International Journal of Innovative Computing, Information and Control
Volume **10**, Number **5**, October **2014**

ICIC International ©2014 ISSN 1349-4198
pp. 1783–1790

QUANTUM FUZZY SIMILARITY HASH BASED ON QUDITS SEQUENCE COMPRESSION

Dong Cao and Yaoliang Song

School of Electronic Engineering and Optoelectronic Technology
Nanjing University of Science and Technology
No. 200, Xiaolingwei, Nanjing 210094, P. R. China
caodongcn@gmail.com; caod@njust.edu.cn

Received December 2013; revised April 2014

Abstract. *Hash algorithm has a wide range of applications. Similarity preserving hashing is a type of improved method of hash scheme. These algorithms can preserve the similarity existing in the relevant data files. The similarity preserving hashing algorithm is widely used in the field of computer forensics and data feature extraction, etc. This paper presents a novel construction method of quantum fuzzy similarity hash algorithm based on quantum information qudits sequence compression and traditional hashing algorithms. With the rapid development of quantum information technology, more and more traditional cryptographic algorithms and security protocols can be attacked by quantum algorithm and quantum Turing machine. Based on the quantum information representation of D-dimensional quantum system in Hilbert space and classical hashing algorithms, we construct a method of qudits sequence compression algorithms. The computational efficiency of sequence compression function is significantly improved. Compared to the classical similarity preserving hashing algorithm based on analogous construction schemes, our proposed novel quantum fuzzy similarity hash algorithm has higher security. This algorithm outperforms other similarity algorithms in security and computational efficiency. Theoretical analysis and performance simulation prove the quantum fuzzy similarity hashing algorithm is effective.*

Keywords: Quantum cryptography, Quantum hashing, Information security

1. **Introduction.** Similarity preserving hashing is a type of improved hash algorithm [1-13]. This improved cryptography scheme can preserve the similarity between these cognate data. This similarity preserving hashing algorithm can be used to distinguish similarity data sequences in which these inserted or adjusted data have highly resemblance. Based on the traditional hashing, similarity preserving hashing algorithm can solve the problem that the traditional hashing cannot achieve. Since these values of traditional hash are different with randomness.

In the literature [2] F. Breitinger and H. Baier proposed a novel construction method of MRSH. This algorithm is a type of classical similarity preserving hashing, and the performance of run time outperforms other similarity preserving hashing. F. Breitinger et al. in [11] proposed a novel similarity preserving hashing based on the methods of majority vote. The computational complexity of the proposed similarity preserving hashing is significantly reduced. However, the cryptographic security of the hashing algorithm is not robust. F. Breitinger and H. Baier in [7] proposed a fuzzy hashing algorithm utilizing with the random sequences. In the literature, they presented a bbHash method. In the problem of against ant blacklisting, this method is more robust and outperforms other similarity algorithms such as ssdeep. Similarity preserving hashing is widely used for computer forensics and data feature extraction, etc.

Based on aforementioned methods of classical similarity preserving hashing algorithm, we present a novel quantum fuzzy similarity hashing utilizing qudits sequence compression. Based on the qudit states representation of information quantum, we present a type of qudits sequence compression scheme in detail. And then, we introduce the method of qudits similarity comparison and performance analysis. Theoretical analysis and performance simulation prove the quantum fuzzy similarity hashing algorithm is security and validity.

2. *D*-Dimensional Quantum Systems.

Based on the representation of D-dimensional quantum system [13,14], we introduce the quantum information representation of 8-dimensional quantum systems.

2.1. Quantum qudit states representation.

The Hilbert space of quantum state is spanned by $|0\rangle, |1\rangle, |2\rangle, |3\rangle, \cdots, |D-1\rangle$, and then the quantum system is a D-dimensional system. Information of the quantum system can be denoted as qudit [13,14]. The Hermitian generators σ_r of $SU(D)$ can be denoted as the following:

