IEEE Transactions on Quantum Engineering

Received 11 August 2023; revised 16 February 2024; accepted 17 February 2024; date of publication 21 February 2024; date of current version 25 March 2024.

Digital Object Identifier 10.1109/TQE.2024.3368073

A Stable Hash Function Based on Parity-Dependent Quantum Walks With Memory (August 2023)

QING ZHOU, XUEMING TANG, SONGFENG LU, AND HAO YANG

Hubei Key Laboratory of Distributed System Security, Hubei Engineering Research Center on Big Data Security, School of Cyber Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

Corresponding authors: Xueming Tang; Songfeng Lu (e-mail: xmtang@hust.edu.cn; lusongfeng@hotmail.com).

This work was supported in part by the National Natural Science Foundation of China under Grant 62101197 and in part by the China Postdoctoral Science Foundation under Grant 2021M691148.

ABSTRACT In this article, we develop a generic controlled alternate quantum walk model by combining parity-dependent quantum walks with distinct arbitrary memory lengths and propose a hash function (called QHFM-P) based on this model. The statistical properties of the proposed scheme are stable with respect to the coin parameters of the underlying controlled quantum walks, and with certain parameter values, the collision resistance property of QHFM-P is better than that of the state-of-the-art hash functions based on discrete quantum walks. Moreover, the proposed hash function can also maintain near-ideal statistical performance when the input message is of small length. In addition, we derive a type of inappropriate initial states of hash functions based on 1-D one-particle quantum walks (with ordinary shift operator) on cycles, with which all messages will be mapped to the same hash value, regardless of the angles adopted by the coin parameters.

INDEX TERMS Controlled alternate quantum walks, hash function, quantum walks with memory (QWM), stability analysis, statistical properties.

I. INTRODUCTION

Cryptographic hash functions not only act as key components of identification, message authentication, digital signatures, and pseudorandom number generation in traditional cryptography, but also play important roles in postquantum cryptography [1]. From a security perspective, cryptographic hash functions can be divided into two broad categories: 1) provably secure hash functions based on hard mathematical problems and 2) dedicated hash functions based on ad hoc constructions, especially on iterative constructions of one-way compression functions. The former only satisfy computational security and are inefficient to be used in practice; the latter are efficiently computable, but the security of which is not built on a firm foundation.

To develop a secure and efficient hash function, more and more researchers have shown interests in hash functions based on quantum computing [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], especially on discrete quantum walks [2], [3], [6], [7], [8], [9], [10], [11], [12], [13] (hereafter, simply DQW-based hash functions). The security of this kind of hash functions is based on quantum mechanics; more precisely, it is ensured by the theoretically infinite possibilities of the initial state and the irreversibility of modulo operation [8].

Quantum walk-based hash functions are constructed by combining different DQW processes, performing the walk controlled by the input message and calculating the output hash from the resulting probability distribution of the walk, where the message bits determine which type of walking process is performed at each time step. In most DQW-based hash functions [6], [7], [8], [9], [10], [11], [12], [13], the coin operator is the only controlled component, while some other schemes let the shift operator [3] or an additional operator together with the coin operator [2] be the changeable part. By choosing appropriate parameter values or suitable transforms for the controlled components, one can obtain a good hash function with near-ideal statistical properties. However, it has not yet been answered that whether DQW-based hash function can behave equally well when adopting different parameter values or different additional operators.

Quantum walks with memory (QWM) [14], [15], [16], [17], [18], [19], [20], [21], [22], [23] are types of modified quantum walks where the next direction of the walking particle is governed by the direction-determine operator, which

specifies how the latest path together with the coin state affects the movement direction of the particle; different QWM models have different direction-determine operators. Unlike usual quantum walks without memory [25] or lively quantum walks [3], where the changeable operator (the coin or the shift operator) can be characterized by a single parameter, the direction-determine operator of QWM is highly flexible. It is influenced by two factors: the memory length and the movement rule. The latter can be an arbitrary relation between the movement direction and the recorded shifts of the walker, which cannot simply be dictated by a few numerical parameters. Hence, compared with hash functions based on quantum walks without memory, a QWM-based hashing scheme [2] involves more variables that may influence its statistical properties, which makes it a good subject for studying the stability of QW-based hash functions with respect to their variable components.

As a preliminary exploration of the stability of QWM-based hash function with respect to direction-determine transform, we replace the underlying quantum walks with two-step memory [15] of Zhou and Lu's [2] hashing scheme (called QHFM) by quantum walks with two-step memory depending on the parity of memory [16] and test the statistical properties of the modified hash function (called QHFM-P). The test result shows, that with the same parameter values, the diffusion and confusion properties, the uniform distribution property, and collision resistance property of QHFM-P are comparable with the corresponding properties of QHFM, and the sensitivity of hash value to message of QHFM-P is generally better than that of QHFM. The good performance of QHFM-P suggests that a QWM-based hash function could be stable with respect to the direction-determine transform, and that adjusting the movement rule of the direction-determine transform could be useful for improving the hash properties. It also implies that QWM have great potential to be used to construct good hash functions. Motivated by this result, we develop a generic controlled quantum walk model (CQWM-P) by combining parity-dependent quantum walks with distinct arbitrary memory lengths, based on which one can construct various hash functions using QWM with different memory lengths.

Among parameters of DQW-based hash functions, the two (or more) coin parameters of the underlying quantum walks are crucial components. In order to explore the stability of QWM-based hash function with respect to different coin angles, we uniformly choose hundreds of pairs of angles from the plane of the two coin parameters of QHFM-P and numerically test each type of hash properties for each pair of angles. We observe that QHFM-P can preserve near-ideal statistical performance when changing the values of its coin parameters, which suggests that QWM-based hash function has nice stability with respect to different coin angles.

During the exploration of the stability of QHFM-P with respect to coin parameters, we recognize that certain initial states lead to more frequent collisions when the underlying quantum walks both adopt balanced Hadamard coin. After theoretical analysis of the evolution of the underlying CQWM-P using Konno's approach [23], [24], we derive a type of inappropriate initial states that makes QHFM-P map all messages to the same hash value, regardless of the values of the coin parameters and the movement rule of the direction-determine transform. Moreover, such a result also applies to hash functions [7], [8], [9] using 1-D one-particle quantum walks with ordinary shift transform and without memory. This indicates that the initial state is also an important factor affecting the security of DQW-based hash functions, and one should not use inappropriate initial states when constructing hash functions using controlled alternate quantum walks. Using the same approach, one can obtain other inappropriate initial states that lead a set of different messages to the same hash value for a specific coin (as well as a given direction-determine transform) adopted by DQW-based hash functions. To provide an initial insight into this issue, we derive the conditions for the initial state that make QHFM-P map messages differing only in the first two bits to the same hash result.

Apart from the stability issue, existing hashing schemes based on discrete quantum walks only consider long messages and do not apply to short messages. For example, if the walker starts at a single position on a cycle with n nodes (n is odd) and performs t steps of 1-D (ordinary) walks controlled by a t-bit message with $t < n - 1$, then the amplitudes from the initial position cannot "spread" to all n nodes; accordingly, the hash bits contributed by the node(s) where the amplitudes remain 0 must be a fixed value, regardless the content of the message. As a result, the hash values of two messages of short length must collide at some positions that can be derived from the length(s) of the messages and the number of nodes. To solve this problem, we add a preprocessing and a postprocessing steps to QHFM-P for short messages. With these steps, the modified scheme is able to maintain near-ideal statistical performance when the length of input message is very small. Moreover, with the preprocessing step, the revised scheme achieves better collision resistance property for ordinary long messages.

As a result, we propose a stable hash function QHFM-P based on QWM, which can preserve near-ideal statistical properties when the coin parameters take different values and when the message is of short length. The main contribution and novelty of this work includes the following.

1) We develop a generic CQWM-P by combining parity-dependent quantum walks with distinct arbitrary memory lengths.
2) We propose a hash function QHFM-P based on CQWM-P by adding a preprocessing step before performing CQWM-P controlled by the input message and appending a postprocessing step for short messages.
3) With some certain coin angles, QHFM-P achieves better collision resistance property than the reported results of the state-of-the-art hash functions.

4) With the pre and postprocessing steps, QHFM-P can preserve near-ideal statistical properties when the message is of short length.
5) The statistical properties of QHFM-P are stable with respect to the two coin parameters.
6) We derive a type of inappropriate initial states of hash functions based on one-particle one-coin quantum walks with or without memory, with which all messages will be mapped to the same hash value, regardless of the values of the coin parameters and the movement rule of the direction-determine transform.

II. CONTROLLED QWM DEPENDING ON THE PARITY OF MEMORY

The 1-D quantum walks with r-step memory depending on the parity of memory (QWM-P), or parity-dependent quantum walks with r-step memory on the line [16], are the quantum system living in a Hilbert space $\mathcal{H}_p \otimes \mathcal{H}_{d_r} \otimes \cdots \otimes \mathcal{H}_{d_2} \otimes \mathcal{H}_{d_1} \otimes \mathcal{H}_c$ spanned by orthogonal basis states $\{|x, d_r, \ldots, d_2, d_1, c\rangle\, |d_r, \ldots, d_1, c \in \mathbb{Z}_2; x \in \mathbb{Z}\}$, where c is the coin state, d_j (0 stands for left and 1 stands for right) records the shift of the walker j steps before (d_1 is the direction of the most recent step, and d_r is the earliest direction that is memorized), and x is the current position of the walker. If the walker moves on a cycle with n nodes, then $x \in \mathbb{Z}_n$. The one-step evolution of QWM-P may be decomposed into three parts. The first is a 2×2 coin operator C on subspace $\mathcal{H}_c$, here C is parameterized by an angle θ ($\theta \in (0, \pi/2)$), i.e.,

$$C = \begin{pmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{pmatrix} \tag{1}$$

the second is the direction-determine transform D on $\mathcal{H}_{d_r} \otimes \cdots \otimes \mathcal{H}_{d_2} \otimes \mathcal{H}_{d_1} \otimes \mathcal{H}_c$, whose action can be written as

$$\begin{aligned} \hat{D}:\ & |d_r, d_{r-1}, \ldots, d_2, d_1, c\rangle \\ & \to |d_{r-1}, d_{r-2}, \ldots, d_1, c \oplus 1 \oplus \text{if even}(d_r, \ldots, d_1), c\rangle \end{aligned} \tag{2}$$

where if even$(d_r, \ldots, d_1)$ equals 1 (respectively, 0) if the number of zeros in the memorized directions $d_r, \ldots, d_1$ is even (respectively, odd), and the third is the shift operator S on $\mathcal{H}_p \otimes \mathcal{H}_{d_1}$, whose action can be expressed as

$$S : |x, d_1\rangle \to |x + 2d_1 - 1, d_1\rangle\,. \tag{3}$$

If the walker moves on a cycle with n nodes, then the shift operator becomes $|x, d_1\rangle \to |x + 2d_1 - 1 (\mathrm{mod} n), d_1\rangle$.

