th International Workshop on Descriptional Complexity of Formal Systems (DCFS) 2014, August 5-8, 2014, Springer, Turku, Finland, pp: 42-52.

Buhrman, H., R. Cleve, J. Watrous and R.D. Wolf, 2001. Quantum fingerprinting. Phys. Rev. Lett., Vol. 87.

Gottesman, D. and I. Chuang, 2001. Quantum digital signatures. MSc Thesis, Cornell University, Ithaca, New York, USA.

Holland, J.H., 1975. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence. 1st Edn., University of Michigan Press, Ann Arbor, MI., USA., ISBN-13: 9780472084609, Pages: 183.

Kirkpatrick, S., C.D. Gelatt Jr. and M.P. Vecchi, 1983. Optimization by simulated annealing. Science, 220: 671-680.

Razborov, A., E.N.D.R.E. Szemeredi and A. Wigderson, 1993. Constructing small sets that are uniform in arithmetic progressions. Comb. Probab. Comput., 2: 513-518.

Vasiliev, A., 2015. Quantum communications based on quantum hashing. Intl. J. Appl. Eng. Res., 10: 31415-31426.