$$\begin{aligned}
&\sigma_r = \Gamma_d = \tfrac{1}{\sqrt{d(d-1)}}\left(\sum_{e=1}^{d-1} |e\rangle\langle e| - (d-1)|d\rangle\langle d|\right), \\
&\qquad 2 \le d \le D, \quad r = 1, 2, 3, \cdots, D-1 \\
&\sigma_r = \Gamma_{de}^{(+)} = \tfrac{1}{\sqrt{2}}\left(|d\rangle\langle d| + |e\rangle\langle e|\right), \\
&\qquad 1 \le d < e \le D, \quad r = D, D+1, \cdots, \tfrac{(D+2)(D-1)}{2} \\
&\sigma_r = \Gamma_{de}^{(-)} = \tfrac{-i}{\sqrt{2}}\left(|d\rangle\langle d| - |e\rangle\langle e|\right), \\
&\qquad 1 \le d < e \le D, \quad r = D\tfrac{(D+2)}{2}, \cdots, D^2-1 \\
&\sigma_r\sigma_s = \tfrac{1}{D}\delta_{rs} + u_{rst}\sigma_t + iv_{rst}\sigma_t
\end{aligned} \tag{1}$$

where $d = 1, 2, \cdots, D$. $|d\rangle, |e\rangle$ stand for the orthonormal basis. The orthogonality is as the following:

$$\begin{aligned}
|d\rangle\langle d| &= \tfrac{I}{D} + \tfrac{1}{\sqrt{d(d-1)}}\left(-(d-1)\Gamma_d + \sum_{e=d+1}^{D}\Gamma_e\right) \\
|d\rangle\langle e| &= \tfrac{1}{\sqrt{2}}\left(\Gamma_{de}^{(+)} + i\Gamma_{de}^{(-)}\right), \quad 1 \le d < e \le D \\
|e\rangle\langle d| &= \tfrac{1}{\sqrt{2}}\left(\Gamma_{de}^{(+)} - i\Gamma_{de}^{(-)}\right), \quad 1 \le d < e \le D
\end{aligned} \tag{2}$$

2.2. Quantum information in 8-dimensional quantum systems.

Let $d = 8$, and then we can obtain the following orthogonal basis:

$$|0\rangle, |1\rangle, |2\rangle, |3\rangle, |4\rangle, |5\rangle, |6\rangle, |7\rangle \tag{3}$$

The qudits can be mapped to qubits as the following orthogonal basis:

$$\begin{gathered}
|0\rangle \Leftrightarrow |000\rangle, \\
|1\rangle \Leftrightarrow |001\rangle, \\
|2\rangle \Leftrightarrow |010\rangle, \\
|3\rangle \Leftrightarrow |011\rangle, \\
\cdots \\
|d\rangle \Leftrightarrow |\cdots\rangle, \\
\cdots \\
|7\rangle \Leftrightarrow |111\rangle
\end{gathered} \tag{4}$$

Each 3-qubits sequence $|B\rangle$ can be related to a qudits sequence. The corresponding relations are as follows:

$$|B\rangle = |b_2 b_1 b_0\rangle \Leftrightarrow |d_1\rangle \tag{5}$$

where $b_2, b_1, b_0 \in \{0, 1\}$, $d_1 \in \{0, 1, 2, 3, \cdots, 7\}$. And then, each classical byte with eight bits can be corresponded to sequence $|B\rangle$.

Giving a classical bits sequence S as follows:

$$\begin{aligned} S = {} & 001011111011010101001010100111 \\ & 110101010001 \cdots b_{k-1} b_k \end{aligned} \tag{6}$$

where $k \in \{0, 1\}$. Then the sequence S can be associated to a particular quantum sequence $|\phi\rangle_t$. The quantum sequence $|\phi\rangle_t$ can be denoted as $|\phi\rangle_t = |\vartheta_1\rangle |\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$, where $i = 1, 2, \cdots, t$, $\vartheta_i \in \{0, 1, 2, 3, \cdots, 7\}$. The quantum sequence has the following form:

$$\begin{aligned} |\phi\rangle_t = {} & |1\rangle\, |3\rangle\, |7\rangle\, |3\rangle\, |2\rangle\, |5\rangle\, |1\rangle\, |2\rangle\, |4\rangle\, |7\rangle\, |6\rangle\, |5\rangle \\ & |2\rangle\, |1\rangle\, |0\rangle \cdots |\vartheta_{t-1}\rangle\, |\vartheta_t\rangle \end{aligned} \tag{7}$$

3. **A Novel Constructed Method of Quantum Fuzzy Similarity Hash.**

In this section, we propose a novel quantum fuzzy similarity hash algorithm. This algorithm is a basic building for constructing fingerprint identification in the next section. The representation of qudits sequence compression algorithms is presented briefly in Subsection 3.1. Based on the qudits sequence compression algorithms, we present the construction method of quantum fuzzy similarity hash algorithms in detail in Subsection 3.2.