A controlled 1-D QWM depending on the parity of memory (CQWM-P) can be obtained by alternately applying QWM-P with different memory lengths (as well as different coin parameters). More precisely, CQWM-P evolves according to a t-bit binary string msg $= (m_1, m_2, \ldots, m_t) \in \{0, 1\}^t$: at the jth time step, if $m_j = 0$ ($j \in \{1, 2, \ldots, t\}$), then the walker performs QWM-P with s_0 step memory (denoted by QW s_0 M-P) and with coin parameter θ_0; if $m_j = 1$, then the walker performs QWM-P with s_1 ($s_1 \neq s_0$) step memory (denoted by QW s_1 M-P) and with coin parameter θ_1.

To enable QW s_0 M-P and QW s_1 M-P to be performed alternately, $|s_0 - s_1|$ redundant qubits can be added to the walk process with less memory length, so that two walks live in the same Hilbert space. For instance, if $0 < s_0 < s_1$, then $s_1 - s_0$ redundant qubits are added to QWM s_0 P. In this case, the basis states of QWM s_0 P become $\{|x, d_{s_1}, \ldots, d_2, d_1, c\rangle\, |d_{s_1}, \ldots, d_1, d \in \mathbb{Z}_2; x \in \mathbb{Z}\}$, wherein the first $s_1 - s_0$ qubits are invariant under the transforms of QWM s_0 P, and the D transform becomes

$$\begin{aligned} D:\ & |d_{s_1}, \ldots, d_{s_0+1}, d_{s_0}, d_{s_0-1}, \ldots, d_2, d_1, c\rangle \\ & \to |d_{s_1}, \ldots, d_{s_0+1}, d_{s_0-1}, \ldots, d_1, c \oplus 1 \oplus \text{if even}(d_{s_0}, \ldots, d_1), c\rangle. \end{aligned} \tag{4}$$

In this work, we focus on CQWM-P with one- and two-step memory, whose evolution operator controlled by msg is the product of t unitary transforms

$$U_{\text{msg}} = U^{(m_t)} U^{(m_{t-1})} \cdots U^{(m_2)} U^{(m_1)} \tag{5}$$

where $U^{(m_j)}$ is the one-step transform defined as

$$U^{(m_j)} = S \cdot \left(I_n \otimes D^{(m_j)}\right) \cdot \left(I_{4n} \otimes C^{(m_j)}\right). \tag{6}$$

In (6), $C^{(0)}$ and $C^{(1)}$ are coin operators parameterized by θ_0 and θ_1, respectively; I_{4n} and I_n are $4n \times 4n$ and $n \times n$ identity operators, respectively; S is the conditional shift operator controlled by the next direction d_1, such a direction is determined by an 8×8 unitary operator $D^{(m_j)}$. If $m_j = 0$, then $D^{(m_j)}$ is the direction-determine transform of QW1M-P, i.e., $\hat{D}^{(0)} : |d_1, c\rangle \to |c \oplus 1 \oplus d_1, c\rangle$, and if $m_j = 1$, then $D^{(m_j)}$ is the direction-determine transform of QW2M-P, i.e., $D^{(1)} : |d_2, d_1, c\rangle \to |d_1, c \oplus (d_1 \oplus d_2), c\rangle$. By appending a redundant qubit d_2 to the state of QW1M-P and letting $D^{(0)} : |d_2, d_1, c\rangle \to |d_2, c \oplus 1 \oplus d_1, c\rangle$ determines the next direction if the controlling bit equals 0, the walk process can switch freely between QW1M-P and QW2M-P.

III. ON THE INITIAL STATES OF CQWM-P WITH ONE- AND TWO-STEP MEMORY

In this section, we derive several conditions for the initial states of CQWM-P such that the resulting probability distributions corresponding to different input messages are the same, which should be taken into account when constructing hash functions based on CQWM-P.

Following the authors in [23] and [24], we decompose the coin operator together with the direction-determine transform $D^{(s)} \cdot [I_4 \otimes C^{(s)}]$ (i.e., the time evolution excluding the shift transform) of $QW(s+1)M - P$ ($s \in \{0, 1\}$) into a "left" part P_s and a "right" part Q_s. Using P_s and Q_s, the one-step time evolution of QWsM-P on a cycle of length n can be

TABLE 1. Actions of the Coin, the Direction-Determine, and the Complete One-Step Transform of QW2M-P

$\|x,j\rangle$	$I_{4n}\otimes C^{(1)}\|x,j\rangle$	$(I_n\otimes D^{(1)})(I_{4n}\otimes C^{(1)})\|x,j\rangle$	$S(I_n\otimes D^{(1)})(I_{4n}\otimes C^{(1)})\|x,j\rangle$
$\|x,000\rangle$	$a\|x,000\rangle+c\|x,001\rangle$	$a\|x,000\rangle+c\|x,011\rangle$	$a\|x-1,0\rangle+c\|x+1,3\rangle$
$\|x,001\rangle$	$b\|x,000\rangle+d\|x,001\rangle$	$b\|x,000\rangle+d\|x,011\rangle$	$b\|x-1,0\rangle+d\|x+1,3\rangle$
$\|x,010\rangle$	$a\|x,010\rangle+c\|x,011\rangle$	$a\|x,110\rangle+c\|x,101\rangle$	$a\|x+1,6\rangle+c\|x-1,5\rangle$
$\|x,011\rangle$	$b\|x,010\rangle+d\|x,011\rangle$	$b\|x,110\rangle+d\|x,101\rangle$	$b\|x+1,6\rangle+d\|x-1,5\rangle$
$\|x,100\rangle$	$a\|x,100\rangle+c\|x,101\rangle$	$a\|x,010\rangle+c\|x,001\rangle$	$a\|x+1,2\rangle+c\|x-1,1\rangle$
$\|x,101\rangle$	$b\|x,100\rangle+d\|x,101\rangle$	$b\|x,010\rangle+d\|x,001\rangle$	$b\|x+1,2\rangle+d\|x-1,1\rangle$
$\|x,110\rangle$	$a\|x,110\rangle+c\|x,111\rangle$	$a\|x,100\rangle+c\|x,111\rangle$	$a\|x-1,4\rangle+c\|x+1,7\rangle$
$\|x,111\rangle$	$b\|x,110\rangle+d\|x,111\rangle$	$b\|x,100\rangle+d\|x,111\rangle$	$b\|x-1,4\rangle+d\|x+1,7\rangle$

expressed as an $n\times n$ block matrix

$$U_n^{(s)}=\begin{bmatrix} O & P_s & O & O & \cdots & O & Q_s\\ Q_s & O & P_s & O & \cdots & O & O\\ O & Q_s & O & P_s & \cdots & O & O\\ O & O & Q_s & O & \cdots & O & O\\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ O & O & O & O & \cdots & O & P_s\\ P_s & O & O & O & \cdots & Q_s & O \end{bmatrix} \tag{7}$$

where the item on the jth row and the $[(j+1) \bmod n]$th column is the 8×8 matrix P_s, the item on the $[(j+1) \bmod n]$th row and the jth column is the 8×8 matrix Q_s ($j\in\{0,1,\ldots,n-1\}$), and the other items are zero matrix O of order 8×8.

Let $\Psi_t(x)=[A_t^{x,0},A_t^{x,1},\ldots,A_t^{x,7}]^T$ (with $A_t^{x,j}\in\mathbb{C}$, $x\in\mathbb{Z}_n$, $j\in\mathbb{Z}_8$, and T denotes the transpose operator) be the amplitudes of walker at position x and at time t, arranging the amplitudes at all positions as an $8n\times 1$ vector

$$\begin{aligned}\Psi_t &= [\Psi_t(0),\Psi_t(1),\ldots,\Psi_t(n-1)]^T\\ &= \begin{bmatrix}\begin{bmatrix}A_t^{0,0}\\A_t^{0,1}\\\vdots\\A_t^{0,7}\end{bmatrix},\begin{bmatrix}A_t^{1,0}\\A_t^{1,1}\\\vdots\\A_t^{1,7}\end{bmatrix},\ldots,\begin{bmatrix}A_t^{n-1,0}\\A_t^{n-1,1}\\\vdots\\A_t^{n-1,7}\end{bmatrix}\end{bmatrix}^T\end{aligned} \tag{8}$$

the time evolution of CQWM-P corresponding to the message msg can be expressed as

$$\Psi_t=U_n^{(m_t)}U_n^{(m_{t-1})}\cdots U_n^{(m_2)}U_n^{(m_1)}\Psi_0 \tag{9}$$

where Ψ_0 denotes the initial amplitudes.

A. P AND Q MATRICES OF QW1M-P AND QW2M-P

The matrices P_0, Q_0, P_1, and Q_1 can be obtained by picking out the rows corresponding to the left and right movers from $D^{(0)}\cdot[I_4\otimes C^{(0)}]$ and $D^{(1)}\cdot[I_4\otimes C^{(1)}]$, respectively. For ease of notation, we denote $C^{(1)}$ by $\begin{bmatrix}a & b\\ c & d\end{bmatrix}$ and write $(x\pm 1) \bmod n$ shortly as $x\pm 1$. To obtain the matrix form of $D^{(1)}\cdot[I_4\otimes C^{(1)}]$, we list the results of sequentially performing the coin operator, the direction-determine transform, and the shift operator of QW2M-P on the computational basis states $|x,j\rangle$ in Table 1, where the second terms j of the states in the first three columns are written in a binary format $j_2j_1j_0$ ($j_2,j_1,j_0\in\{0,1\}$).

According to the third column of Table 1, one can get the matrix of the coin together with the direction-determine transform of QW2M-P

$$D^{(1)}(I_{4n}\otimes C^{(1)})=\begin{bmatrix} a & b & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & c & d & 0 & 0\\ 0 & 0 & 0 & 0 & a & b & 0 & 0\\ c & d & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & a & b\\ 0 & 0 & 0 & 0 & c & d & 0 & 0\\ 0 & 0 & 0 & 0 & a & b & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & c & d \end{bmatrix}. \tag{10}$$

After the direction-determine transform, the walker in state $|x,j_2j_1j_0\rangle$ moves according to the value of j_1: if $j_1=0$, then the walker moves left and the state becomes $|x-1 \bmod n),j_2j_1j_0\rangle$, and if $j_1=1$, the walker moves right and the state becomes $|x+1 \bmod n),j_2j_1j_0\rangle$. Thus, states $|x,000\rangle$, $|x,001\rangle$, $|x,100\rangle$, and $|x,101\rangle$ indicate a left mover, whereas $|x,010\rangle$, $|x,011\rangle$, $|x,110\rangle$, and $|x,111\rangle$ indicate a right mover. Picking out the first, second, fifth, and sixth rows (corresponding to states $|x,j_20j_1\rangle$) from $D^{(1)}(I_{4n}\otimes C^{(1)})$, one gets the "left" part of the coin together with the direction-determine transform

$$P_1=\begin{bmatrix} a & b & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & c & d & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & a & b\\ 0 & 0 & 0 & 0 & c & d & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{11}$$

The "right" part of the coin together with the direction-determine transform is obtained by picking out the third, fourth, seventh, and eight rows (corresponding to states

$|x, j_2 1 j_1\rangle)$ from $D^{(1)}(I_{4n} \otimes C^{(1)})$

$$Q_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & b & 0 & 0 \\ c & d & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a & b & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c & d \end{bmatrix}. \tag{12}$$

Similarly, one can get P_0 and Q_0 by picking out the rows of the matrix form of $D^{(0)}(I_{4n} \otimes C^{(0)})$ corresponding to states $|x, j_2 0 j_1\rangle$ and $|x, j_2 1 j_1\rangle$, respectively.

B. INITIAL STATES LEADING DIFFERENT MESSAGES TO THE SAME DISTRIBUTION

The expression of $U_n^{(m_t)} U_n^{(m_{t-1})} \cdots U_n^{(m_1)}$ can be simplified and clarified by using the sum $\Xi_t(l, r)$ of all paths in the trajectory of the walker consisting of l steps to the left and r steps to the right at time t (with $l + r = t$)

$$\Xi_t(l, r) = \sum_{\substack{l_j, r_j \in \{0,1\} \\ l_j \oplus r_j = 1 \\ l_1 + \cdots + l_t = l \\ r_1 + \cdots + r_t = r}} P_{m_t}^{l_t} Q_{m_t}^{r_t} P_{m_{t-1}}^{l_{t-1}} Q_{m_{t-1}}^{r_{t-1}} \cdots P_{m_1}^{l_1} Q_{m_1}^{r_1} \tag{13}$$

and the three equations for $\Xi_t(l, r)$

$$P_{m_{t+1}} \Xi_t(l, t-l) + Q_{m_{t+1}} \Xi_t(l+1, t-l-1) = \Xi_{t+1}(l+1, t-l) \tag{14}$$

$$P_{m_t} \Xi_t(l, 0) = \Xi_{t+1}(l+1, 0) \tag{15}$$

$$Q_{m_t} \Xi_t(0, r) = \Xi_{t+1}(0, r+1) \tag{16}$$

which can be proved using combinatorial approaches. For instance, the matrix of $U_5^{(m_3)} U_5^{(m_2)} U_5^{(m_1)}$ can be expressed in terms of $\Xi_3(l, r)$ with $l + r = 3$ and $l, r \in \{0, 1, 2, 3\}$

$$\begin{bmatrix} O & \Xi_3(2,1) & \Xi_3(0,3) & \Xi_3(3,0) & \Xi_3(1,2) \\ \Xi_3(1,2) & O & \Xi_3(2,1) & \Xi_3(0,3) & \Xi_3(3,0) \\ \Xi_3(3,0) & \Xi_3(1,2) & O & \Xi_3(2,1) & \Xi_3(0,3) \\ \Xi_3(0,3) & \Xi_3(3,0) & \Xi_3(1,2) & O & \Xi_3(2,1) \\ \Xi_3(2,1) & \Xi_3(0,3) & \Xi_3(3,0) & \Xi_3(1,2) & O \end{bmatrix}.$$

By matrix multiplications and (14)–(16), one can summarize the following characteristics of the matrix of $U_n^{(m_t)} \cdots U_n^{(m_1)}$.

c1) The components of the ith row ($i \in \{2, \ldots, n\}$) can be obtained through a right circular shift of the components of the first row by $i - 1$ positions, and the components of the ith column can be obtained through a the downside circular shift of the components of the first column by $i - 1$ positions.

c2) The sum of all components of every row equals the sum of all possible paths of the walker that performs CQWM at time t, i.e., $\sum_{l+r=t}^{l,r \in \mathbb{Z}_{t+1}} \Xi_t(l, r) = (P_{m_t} + Q_{m_t}) \cdots (P_{m_2} + Q_{m_2})(P_{m_1} + Q_{m_1})$, and the sum of all components of every column also equals $\sum_{l+r=t}^{l,r \in \mathbb{Z}_{t+1}} \Xi_t(l, r)$.

Since the matrix expressions using $\Xi_t(l, r)$ are independent of the movement rule specified by the transform other than the shift operator (i.e., the coin together with the direction-determine transform), the characteristics are applicable to all quantum walks on cycles whose time evolution (excluding the shift transform) can be decomposed into a "left" and a "right" part. As a result, we derive the following lemma for 1-D one-particle controlled quantum walks on cycles where the walker moves one step to the left or right at each time step.

Lemma 1: Suppose that a walker performs a controlled 1-D one-particle quantum walk on a cycle with n nodes (with n being odd and with ordinary shift transform) controlled by a binary message msg, if the initial amplitudes of the walker satisfy $\Psi_0(0) = \Psi_0(1) = \cdots = \Psi_0(n-1)$, then the probability of finding the walker being at an arbitrary node is $1/n$ at any time, regardless of the value of msg.

Proof: Let $\Psi_0(0) = \Psi_0(1) = \cdots = \Psi_0(n-1) = \varphi$ and $[\sum_{l+r=t}^{l,r \in \mathbb{Z}_{t+1}} \Xi_t(l, r)]\Psi_0(0) = \varphi'$, according to (9) and characteristic (c2), the amplitudes corresponding to message msg $= (m_1, m_2, \ldots, m_t)$ are

$$U_n^{(m_t)} \cdots U_n^{(m_1)} \Psi_0 = [\varphi', \varphi', \ldots, \varphi']^T$$

meaning that $\Psi_t(0) = \Psi_t(1) = \cdots = \Psi_t(n-1) = \sum_{l+r=t}^{l,r \in \mathbb{Z}_{t+1}} \Xi_t(l, r)\Psi_0(0)$. Since $\sum_{x=0}^{n-1} \|\Psi_t(x)\|^2 = 1$, the probability that the walker locates an arbitrary position is $1/n$, regardless of the value of $(m_1, m_2, \ldots, m_t)$. ■

In most DQW-based hash functions, the amplitudes of the initial states are concentrated at a single position. Using expression (7), one can also derive some improper initial states of this type for the hash function based on QW1M-P and QW2M-P such that different input messages lead to same output states if the underlying quantum walks both use the unbiased Hadamard coin operator. To see this, suppose $\Psi_0' = [\Psi_0'(0), \ldots, \Psi_0'(n-1)]^T$, $\Psi_0'(0) = \phi = [k_0, k_1, \ldots, k_7]^T$, and $\Psi_0'(1) = \cdots = \Psi_0'(n-1) = o = [0, 0, 0, 0, 0, 0, 0, 0]^T$ (with $\sum_j |k_j|^2 = 1$), the resulting amplitudes corresponding to a one-bit message m_1 can be calculated by

$$U_n^{(m_1)} \Psi_0' = [o, Q_{m_2}\phi, o, o, \ldots, P_{m_2}\phi]^T.$$

Let $C^{(0)} = C^{(1)} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$, the amplitudes of being at position 1 corresponding to messages "0" and "1" are

$$Q_0\phi = \frac{1}{\sqrt{2}}[0, 0, k_0 + k_1, k_2 - k_3, 0, 0, k_4 + k_5, k_6 - k_7]^T$$

and

$$Q_1\phi = \frac{1}{\sqrt{2}}[0, 0, k_4 + k_5, k_0 - k_1, 0, 0, k_2 + k_3, k_6 - k_7]^T$$

respectively, and the amplitudes of being at position $n-1$ corresponding to messages "0" and "1" are $P_0\phi =$