3.1. **Qudits sequence compression algorithms.**

Giving a qudits sequence $|\phi\rangle_t$, we perform a compression algorithm. In this paper, we always set $d = 8$, that is, we only discuss the 8-dimensional quantum systems [3,13,14]. The details of qudits sequence compression algorithms are as follows:

Let $|\phi\rangle_t = |\vartheta_1\rangle |\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$, where $i = 1, 2, \cdots, t$, $\vartheta_i \in \{0, 1, 2, 3, \cdots, 7\}$. We scan the qudits sequence from start to finish, and search these qudits $\left|\vartheta_{[\![s]\!]}\right\rangle$ in the qudits sequence $|\phi\rangle_t$ that equal the first qudit $|\vartheta_1\rangle$, where $[\![s]\!] = 2, 3, \cdots, t$, and $1 \leq s \ll t$. According to the information and coding theory, the distribution probability $P(|\vartheta_i\rangle)$ of an arbitrary qudit $|\vartheta_i\rangle$ existing in the qudits sequence $|\phi\rangle_t$ is approximately the same as long as the t is large enough. When t tends to infinity, the distribution probability $P(|\vartheta_i\rangle)$ tends to 1/8. However, the distribution location of $|\vartheta_s\rangle$ might be random.

Every qudits in the qudits sequence $|\phi\rangle_t = |\vartheta_1\rangle |\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$ are searched, and determine these elements $\left|\vartheta_{[\![s]\!]}\right\rangle$ matching the first qudit $|\vartheta_1\rangle$. That is $\left|\vartheta_{[\![s]\!]}\right\rangle = |\vartheta_1\rangle$ or not. In accordance with the order from the first to the last, each qudit in the qudits sequence $|\phi\rangle_t = |\vartheta_1\rangle |\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$ is labeled with a serial number $1, 2, 3, \cdots, t$.

If $\left|\vartheta_{[\![s]\!]}\right\rangle = |\vartheta_1\rangle$, then save $\left|\vartheta_{[\![s]\!]}\right\rangle$. Label these locations L of these qudits $\left|\vartheta_{[\![s]\!]}\right\rangle$, where $2 \leq L \leq t$. These qudits $\left|\vartheta_{[\![s]\!]}\right\rangle$ composited a subset $\wp_\vartheta$ of the set $\{|\vartheta_1\rangle, |\vartheta_2\rangle, |\vartheta_3\rangle, \cdots, |\vartheta_i\rangle, \cdots, |\vartheta_t\rangle\}$.

Based on these qudits $\left|\vartheta_{[\![s]\!]}\right\rangle$ in subset $\wp_\vartheta$, we determine position points, and then we can partition the qudits sequence $|\phi\rangle_t = |\vartheta_1\rangle |\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$ into $\Im_s$ as follows:

$$\begin{aligned} \Im_1 &= |\vartheta_1\rangle\, |\vartheta_2\rangle \cdots \left|\vartheta_{[\![1]\!]}\right\rangle \\ \Im_2 &= \left|\vartheta_{[\![1]\!]+1}\right\rangle \left|\vartheta_{[\![1]\!]+2}\right\rangle \cdots \left|\vartheta_{[\![2]\!]}\right\rangle \\ & \cdots \\ \Im_s &= \left|\vartheta_{[\![s-1]\!]+1}\right\rangle \left|\vartheta_{[\![s-]\!]+2}\right\rangle \cdots \left|\vartheta_{[\![s]\!]}\right\rangle \end{aligned} \tag{8}$$

According with these fragments $\Im_1, \Im_2, \cdots, \Im_s$, we perform compression algorithms in detail as the following formula:

$$C\left(\Im_h\right) = \left| \left(\sum_{r=\llbracket h-1 \rrbracket+1}^{\llbracket h \rrbracket} \vartheta_r \right) \bmod (8) \right\rangle \tag{9}$$

where $h = 1, 2, 3, \cdots, s$, letting $\llbracket 0 \rrbracket = 0$. Therefore, all these s compressed fragments $\Im_1, \Im_2, \cdots, \Im_s$ are as follows:

$$\begin{aligned}
C\left(\Im_1\right) &= \left| \left(\sum_{r=\llbracket 1-1 \rrbracket+1}^{\llbracket 1 \rrbracket} \vartheta_r \right) \bmod (8) \right\rangle = \left| \left(\sum_{r=\llbracket 0 \rrbracket+1}^{\llbracket 1 \rrbracket} \vartheta_r \right) \bmod (8) \right\rangle \\
&= \left| \left(\vartheta_1 + \vartheta_2 + \cdots + \vartheta_{\llbracket 1 \rrbracket} \right) \bmod (8) \right\rangle \\
C\left(\Im_2\right) &= \left| \left(\sum_{r=\llbracket 2-1 \rrbracket+1}^{\llbracket 2 \rrbracket} \vartheta_r \right) \bmod (8) \right\rangle = \left| \left(\sum_{r=\llbracket 1 \rrbracket+1}^{\llbracket 2 \rrbracket} \vartheta_r \right) \bmod (8) \right\rangle \\
&= \left| \left(\vartheta_{\llbracket 1 \rrbracket+1} + \vartheta_{\llbracket 1 \rrbracket+2} + \cdots + \vartheta_{\llbracket 2 \rrbracket} \right) \bmod (8) \right\rangle \\
&\cdots \\
C\left(\Im_s\right) &= \left| \left(\sum_{r=\llbracket s-1 \rrbracket+1}^{\llbracket s \rrbracket} \vartheta_r \right) \bmod (8) \right\rangle \\
&= \left| \left(\vartheta_{\llbracket s-1 \rrbracket+1} + \vartheta_{\llbracket s-1 \rrbracket+2} + \cdots + \vartheta_{\llbracket s \rrbracket} \right) \bmod (8) \right\rangle
\end{aligned} \tag{10}$$

Each compressed value $C\left(\Im_h\right)$ is a qudit.

3.2. **Quantum fuzzy similarity hash algorithms.** These compressed values of fragments $\Im_1, \Im_2, \cdots, \Im_s$ are arranged successively as the following new sequences $\mathfrak{R}_C$:

$$\mathfrak{R}_C = \left[C\left(\Im_1\right), C\left(\Im_2\right), \cdots, C\left(\Im_s\right) \right] \tag{11}$$

And then the new sequences $\mathfrak{R}_C$ are divided into many packets [3,11] in accordance with the order from left to right. Each packet contains eight elements. Right shifting four qudits, another adjacent packet is generated. From top to down, we can arrange these packets as a packet matrix M_p as the following:

$$M_p = \begin{bmatrix}
C\left(\Im_1\right) & C\left(\Im_2\right) & C\left(\Im_3\right) & C\left(\Im_4\right) & \cdots & C\left(\Im_8\right) \\
C\left(\Im_5\right) & C\left(\Im_6\right) & C\left(\Im_7\right) & C\left(\Im_8\right) & \cdots & C\left(\Im_{12}\right) \\
C\left(\Im_9\right) & C\left(\Im_{10}\right) & C\left(\Im_{11}\right) & C\left(\Im_{12}\right) & \cdots & C\left(\Im_{16}\right) \\
\vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\
C\left(\Im_{s-11}\right) & C\left(\Im_{s-10}\right) & C\left(\Im_{s-9}\right) & C\left(\Im_{s-8}\right) & \cdots & C\left(\Im_{s-4}\right) \\
C\left(\Im_{s-7}\right) & C\left(\Im_{s-6}\right) & C\left(\Im_{s-5}\right) & C\left(\Im_{s-4}\right) & \cdots & C\left(\Im_s\right)
\end{bmatrix} \tag{12}$$

In the matrix, we assume that $s = 4k$ and k is a nature number. If s does not meet the conditions, we can pad zero to satisfying the conditions. To simplify the discussion, we let $s = 4k$ and k is a nature number. Each row in the packets matrix M_p corresponds to a 32-bits sequence. The packets matrix M_p can be denoted as the following:

$$M_p = \begin{bmatrix} M_p\,[\![1]\!] \\ M_p\,[\![5]\!] \\ M_p\,[\![9]\!] \\ \vdots \\ M_p\,[\![s-11]\!] \\ M_p\,[\![s-7]\!] \end{bmatrix} \tag{13}$$

Based on the quantum hashing algorithm proposed in the lecture [15], we construct a set of quantum hash cluster Ω_h as the following:

$$\Omega_h = \left\{ \left| h_K^{[1]}(\cdot) \right\rangle, \left| h_K^{[2]}(\cdot) \right\rangle, \cdots, \left| h_K^{[p]}(\cdot) \right\rangle \right\} \tag{14}$$