TABLE 2. Conditions Corresponding to Two-Bit Messages Such That the Resulting States Are Equivalent

(msg, msg′)	Conditions
("00," "01")	$k_0 = k_1 = k_2 = k_3 = 0$, $k_4 = k_5, k_6 = -k_7$
("01," "10")	$k_0 = k_1 = k_2 = k_3 = 0$, $k_4 = -k_5, k_6 = -k_7$
("10," "11")	$k_0 = k_1 = k_4 = k_5 = 0$, $k_2 = k_3, k_6 = -k_7$
("00," "11")	$k_0 = k_1 = k_2 = k_3 = k_4 = k_5 = 0, k_6 = -k_7$
("00," "10") and ("01," "11")	$k_0 = k_2 = k_4$, $k_1 = k_3 = k_5$

$(1/\sqrt{2})[k_2 + k_3, k_0 - k_1, 0, 0, k_6 + k_7, k_4 - k_5, 0, 0]^T$ and $Q_0\phi = (1/\sqrt{2})[k_0 + k_1, k_4 - k_5, 0, 0, k_6 + k_7, k_2 - k_3, 0, 0]^T$, respectively. Solving the equations $Q_0\phi = Q_1\phi$ and $P_0\phi = P_1\phi$, one gets

$$\begin{cases} k_0 = k_2 = k_4 \\ k_1 = k_3 = k_5. \end{cases} \tag{17}$$

Hence, if the initial amplitudes Ψ_0' satisfy (17), then the output amplitudes corresponding to different one-bit message are equivalent, which means two messages that differ only in the first bit have the same hash value.

For a two-bit message (m_1, m_2), the corresponding resulting amplitudes are $U_n^{(m_2)}U_n^{(m_1)}\Psi_0' = [(P_{m_2}Q_{m_1} + Q_{m_2}P_{m_1})\phi, o, Q_{m_2}Q_{m_1}\phi, o, \ldots, o, o, P_{m_2}P_{m_1}\phi, o]^T$. Solving the (linear) equations

$$\begin{cases} (P_0Q_0 + Q_0P_0)\phi = (P_1Q_0 + Q_1P_0) \\ (Q_0)^2\phi = Q_1Q_0\phi \\ (P_0)^2\phi = P_1P_0\phi \end{cases}$$

one get the condition for ϕ such that the messages "00" and "01" lead to the same quantum states

$$\begin{cases} k_0 = k_1 = k_2 = k_3 = 0 \\ k_6 = -k_7 \\ k_4 = k_5. \end{cases} \tag{18}$$

Therefore, if the initial amplitudes $\{k_j | j \in \mathbb{Z}_8\}$ at the starting position satisfy (18), then two messages that differ only in the first two bits (one start with "00" and the other start with "01") have the same hash value. In an analogous way, one can get the condition corresponding to other two-bit messages pairs. The conditions for Ψ_0' corresponding to two-bit message pair (msg, msg′) with msg $= (m_1, m_2) \in \{0, 1\}^2$ and msg′ $= (m_1', m_2') \in \{0, 1\}^2$ such that $U_n^{(m_2)}U_n^{(m_1)}\Psi_0' = U_n^{(m_2')}U_n^{(m_1')}\Psi_0'$ are listed in Table 2.

The condition corresponding to message pairs ("00,""10") and ("01,""11") is the same as that corresponding to the one-bit message pair ("0," "1") [see (17)], because the messages in both pairs differ only in the first bit, which can be reduced to the one-bit scenario discussed previously.

IV. HASH FUNCTION USING PARITY-DEPENDENT QUANTUM WALKS WITH ONE- AND TWO-STEP MEMORY ON CYCLES

The main part of the proposed hash function is constructed by numerically simulating CQWM-P with one- and two-step memory on cycles under the control of the input message, and then calculating the hash value from the resulting probability distribution of this walk. In order to make it applicable to short messages, we add pre and postprocessing steps, respectively, before and after the running of CQWM-P controlled by the message bits, in which we use QW2M because it lives in the same Hilbert space as QW2M-P does. The objective of the preprocessing step is to let the initial amplitudes spread to all nodes, so that the information of the short messages can quickly affect every node; the postprocessing can further diffuse the information of the message over the cycle and confuse the information of message bits at different positions. Since the preprocessing step is useful for improving collision resistance property when handling ordinary long messages, we make it a fixed step for all inputs.

Specifically, our hashing algorithm is parameterized by three positive integers $\{n, m, l | n \bmod 2 = 1, 10^l \gg 2^m\}$ and three angles $\{\theta_0, \theta_1, \alpha | \theta_0, \theta_1, \alpha \in (0, \pi/2)\}$, where n specifies the total number of nodes in the cycle that the walker moves along, m is the number of hash bits that are contributed by each node, l is the number of digits in the probability value (associated with each node) that are used to calculate the hash result, θ_0 (respectively, θ_1) is the coin parameter of QW1M-P (respectively, QW2M-P), and α is the parameter of the initial state of the walker. Given the t-bit input message msg, the $m \times n$-bit hash value H(msg) is calculated as follows.

1) *Initialization:* Initialize the walker in the state $|\psi_0\rangle = \cos\alpha\,|0,1,0,1\rangle + \sin\alpha\,|0,1,1,0\rangle$.
2) *Preprocessing:* Perform n steps of QW2M on the initial states $|\psi_0\rangle$ and get an intermediate state $|\psi_n\rangle$.
3) *Controlled quantum walk:* Apply U_{msg} to $|\psi_n\rangle$ and get $|\psi_{n+t}\rangle = U_{\text{msg}}\,|\psi_n\rangle$.
4) *Postprocessing a):* If $t < n$, first perform $n - t$ steps of QW2M on $|\psi_{n+t}\rangle$ and get $|\psi_{2n}\rangle$, then calculate an intermediate hash value H'(msg) from the amplitudes of $|\psi_{2n}\rangle$.
   Postprocessing b): if $t < n$, then apply $U_{H'(\text{msg})}$ to $|\psi_{2n}\rangle$ and get $|\psi_{n(m+2)}\rangle$.
5) *Obtaining the hash result:* If $t < n$, then calculate the probability distribution prob $= (p_0, p_1, \ldots, p_{n-1})$ from the amplitudes of $|\psi_{n(m+2)}\rangle$, where p_x is the probability that the walker locates at node x when the walk is finished; if $t >= n$, then calculate the resulting probability distribution from $|\psi_{n+t}\rangle$. The hash value of msg is a sequence of n blocks $H(\text{msg}) = B_0\|B_1\|\ldots\|B_{n-1}$, where each block B_x is the m-bit binary representation of $\lfloor p_x \cdot 10^l \rfloor \bmod 2^m$ ($\lfloor\cdot\rfloor$ denotes the floor of a number), and $B_x\|B_{x+1}$ denotes the concatenation of B_x and B_{x+1}.

IEEE Transactions on Quantum Engineering

V. STATISTICAL PERFORMANCE ANALYSIS

The proposed scheme is a kind of dedicated hash function, the security of which is hard to prove, and it is commonly evaluated by means of statistical analysis. To facilitate comparison and discussion, we consider two typical instances QHFM-P-264 and QHFM-P-296 of the proposed scheme, where QHFM-P-L produces L-bit hash values and will be compared with the existing DQW-based hash functions with L-bit output length. The values of m, l, θ_0, θ_1, and α for the two instances are the same, which are taken to be $8, 8, \pi/4, \pi/3$, and $\pi/3$, respectively; the only distinction between QHFM-P-296 and QHFM-P-264 lies in the value of n, which is taken to be 37 for the 296-bit output length and 33 for the 264-bit output length.

Our statistical tests consider four kinds of properties: sensitivity of hash value to message, diffusion and confusion properties, uniform distribution property, and collision resistance property, the latter three are assessed by analyzing the same collection of hash values, whose input messages come from the public arXiv dataset available online.[1]

A. SENSITIVITY OF HASH VALUE TO MESSAGE

Let msg_0 be an original message and msg_i ($i \in \{1, 2, 3\}$) be the slightly modified result of msg_0, the sensitivity of hash value to message is assessed by comparing the hash value $H(\text{msg}_i)$ of msg_i with the hash result $H(\text{msg}_0)$ of msg_0. Specifically, the original and modified messages are obtained under the following four conditions.

1) *Condition 0:* Randomly select a record from the dataset, take the texts of the abstract field within this record as msg_0
2) *Condition 1:* Invert a bit of msg_0 at a random position and then get the modified message msg_1.
3) *Condition 2:* Insert a random bit into msg_0 at a random position and then obtain msg_2.
4) *Condition 3:* Delete a bit from msg_0 at a random position and then obtain msg_3.

Corresponding to the above conditions, we list, as an example, four hash values in hexadecimal format produced by QWM-P-296 as follows.

1) $H(\text{msg}_0)$ ="DA EA FA 86 97 C2 15 ED 07 0D 19 4F 53 28 9E 0B 16 FA 6D F5 BF DA C2 7D 3B E1 A6 76 53 BA 3E A0 C3 E0 CE C0 26."
2) $H(\text{msg}_1)$ ="A8 11 FE 90 3D 59 BB 2B CA B7 2E 11 56 2B 0D 17 99 E8 1 C B0 B6 50 E9 F4 26 E9 0 C 78 32 50 FA 6E 12 12 4E B2 06."
3) $H(\text{msg}_2)$ ="BC 92 26 7 C 75 61 D5 49 6F 2E EA C3 36 B0 86 19 74 68 D3 27 57 B0 3F DA D0 63 87 8 A B2 28 B2 36 38 3F 93 9 A AC."
4) $H(\text{msg}_3)$ ="20 65 B2 B5 7E 8F B1 BF 79 95 BD 03 56 6D 33 7E A8 B5 3 A 06 1 A 6D 54 2D 78 36 F2 4 A 03 E5 66 51 E7 1E 75 0 C A3."

[1] [Online]. Available: https://www.kaggle.com/Cornell-University/arxiv

TABLE 3. Test Results of the Diffusion and Confusion Properties

Hash Instances or Schemes	$\overline{B}$	$\overline{P}(\%)$	ΔB	$\Delta P(\%)$	$I_{\text{DC}}(\%)$
QHFM-P-296	147.9528	49.9841	8.5678	2.8945	1.4552
QHFM-P-264	131.8894	49.9581	8.0638	3.0545	1.5482
QHFL-296 [3]	148.1900	50.0600	8.5500	2.8900	1.4750
QHFL-264 [3]	132.0300	50.0100	8.1100	3.0700	1.5400
QHFM-296 [2]	147.9101	49.9696	8.5997	2.9053	1.4679
QHFM-264 [2]	131.8667	49.9495	8.1378	3.0825	1.5665
Yang21-296 [6]	147.8640	49.9541	8.6141	2.9102	1.4781
Yang19-264 [7]	131.6803	49.8789	8.8877	3.3666	1.7439
Yang18-264 [8]	132.1108	50.0420	8.0405	3.0457	1.5439

The plots of $H(\text{msg}_0)$, $H(\text{msg}_1)$, $H(\text{msg}_2)$, and $H(\text{msg}_3)$ in the binary format are shown in Fig. 1 , where each asterisk (*) in the jth subgraph ($j > 0$) marks a different bit between $H(\text{msg}_j)$ and $H(\text{msg}_0)$. It indicates that a slight modification in the input message can lead to a significant change in the hash result, and the positions of changed bits are evenly distributed over the entire interval [1296] of position numbers. A similar result can be obtained using QHFM-P with other output lengths, thus, the output result of the proposed hash scheme is highly sensitive to its input message.

B. DIFFUSION AND CONFUSION PROPERTIES

To test the diffusion and confusion properties of the proposed hash function, the statistical experiment getting msg_0 and msg_1 is independently repeated N times, then the hash values of those N pairs of messages are analyzed. Let B_i ($i \in \{1, \ldots, N\}$) be the Hamming distance between $H(\text{msg}_0)$ and $H(\text{msg}_1)$ obtained in the ith experiment, the diffusion and confusion properties are assessed based on the following four indicators:

1) mean changed bit number $\overline{B} = \sum_{i=1}^{N} B_i/N$;
2) mean changed probability $\overline{P} = \overline{B}/(n \times m) \times 100\%$;
3) standard deviation of the changed bit number $\Delta B = \sqrt{\sum_{i=1}^{N}(B_i - \overline{B})^2/(N-1)}$;
4) standard deviation of the changed probability $\Delta P = \sqrt{\sum_{i=1}^{N}[B_i/(n \times m) - \overline{P}]^2/(N-1)} \times 100\%$.

The ideal value of $\overline{P}$ is 50%, and smaller ΔB and ΔP are more desirable. Following the work in [2], we take $N = 10\,000$ and use $I_{\text{DC}} = (\Delta P + |\overline{P} - 50\%|)/2 \times 100\%$ as a composite indicator for the diffusion and confusion properties. The test results of the diffusion and confusion properties for the proposed hash functions are presented in Table 3 . For comparison, the reported results for the existing DQW-based hash schemes with 296- or 264-bit output length are also listed in Table 3, where Yang21-296 is the second instance (with $p = 2/n$) in [6].

It can be seen that the test results for QHFM-P are very close to those for its peers; thus, the diffusion and confusion properties of the proposed scheme are on a par with those of existing schemes with output length 296 or 264.

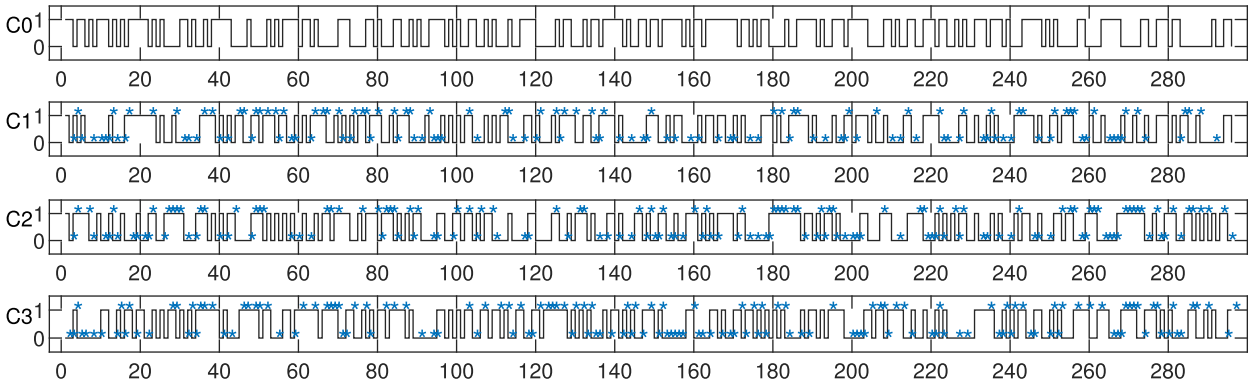

FIGURE 1. Plots of the hash values produced by QHFM-P-296 under the four conditions, where Cj stands for Condition j ($j = 0, 1, 2, 3$).

TABLE 4. Test Results of the Uniform Distribution Property

Hash Instances or Schemes	$\overline{T}$	$\lvert\overline{T} - 5000\rvert$	ΔT
QHFM-P-296	4998.41	1.59	51.1603
QHFM-P-264	4995.82	4.18	47.9474
QHFM-296 [2]	4996.96	3.04	48.4334
QHFM-264 [2]	4994.95	5.05	48.9253
Yang21-296 [6]	4998.10	1.90	N/A
Yang19-264 [7]	4996.60	3.40	N/A
Yang18-264 [8]	5003.90	3.90	N/A

C. UNIFORM DISTRIBUTION ANALYSIS

The uniform distribution property of the proposed hash function is assessed by analyzing the N pairs of hash values used in the diffusion and confusion test in a different way. Let T_j ($j \in \{1, 2, \ldots, n \times m\}$) be the number of experiments in which the jth bit of $H(\text{msg}_0)$ is different from the jth bit of $H(\text{msg}_1)$, then the uniform distribution analysis considers the following two indicators:

1) mean number of experiments with flipped hash bit over $n \times m$ bit-positions
$\overline{T} = \sum_{j=1}^{n\times m} T_j/(n \times m)$;
2) standard deviation of the number of experiments with flipped hash bit
$\Delta T = \sqrt{\sum_{j=1}^{n\times m}(T_j - \overline{T})^2/(n \times m - 1)}$.

The ideal value of $\overline{T}$ is $N/2$, and smaller ΔT suggests better uniform distribution property. As shown in Table 4 , the values of $\overline{T}$ and ΔT for QHFM-P-264 as well as the value of ΔT for QHFM-P-296 are close to the corresponding values for the state-of-the-art DQW-based hash functions[2] with the same output length. Since the reported value of $N \times \overline{P}$ (which is equivalent to $\overline{T}$ theoretically, see [2]) for Yang21-296 is 4995.41, the value of $|\overline{T} - 5000|$ for Yang21-296 may be considered between (or close to) 1.90 and 4.59, so the tested result of the mean number of experiments of QHFM-P-296 (with the value of $|\overline{T} - 5000|$ being 1.59) can be regarded as better than those of Yang21-296 and QHFM-296.

[2]The values of $\overline{T}$ and ΔT for QHFL are both not available, so we only compare QHFM with the remaining four state-of-the-art schemes.

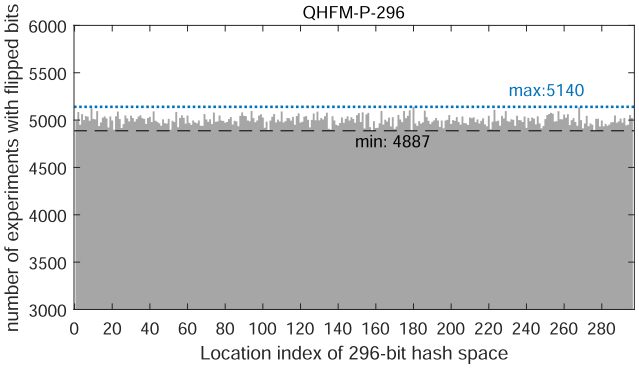

FIGURE 2. Histogram of the 296-bit hash space, where N =10 000.

To provide an intuitive description, we plot the number of experiments with flipped hash bit on every bit position of QHFM-P-296 in Fig. 2, where the number of experiments with flipped hash bit on every bit-position is very close to $N/2$, suggesting that the proposed scheme has good uniform distribution property.

D. COLLISION RESISTANCE

The collision resistance test is carried out by counting the number of experiments in which the hash values of the original and modified messages collide at a certain number of bytes, and then comparing the counting result with its theoretical value.

For ease of exposition, we use $\{\text{msg}_0^{(i)}, \text{msg}_1^{(i)}\}$ to denote the original and modified messages obtained under Condition 0 and Condition 1 in the ith experiment, $\{H(\text{msg}_0^{(i)}), H(\text{msg}_1^{(i)})\}$ to denote the hash values of $\{\text{msg}_0^{(i)}, \text{msg}_1^{(i)}\}$, and $g = \lceil (n \times m)/8 \rceil$ to denote the number of bytes that a hash result produced by the proposed hash function can be divided into.[3] The collision resistance test counts the numbers $\{W_N^e(\omega) | \omega = 0, 1, \ldots, g\}$ of experiments in which $H(\text{msg}_0)$ and $H(\text{msg}_1)$ have ω identical bytes (ω is also called the number of hits). For instance, if the first, third,

[3]If $n \times m$ is not divisible by 8, then add a prefix of $(8 - n \times m)$ mod 8 zeros to the hash value.