And then each row of the packets matrix M_p is calculated utilizing the set of quantum hash cluster as the following:

$$\begin{aligned}
\Omega_h\,(1) &= \left\{ \left| h_K^{[1]}(M_p\,[\![1]\!]) \right\rangle, \left| h_K^{[2]}(M_p\,[\![1]\!]) \right\rangle, \cdots, \left| h_K^{[p]}(M_p\,[\![1]\!]) \right\rangle \right\} \\
\Omega_h\,(5) &= \left\{ \left| h_K^{[1]}(M_p\,[\![5]\!]) \right\rangle, \left| h_K^{[2]}(M_p\,[\![5]\!]) \right\rangle, \cdots, \left| h_K^{[p]}(M_p\,[\![5]\!]) \right\rangle \right\} \\
\Omega_h\,(9) &= \left\{ \left| h_K^{[1]}(M_p\,[\![9]\!]) \right\rangle, \left| h_K^{[2]}(M_p\,[\![9]\!]) \right\rangle, \cdots, \left| h_K^{[p]}(M_p\,[\![9]\!]) \right\rangle \right\} \\
&\cdots \\
\Omega_h\,(s-11) &= \left\{ \left| h_K^{[1]}(M_p\,[\![s-11]\!]) \right\rangle, \left| h_K^{[2]}(M_p\,[\![s-11]\!]) \right\rangle, \cdots, \left| h_K^{[p]}(M_p\,[\![s-11]\!]) \right\rangle \right\} \\
\Omega_h\,(s-7) &= \left\{ \left| h_K^{[1]}(M_p\,[\![s-7]\!]) \right\rangle, \left| h_K^{[2]}(M_p\,[\![s-7]\!]) \right\rangle, \cdots, \left| h_K^{[p]}(M_p\,[\![s-7]\!]) \right\rangle \right\}
\end{aligned} \tag{15}$$

According with the sequence $(\Omega_h\,(1), \Omega_h\,(5), \Omega_h\,(9), \cdots, \Omega_h\,(s-11), \Omega_h\,(s-7))$, we can identify subtle changes in these quantum sequences from other irrelevant quantum data.

3.3. **Qudits similarity comparison.** Motivated by classical similarity preserving hashing algorithms and Bloom filters methods [13,16], we construct the method of similarity comparison between these qudits information sequences in our proposed schemes. Based on these differences of corresponding bit, the similarity can be quantified.

Giving two qudits sequence $|\phi\rangle_t = |\vartheta_1\rangle\,|\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$ and $|\varphi\rangle_t = |\omega_1\rangle\,|\omega_2\rangle \cdots |\omega_i\rangle \cdots |\omega_t\rangle$, where $i = 1, 2, \cdots, t$, $\vartheta_i, \omega_i \in \{0, 1, 2, 3, \cdots, 7\}$. Each qudits in these two qudits sequence $|\phi\rangle_t = |\vartheta_1\rangle\,|\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$ and $|\varphi\rangle_t = |\omega_1\rangle\,|\omega_2\rangle \cdots |\omega_i\rangle \cdots |\omega_t\rangle$ are scanned. Providing the first qudit in $|\phi\rangle_t$ and $|\varphi\rangle_t$ are the same, that is $|\vartheta_1\rangle = |\omega_1\rangle$. Label these locations of these qudits $\left|\vartheta_{[s]}\right\rangle$ and $\left|\omega_{[r]}\right\rangle$, where $2 \le s, r \le t$, and $\left|\vartheta_{[s]}\right\rangle = |\vartheta_1\rangle\,, \left|\omega_{[r]}\right\rangle = |\omega_1\rangle$.

Based on these qudits $\left|\vartheta_{[s]}\right\rangle$ and $\left|\omega_{[r]}\right\rangle$, the qudits sequence $|\phi\rangle_t = |\vartheta_1\rangle\,|\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$ and $|\varphi\rangle_t = |\omega_1\rangle\,|\omega_2\rangle \cdots |\omega_i\rangle \cdots |\omega_t\rangle$ can be partitioned into $\Im_1, \Im_2, \cdots, \Im_s$ and $\aleph_1, \aleph_2, \cdots, \aleph_r$.