IEEE Transactions on Quantum Engineering

TABLE 5. Test Results of the Collision Resistance Property

Hash Instances or Schemes	$\{W_N^e(\omega)\|\omega=0,1,2,3,4+\}$	$D_{\mathrm{KL}}\left(P^e\|P^t\right)$	$\overline{d}^e_{\mathrm{byte}}$	$\lvert\overline{d}^e_{\mathrm{byte}}-\overline{d}^t_{\mathrm{byte}}\rvert$
QHFM-P-296	8640, 1270, 86, 4, 0	0.0000391	85.36	0.03
QHFM-P-264	8798, 1124, 74, 4, 0	0.0000589	85.29	0.04
QHFL-296 [3]	8637, 1278, 81, 4, 0	0.0000998	N/A	0.15
QHFL-264 [3]	8784, 1133, 82, 1, 0	0.0002427	N/A	0.02
QHFM-296 [2]	8605, 1312, 81, 2, 0	0.0003610	85.36	0.03
QHFM-264 [2]	8762, 1159, 74, 5, 0	0.0001457	85.27	0.06
Yang21-296 [6]	8321, 1547, 110, 22, 0	0.0086163	85.22	0.11
Yang19-264 [7]	9019, 923, 52, 2, 4	0.0056472	89.76	4.43
Yang18-264 [8]	8904, 1026, 68, 2, 0	0.0009686	83.64	1.69

and fourth bytes of $H(\mathrm{msg}_0)$ are, respectively, the same as the first, third, and fourth bytes of $H(\mathrm{msg}_1)$ in the 25th experiment, then $\{H(\mathrm{msg}_0^{(25)}), H(\mathrm{msg}_1^{(25)})\}$ makes an incremental contribution of 1 to $W_N^e(3)$.

The theoretical value of $W_N^t(\omega)$ is calculated using the binomial distribution formula

$$W_N^t(\omega) = \mathrm{int}\left[N \times P^t(\omega)\right] = \mathrm{int}\left[N \times \frac{g!}{\omega!(g-\omega)!}\left(\frac{1}{2^8}\right)^{\omega}\left(1-\frac{1}{2^8}\right)^{g-\omega}\right] \tag{19}$$

where int[·] denotes rounding a real number to its nearest integer, and $P^t(\omega)$ is the theoretical probability that ω hits occur in $\{H(\mathrm{msg}_0), H(\mathrm{msg}_1)\}$. Substituting $g = \lceil 296/8 \rceil = 37$ and N=10 000 into (19), one can get $\{W_N^t(\omega)|\omega = 0, 1, 2, 3\} = \{8652, 1255, 89, 4\}$ for hash functions with 296-bit output length. Similarly, the values of $W_N^t(\omega)$ with $\omega = 0, 1, 2, 3$ for 264-bit hash functions are 8788, 1137, 71, and 3, respectively. For both 296- and 264-bit hash functions, the value of $W_N^t(\omega)$ with $\omega \geq 4$ is 0.

Let $P^e(\omega) = W_N^e(\omega)/N$ be the experimental probability that $H(\mathrm{msg}_0)$ and $H(\mathrm{msg}_1)$ have ω identical bytes, the collision resistance property of the proposed hash function can be assessed by the Kullback–Leibler divergence (KL divergence) of P^e from P^t

$$D_{\mathrm{KL}}\left(P^e\|P^t\right) = \sum_{\omega=0}^{g} P^e(\omega)\log_2\left(\frac{P^e(\omega)}{P^t(\omega)}\right). \tag{20}$$

The smaller the value of $D_{\mathrm{KL}}(P^e\|P^t)$ is, the closer the P^e is to P^t. Note that there exist some values of ω such that $P^e(\omega) = 0$ and $P^t(\omega) \neq 0$, hence, we cannot use $D_{\mathrm{KL}}(P^t\|P^e)$ to indicate the distance between P^e and P^t.

In addition to the KL divergence, the mean of the absolute difference per byte between $H(\mathrm{msg}_0)$ and $H(\mathrm{msg}_1)$ over N independent experiment can also be used to assess the collision resistance property. Let $t(e_j)^{(i)}$ and $t(e'_j)^{(i)}$ be the decimal value of the jth byte of $H(\mathrm{msg}_0^{(i)})$ and $H(\mathrm{msg}_1^{(i)})$, respectively, and the mean of the absolute difference per byte is given by

$$\overline{d}^e_{\mathrm{byte}} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{g}\frac{1}{g}\left|t(e_j)^{(i)} - t(e'_j)^{(i)}\right| \tag{21}$$

and the theoretical value of $\overline{d}^e_{\mathrm{byte}}$ is $\overline{d}^t_{\mathrm{byte}} = 85.33$ [6].

The test results of the collision resistance property for the proposed hash function are given in Table 5, where $W_N^e(4+)$ denotes the number of experiments in which at least four hits occur in $\{H(\mathrm{msg}_0), H(\mathrm{msg}_1)\}$. The experimental values of $D_{\mathrm{KL}}(P^e\|P^t)$ for both instances of QHFM-P are noticeably smaller than those of its peers, and the values of $|\overline{d}^e_{\mathrm{byte}} - \overline{d}^t_{\mathrm{byte}}|$ for QHFM-P-296 and QHFM-P-264 are close to the best reported results of the corresponding indicator listed in Table 5. In addition, in our subsequent stability tests, more than 150 pairs of coin angles lead to very small KL divergences that are smaller than 0.0001, and about half instances of QHFM-P-296 have a KL divergence value smaller than 0.0002. Thus, the proposed hash function has excellent collision resistance property, and with certain coin angles, the collision resistance property of QHFM-P is better than that of the state-of-the-art hash functions based on discrete quantum walks.

E. STATISTICAL PROPERTIES FOR SHORT MESSAGES

To evaluate the effectiveness of the proposed scheme for short messages, we calculate the hash value of every binary string of length ranging from 0 to 17 and test the four kinds of statistical properties of QHFM-P-296 for each length.

As a representative result of the sensitivity of hash value to message for short messages, we plot the hash values of messages of length less than 3 in Fig. 3, where each * marks a different bit between H("0") and the remaining hash values. Let msg_0 = "0," then msg_1 = "1" is the result of inverting msg_0 by one bit (corresponding to Condition 1), msg_2 = "00,"msg_{2a} = "01," and msg_{2b} = "10" are the results of adding one bit to msg_0 (corresponding to Condition 2), and msg3 = "" is the result of removing one bit from msg_0 (corresponding to Condition 3). Fig. 3 indicates that a slight modification in the one-bit message can lead to a significant change in the hash value, and the position of changed bits are evenly distributed over the entire interval [1296] of position numbers. Similar results can be obtained for other short messages; thus, the hash value produced by the proposed scheme is highly sensitive to the input message of short length.

To examine the latter three properties of the proposed scheme for short messages, we perform QHFM-P-296 on all messages of length less than 18 and calculate the indicators

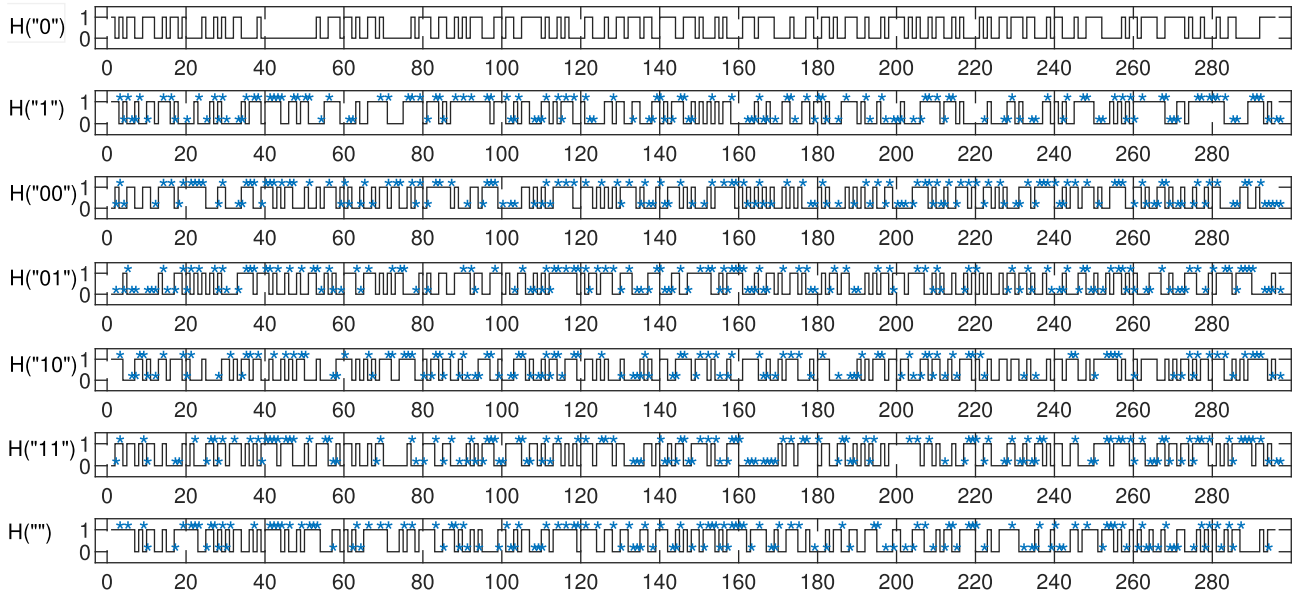

FIGURE 3. Plots of the hash values of binary strings of length less than 3 (produced by QHFM-P-296).