Compression algorithms are performed based on these formulae

$$\begin{aligned}
C\,(\Im_h) &= |\left(\sum_{r=[h-1]+1}^{[h]} \vartheta_r \right) \bmod (8)\rangle \\
C\,(\aleph_k) &= |\left(\sum_{r=[\![k-1]\!]+1}^{[\![k]\!]} \omega_r \right) \bmod (8)\rangle
\end{aligned} \tag{16}$$

We can obtain two compressed qudits sequences

$$\begin{array}{c}[C(\Im_1), C(\Im_2), \cdots, C(\Im_s)]\\ [C(\aleph_1), C(\aleph_2), \cdots, C(\aleph_r)]\end{array} \tag{17}$$

Furthermore, we can construct packets matrices $M_\Im$ and $M_\aleph$

$$\begin{array}{c} M_\Im = \begin{pmatrix} C(\Im_1) & C(\Im_2) & \cdots & C(\Im_8)\\ C(\Im_5) & C(\Im_6) & \cdots & C(\Im_{12})\\ \vdots & \vdots & \ddots & \vdots\\ C(\Im_{s-7}) & C(\Im_{s-6}) & \cdots & C(\Im_s)\end{pmatrix}\\ M_\aleph = \begin{pmatrix} C(\aleph_1) & C(\aleph_2) & \cdots & C(\aleph_8)\\ C(\aleph_5) & C(\aleph_6) & \cdots & C(\aleph_{12})\\ \vdots & \vdots & \ddots & \vdots\\ C(\aleph_{r-7}) & C(\aleph_{r-6}) & \cdots & C(\aleph_r)\end{pmatrix}\end{array} \tag{18}$$

Quantum hash cluster can be obtained as $\Omega_\Im$ and $\Omega_\aleph$. The quantum hash cluster $\Omega_\Im$ has the following formula:

$$\begin{array}{l}\Omega_\Im(1) = \left\{\left|h_K^{[1]}(M_\Im[\![1]\!])\right\rangle, \left|h_K^{[2]}(M_\Im[\![1]\!])\right\rangle, \cdots, \left|h_K^{[p]}(M_\Im[\![1]\!])\right\rangle\right\}\\ \Omega_\Im(5) = \left\{\left|h_K^{[1]}(M_\Im[\![5]\!])\right\rangle, \left|h_K^{[2]}(M_\Im[\![5]\!])\right\rangle, \cdots, \left|h_K^{[p]}(M_\Im[\![5]\!])\right\rangle\right\}\\ \cdots\\ \Omega_\Im(s-7) = \left\{\left|h_K^{[1]}(M_\Im[\![s-7]\!])\right\rangle, \left|h_K^{[2]}(M_\Im[\![s-7]\!])\right\rangle, \cdots, \left|h_K^{[p]}(M_\Im[\![s-7]\!])\right\rangle\right\}\end{array} \tag{19}$$

The quantum hash cluster $\Omega_\aleph$ has the following formula:

$$\begin{array}{l}\Omega_\aleph(1) = \left\{\left|h_K^{[1]}(M_\aleph[\![1]\!])\right\rangle, \left|h_K^{[2]}(M_\aleph[\![1]\!])\right\rangle, \cdots, \left|h_K^{[p]}(M_\aleph[\![1]\!])\right\rangle\right\}\\ \Omega_\aleph(5) = \left\{\left|h_K^{[1]}(M_\aleph[\![5]\!])\right\rangle, \left|h_K^{[2]}(M_\aleph[\![5]\!])\right\rangle, \cdots, \left|h_K^{[p]}(M_\aleph[\![5]\!])\right\rangle\right\}\\ \cdots\\ \Omega_\aleph(r-7) = \left\{\left|h_K^{[1]}(M_\aleph[\![r-7]\!])\right\rangle, \left|h_K^{[2]}(M_\aleph[\![r-7]\!])\right\rangle, \cdots, \left|h_K^{[p]}(M_\aleph[\![r-7]\!])\right\rangle\right\}\end{array} \tag{20}$$

These two quantum hash clusters $\Omega_\Im$ and $\Omega_\aleph$ constitute sequence as the following:

$$\begin{array}{c}\Omega_\Im = \{\Omega_\Im(1), \Omega_\Im(5), \Omega_\Im(9), \cdots, \Omega_\Im(s-7)\}\\ \Omega_\aleph = \{\Omega_\aleph(1), \Omega_\aleph(5), \Omega_\aleph(9), \cdots, \Omega_\aleph(s-7)\}\end{array} \tag{21}$$

Based on the characteristic of quantum hash cluster Ω_h, each item in $\Omega_\Im$ and $\Omega_\aleph$ stands for a qudit. Therefore, these two quantum hash clusters $\Omega_\Im$ and $\Omega_\aleph$ stand for $\frac{s}{4}$-qudit sequences and $\frac{r}{4}$-qudit sequences respectively.

The similarity measurement can be presented by the generalized quantum distance between $\Omega_\Im$ and $\Omega_\aleph$. The generalized quantum distance can be denoted as follows [11,17-19]:

$$\frac{\sum\limits_{u=1}^{(s/4)-1}[\Omega_\Im(4u-3) - \Omega_\aleph(4u-3)] \bmod 8}{\|\Omega_\Im\|_\diamondsuit + \|\Omega_\aleph\|_\diamondsuit} \tag{22}$$

where $\|\cdot\|_\diamondsuit$ stands for diamond norm [17-19].

$$\begin{array}{c}\|\Omega_\Im\|_\diamondsuit = \left\|\Omega_\Im \otimes I_{L(F)}\right\|_{tr}\\ \|\Omega_\aleph\|_\diamondsuit = \left\|\Omega_\aleph \otimes I_{L(F)}\right\|_{tr}\end{array} \tag{23}$$

The smaller generalized quantum distance is, the more similarity are these two quantum hash clusters $\Omega_\Im$ and $\Omega_\aleph$.

4. **Performance Analysis.** Those classical fuzzy hash and similarity preserving hashing algorithm cannot against the attack which comes from the quantum algorithm [11]. The security of those classic algorithms is insecurity. In this paper, we construct a quantum fuzzy similarity hashing algorithm based on the qudits sequence compression scheme. Our proposed methods can effectively against the quantum algorithms attack.

Give $|\phi\rangle_t = |\vartheta_1\rangle\, |\vartheta_2\rangle \cdots |\vartheta_i\rangle \cdots |\vartheta_t\rangle$, where $i = 1, 2, \cdots, t$, $\vartheta_i \in \{0, 1, 2, 3, \cdots, 7\}$. Let $t = 1000$, and then the qudits sequence has 1000 qudit. This qudits sequence can be visually represented in Figure 1.

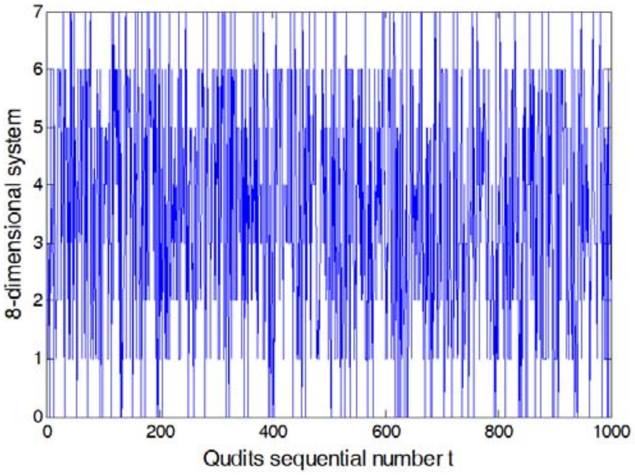

Figure 1. Visually represented of qudits sequence

The abscissa represents the qudits sequential number, and the ordinate represents qudit where $d = 8$. This qudits sequence is compressed based on the method of qudits sequence compression algorithms that is proposed in the Subsection 3.1. The result is illustrated in Figure 2.

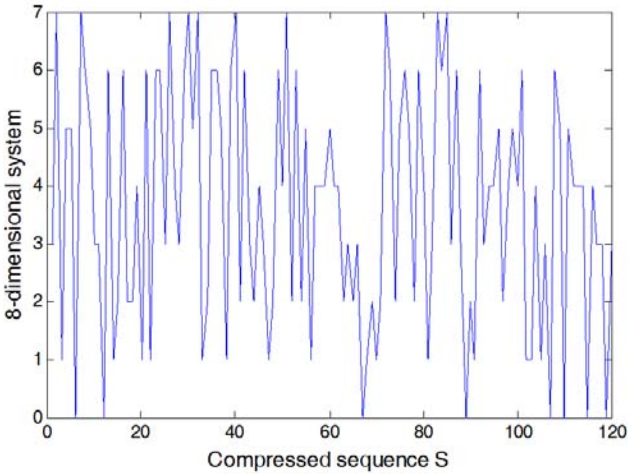

Figure 2. Compressed qudits sequence representation

The abscissa in Figure 2 represents the compressed qudits sequential number s, and the ordinate represents 8-dimensional qudit system.