TABLE 6. Test Results of Statistical Properties for Short Messages

message length t	$\lvert\overline{P}-50\%\rvert$	$\Delta P(\%)$	$\lvert\overline{T}-N'/2\rvert$	ΔT	$D_{\mathrm{KL}}\left(P^e\Vert P^t\right)$	$\lvert\overline{d}^{\,e}_{\mathrm{byte}}-\overline{d}^{\,t}_{\mathrm{byte}}\rvert$
1	2.7027	0	0.02	0.4993	0.2089228	10.57
2	0.5068	1.4432	0.02	0.9793	0.0938697	1.80
3	1.1824	2.8080	0.14	1.5982	0.0271810	2.63
4	0.4329	2.4174	0.14	2.6312	0.0456890	1.08
5	0.6461	2.7706	0.52	4.6477	0.0026518	0.26
6	0.2921	3.0371	0.56	7.2347	0.0095323	1.34
7	0.1169	2.8418	0.52	10.4716	0.0072515	0.16
8	0.1478	2.8778	1.51	17.3573	0.0016027	0.37
9	0.0848	2.8078	1.95	24.2336	0.0010707	2.13
10	0.0058	2.9212	0.30	34.5544	0.0004007	0.37
11	0.0151	2.9309	1.70	58.3720	0.0001809	0.65
12	0.0352	2.9011	8.64	77.4177	0.0000839	0.35
13	0.0073	2.9223	3.91	111.3395	0.0000262	0.34
14	0.0114	2.8965	13.06	175.5284	0.0000384	0.21
15	0.0059	2.8975	14.41	260.7551	0.0000205	0.28
16	0.0030	2.9030	15.93	343.1278	0.0000052	0.02
17	0.0028	2.9051	31.73	526.4497	0.0000019	0.77

of the diffusion and confusion properties, the uniform distribution property, and the collision resistance property for each length. For example, the mean changed bit number $\overline{B}$ for message length t is obtained by dividing the summation of the Hamming distance between all t-bit message pairs that only differ in one bit by the total number N' of pairs. The t-bit original and modified messages $(\mathrm{msg}_0, \mathrm{msg}_1)$ can be traversed as follows: let msg_0 be taken from 0 to 2^t-1, and let the position j of the bit (within msg_0) that is to be inverted be taken from 1 to t; for each value of msg_0 and each value of j, if the jth bit of msg_0 is 0, then flip the jth bit of msg_0 and obtain the modified message msg_1. One can derive that the total number of t-bit message pairs is $N'=2^{t-1}\times t$.

In Table 6, we list the test results of the primary indicators for different message length. One can see that the value of $|\overline{P}-50\%|$ for t-bit messages tends to be smaller as t increases, which becomes closer to or smaller than that for long messages. The results of the uniform distribution property for short messages are comparable with that for long messages in the sense that, for different message length $t <= 17$, the values of ΔT are close to $\sqrt{N'}/2$, and the values of $|T-N'/2|$ are less than $\sqrt{N'}/20$, which agrees with the corresponding results given in Table 4. In addition, the value of $D_{\mathrm{KL}}(P^e\|P^t)$ tends to be smaller as t increases, and the value of $\overline{d}^{\,e}_{\mathrm{byte}}$ is close to the theoretical value $\overline{d}^{\,t}_{\mathrm{byte}}$ for most message length. Hence, the proposed hash function also possesses good diffusion and confusion properties, good uniform distribution property, and good collision resistance property for short messages.

VI. STABILITY WITH RESPECT TO COIN PARAMETERS

Recall from Section III that the proposed hash function is parameterized by three integers and three angles, among which the two coin angles are crucial components of the

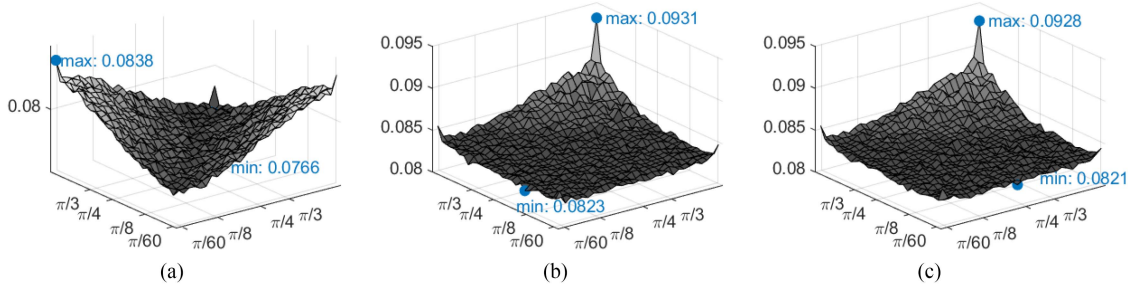

FIGURE 4. Mean JS divergences between Pd_0 and Pd_j ($j = 1, 2, 3$) for θ_0 and θ_1 in $[\pi/60, 29\pi/60]$. (a) $\bar{D}_{\mathrm{JS}}(\mathrm{Pd}_0,\mathrm{Pd}_1)$: (b) $\bar{D}_{\mathrm{JS}}(\mathrm{Pd}_0,\mathrm{Pd}_2)$: $\bar{D}_{\mathrm{JS}}(\mathrm{Pd}_0,\mathrm{Pd}_3)$.

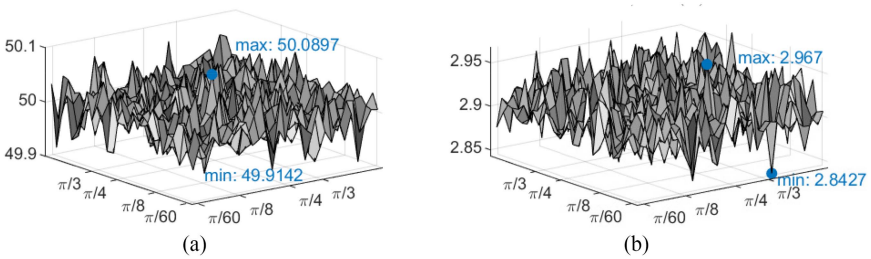

FIGURE 5. $\overline{P}$ and ΔP for θ_0 and θ_1 in $[\pi/60, 29\pi/60]$. (a) $\bar{P}$(%): (b) ΔP(%).

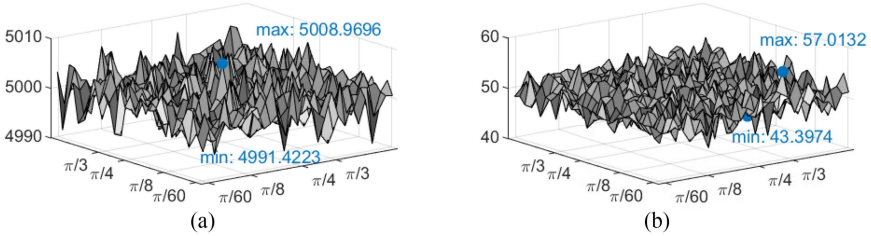

FIGURE 6. $\overline{T}$ and ΔT for θ_0 and θ_1 in $[\pi/60, 29\pi/60]$. (a) $\bar{T}$: (b) ΔT.

underlying controlled alternate quantum walks and may affect the four statistical properties considered in Section IV.

To explore how robust the hashing properties of QHFM-P are with respect to different coin angles, we uniformly divide $(0, \pi/2)$ into $c = 30$ subintervals, take the endpoints (except 0 and $\pi/2$) of those subintervals as the candidate values of each coin parameter, and then conduct $N = 10000$ experiments for each pair of values. During the experiments for each angle pair, primary indicators of each type of statistical property are calculated. Specifically, we take $\overline{P}$ together with ΔP, $\overline{T}$ together with ΔT, and $D_{\mathrm{KL}}(P^e \| P^t)$ together with $|\overline{d}^{\,e}_{\mathrm{byte}} - \overline{d}^{\,t}_{\mathrm{byte}}|$ to be the primary indicators of the diffusion and confusion properties, the uniform distribution property, and the collision resistance property, respectively. Following the work in [3], we take the mean Jensen–Shannon divergence (JS divergence) [26] (over N random experiments) between the resulting probability distributions corresponding to the original and modified messages to be the quantitative indicator of the sensitivity of hash value to message. Suppose, in an experiment, the probability distribution produced by the underlying controlled alternate quantum walks controlled by msg_j ($j = 0, 1, 2, 3$) is Pd_j, the JS divergence between Pd_0 and Pd_1 is

$$D_{\mathrm{JS}}(\mathrm{Pd}_0, \mathrm{Pd}_1) = \frac{D_{\mathrm{KL}}(\mathrm{Pd}_0 \| Md)}{2} + \frac{D_{\mathrm{KL}}(\mathrm{Pd}_1 \| Md)}{2} \quad (22)$$

where $Md = (\mathrm{Pd}_0 + \mathrm{Pd}_1)/2$, and $D_{\mathrm{KL}}(\mathrm{Pd}_0 \| Md)$ is the KL divergence of Pd_0 from Md [see (20)].

The results of the stability test for the four kinds of properties of QHFM-P-296 are illustrated in Figs. 4–7, where $\overline{D}_{\mathrm{JS}}(\mathrm{Pd}_0, \mathrm{Pd}_j)$ ($j = 1, 2, 3$) in Fig. 4 denotes the average value of $D_{\mathrm{JS}}(\mathrm{Pd}_0, \mathrm{Pd}_j)$ over N experiments. One may notice that the shape of Fig. 5(a) is identical to that of Fig. 6(a), this is because $\overline{T}$ is directly proportional to $\overline{P}$ when the N pairs of original and modified messages used in the diffusion and confusion tests are reused in the uniform distribution test.