As can be seen from the comparison of these two figures, the compression rate of the 8-dimensional qudit system is approximately eight times.

5. **Conclusion.** Based on the special characteristic of classical similarity preserving hashing schemes and qudits sequence compression methods for segmentation, we propose a novel construction method of quantum fuzzy similarity hash. Utilizing construction methods of classical similarity preserving hash, this type of quantum fuzzy similarity hash proposed in this paper can efficiently solve the problem that identifies identical or similarity qudits information. Performance analysis proves that this quantum fuzzy similarity hash schemes can resist the quantum algorithms attack.

Acknowledgment. This work is partially supported by the National Natural Science Foundation of China (No. 61071145, No. 61271331), the Specialized Research Fund for the Doctoral Program of Higher Education (200802880014), the China Postdoctoral Science Foundation (2013M541671) and Jiangsu Province Postdoctoral Science Foundation (1302076B). The authors also gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] V. Roussev, Data fingerprinting with similarity digests, in *Advances in Digital Forensics VI, ser. IFIP Advances in Information and Communication Technology*, K.-P. Chow and S. Shenoi (eds.), Boston, Springer, 2010.

[2] F. Breitinger and H. Baier, Similarity preserving hashing: Eligible properties and a new algorithm MRSH-v2, *The 4th ICST Conference on Digital Forensics & Cyber Crime*, 2012.

[3] J. Kornblum, Identifying almost identical files using context triggered piecewise hashing, *Digital Investigation*, pp.91-97, 2006.

[4] N. Harbour and K. Jesse, *dcfldd*, http://sourceforge.net/projects/dcfldd/?source=directory, 2013.

[5] H. Baier and F. Breitinger, Security aspects of piecewise hashing in computer forensics, *The 6th International Conference on IT Security Incident Management and IT Forensics*, Stuttgart, Germany, pp.21-36, 2011.

[6] K. Jesse, *ssdeep*, http://sourceforge.net/projects/ssdeep/?source=directory, 2013.

[7] F. Breitinger and H. Baier, A fuzzy hashing approach based on random sequences and hamming distance, *ADFSL Conference on Digital Forensics, Security and Law*, pp.89-101, 2012.

[8] Y. P. Huang and T. W. Chang, An efficient fuzzy hashing model for image retrieval, *Fuzzy Information Processing Society*, Montreal, Canada, pp.223-228, 2006.

[9] T. B. J. Andrew and J. Kim, Fuzzy hash: A secure biometric template protection technique, *Frontiers in the Convergence of Bioscience and Information Technologies*, Jeju City, Korea, pp.688-694, 2007.

[10] K. Marshall, J. Rosenthal, D. Schipani and A. L. Trautmann, *Subspace Fuzzy Vault*, http://arxiv.org/abs/1210.7190, 2013.

[11] F. Breitinger, K. P. Astebol, H. Baier and C. Busch, mvHash-B – A new approach for similarity preserving hashing, *The 7th International Conference on IT Security Incident Management and IT Forensics*, 2013.

[12] D. Lemire and K. Owen, Recursive n-gram hashing is pairwise independent, at best, *Computer Speech and Language*, vol.24, no.4, pp.698-710, 2010.

[13] P. Rungta, W. J. Munro, K. Nemoto, P. Deuar, G. J. Milburn and C. M. Caves, *Qudit Entanglement*, http://arxiv.org/pdf/quant-ph/0001075v2.pdf, 2013.

[14] K. Brennen, D. P. Leary and S. S. Bullock, Criteria for exact qudit universality, *Physical Review A*, vol.71, no.5, 2005.

[15] F. Ablayev and A. Vasiliev, *Quantum Hashing*, http://arxiv.org/abs/1310.4922, 2013.

[16] B. H. Bloom, Space/time trade-offs in hash coding with allowable errors, *Commun. of the ACM*, vol.13, no.7, pp.422-426, 1970.

[17] J. Watrous, *Quantum Computational Complexity*, http://http://arxiv.org/abs/0804.3401, 2003.

[18] A. Kitaev, Quantum computations: Algorithms and error correction, *Russian Mathematical Surveys*, vol.52, no.6, pp.1191-1249, 1997.

[19] A. Kitaev, A. Shen and M. Vyalyi, *Classical and Quantum Computation (Graduate Studies in Mathematics)*, American Mathematical Society, 2002.