It can be seen from Fig. 4(a) that the average JS divergences between Pd_0 and Pd_1 for different values of coin parameters fall within a narrow range, indicating that the value of $\overline{D}_{\mathrm{JS}}(\mathrm{Pd}_0, \mathrm{Pd}_1)$ is quite stable with respect to θ_0 and θ_1. Similarly, the maximum and minimum values of $\overline{D}_{\mathrm{JS}}(\mathrm{Pd}_0, \mathrm{Pd}_2)$ [or $\overline{D}_{\mathrm{JS}}(\mathrm{Pd}_0, \mathrm{Pd}_3)$] are very close to each

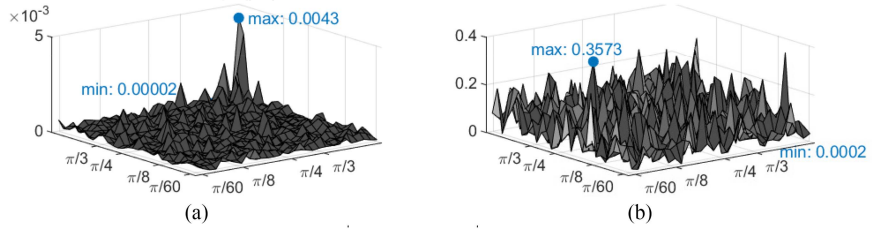

FIGURE 7. $D_{\mathrm{KL}}(P^e \| P^t)$ and $|\overline{d}^{e}_{\mathrm{byte}} - \overline{d}^{t}_{\mathrm{byte}}|$ for θ_0 and θ_1 in $[\pi/60, 29\pi/60]$. (a) $D_{\mathrm{KL}}(P^e \| P^t)$: (b) $|\bar{d}^{e}_{\mathrm{byte}} - \bar{d}^{t}_{\mathrm{byte}}|$.

other, suggesting that the sensitivity of hash value to message of the proposed hash function is stable with respect to the coin parameters.

In Fig. 5, the mean changed probability $\overline{P}$ fluctuates around 50%; meanwhile, the value of the standard deviation ΔP is small and does not vary significantly for different values of θ_0 and θ_1. Hence, the diffusion and confusion properties of QHFM-P are robust with regard to different coin angles. Likewise, the experimental values of $\overline{T}$ for different coin angles are close to the theoretical value $N/2$, and the results of ΔT displayed in Fig. 6 are small relative to the number of experiments. So, the uniform distribution property of the proposed hash function is also stable with respect to coin angles.

In Fig. 7, the values of $|\overline{d}^{e}_{\mathrm{byte}} - \overline{d}^{t}_{\mathrm{byte}}|$ for different hash instances of QHFM-P (with different coin angles) are small, and most coin angle pairs lead to very small values of $D_{\mathrm{KL}}(P^e \| P^t)$: 728 out of 841 hash instances of QHFM-P have a KL divergence that is smaller than 0.0005, and only 19 instances have a KL divergence that is greater than 0.001. Thus, the collision resistance property of the proposed hash function is stable with respect to coin parameters. With certain coin angles (135 out of 841 instances), the KL divergence becomes smaller than 0.00009, which suggests a better collision resistance property than that of the state-of-the-art hash functions based on discrete quantum walks (the best reported result of the KL divergence comes from QHFL-296; see Table 5).

VII. TIME AND SPACE COMPLEXITY

Similar to existing DQW-based hash schemes, the proposed hash algorithm can be efficiently computed using a classical computer. Thus, it is more appropriate to consider QHFM-P as a classical algorithm and to concentrate on classical complexity.

By rewriting the four-term basis state $|x, d_2, d_1, c\rangle$ of QWHF-P in $\mathcal{H}_p \otimes \mathcal{H}_{d_2} \otimes \mathcal{H}_{d_1} \otimes \mathcal{H}_c$ as a two-term basis state $|x, j\rangle = |x, 2^2 d_2 + 2^1 d_1 + 2^0 c\rangle$ in $\mathcal{H}_p \otimes \mathcal{H}^8$, where $\mathcal{H}^8$ is the 8-D Hilbert space, the quantum state of the walker performing parity-dependent quantum walks with one- and two-step memory on a cycle of length n after t time steps can be expressed as

$$|\psi_t\rangle = \sum_{x=0}^{n-1} \sum_{j=0}^{7} A_t^{x,j} |x, j\rangle . \tag{23}$$

Combining (23) with the first and last columns of Table 1, one can obtain the relation between $\{A_{t+1}^{x,j} | x \in \mathbb{Z}_n, j \in \mathbb{Z}_8\}$ and $\{A_t^{x,j} | x \in \mathbb{Z}_n, j \in \mathbb{Z}_8\}$ (hereafter, simply $\{A_{t+1}^{x,j}\}$ and $\{A_t^{x,j}\}$, respectively) after one step of QW2M-P as follows:

$$\begin{aligned}
A_{t+1}^{x,0} &= aA_t^{x+1,0} + bA_t^{x+1,1} \\
A_{t+1}^{x,1} &= cA_t^{x+1,4} + dA_t^{x+1,5} \\
A_{t+1}^{x,2} &= aA_t^{x-1,4} + bA_t^{x-1,5} \\
A_{t+1}^{x,3} &= cA_t^{x-1,0} + dA_t^{x-1,1} \\
A_{t+1}^{x,4} &= aA_t^{x+1,6} + bA_t^{x+1,7} \\
A_{t+1}^{x,5} &= cA_t^{x+1,2} + dA_t^{x+1,3} \\
A_{t+1}^{x,6} &= aA_t^{x-1,2} + bA_t^{x-1,3} \\
A_{t+1}^{x,7} &= cA_t^{x-1,6} + dA_t^{x-1,7}.
\end{aligned} \tag{24}$$

Relation (24) shows that the eight amplitudes of being at position x at time $t+1$ can be calculated from the amplitudes of being at positions $x \pm 1$ at time t using 16 multiplications and 8 additions, meaning that the $8n$ amplitudes of being at n locations can be calculated using $O(n)$ basic arithmetic operations. Similarly, one can obtain the relation between $\{A_{t+1}^{x,j}\}$ and $\{A_t^{x,j}\}$ for QW1M-P, where the amplitudes can also be updated using $O(n)$ basic operations at each time step.

If one wants to obtain an L-bit hash value (L is a multiple of m) of an M-bit message, then the walker moves M steps on a cycle with $L/m = O(L)$ nodes, here m is constant with respect to M. The assignment of the initial amplitudes $\{A_0^{x,j}\}$ can be carried out with $O(L)$ time and $O(L)$ memory space, the values of $\{A_M^{x,j}\}$ can be calculated from $\{A_0^{x,j}\}$ using $O(ML)$ basic operations with $O(L)$ space, and the hash value can be computed from $\{A_M^{x,j}\}$ using $O(L)$ multiplications and $O(L)$ modulo operations with $O(L)$ space. As a result, the time and space complexity of QHFM-L are $O(ML)$ and $O(L)$, respectively, which are the same as those of the state-of-the-art hash functions [2], [3], [6], [7], [8] based on discrete quantum walks.

VIII. CONCLUSION

In this article, a new QWM-based hash function QHFM-P is proposed and analyzed. Similar to the existing QWM-based hash function QHFM [2], the proposed scheme is also

constructed by using quantum walks with one- and two-step memory. The major distinction between QHFM-P and QHFM lies in that the underlying walks with two-step memory of QHFM and QHFM-P are different extensions of Mc Gettrick's QW1M model [14] and that pre and postprocessing steps are applied for short messages before and after the performance of the controlled quantum walks. With these modifications, the proposed scheme acheives better sensitivity of hash value to message and better collision resistance property than QHFM. QHFM-P has the same time and space complexity as those of QHFM, and its four kinds of statistical properties are quite stable with respect to the coin parameters.

The proposed hashing scheme has various applications in classical (including postquantum) and quantum cryptography, such as modification detection, authentic channel, confirmation of knowledge, key derivation, pseudorandom number generation, hash based public-key signature, and privacy amplification in quantum key distribution. For instance, it can act as the underlying hash function of Merkle's tree authentication scheme and be used to generating the MSS key pairs (as an PRNG).

It is noteworthy that there are some types of initial states that can significantly compromise the security of QHFM-P. Considering the scope and focus of this article, we only provides an initial insight into the consideration of the impact of the initial states on the security of QWM-based hash functions. To avoid using inappropriate initial states in QWM-based hash functions, a more complete analysis for the evolution of CQWM-P is required, and more general constraints are needed to be established for the initial states, which will be an important topic in our future work.

REFERENCES

[1] D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, pp. 188–194, Sep. 2017, doi: 10.1038/nature23461.

[2] Q. Zhou and S. F. Lu, "Hash function based on controlled alternate quantum walks with memory," *IEEE Trans. Quantum Eng.*, vol. 3, 2021, Art. no. 3100310, doi: 10.1109/TQE.2021.3130256.

[3] P. Hou, T. Shang, Y. Zhang, Y. Tang, and J. Liu, "Quantum hash function based on controlled alternate lively quantum walks," *Sci. Rep.*, vol. 13, no. 1, Apr. 2023, Art. no. 5887, doi: 10.1038/s41598-023-33119-w.

[4] W. M. Shi, S. Wang, T. Pan, Y. G. Yang, and Y. H. Zhou, "Continuous-time quantum hash function based on one-dimensional cycle lattice," *Mod. Phys. Lett. B.*, vol. 36, no. 19, 2022, Art. no. 2150241, doi: 10.1142/S0217984921502419.

[5] J. J. Shi et al., "A quantum hash function with grouped coarse-grained boson sampling," *Quantum Inf. Process.*, vol. 21, no. 2, Jan. 2022, Art. no. 73, doi: 10.1007/s11128-022-03416-w.

[6] Y. G. Yang, J. R. Dong, Y. L. Yang, Y. H. Zhou, and W. M. Shi, "Usefulness of decoherence in quantum-walk-based hash function," *Int. J. Theor. Phys.*, vol. 60, pp. 1025–1037, Jan. 2021, doi: 10.1007/s10773-021-04724-0.

[7] Y. G. Yang, J. L. Bi, D. Li, Y. H. Zhou, and W. M. Shi, "Hash function based on quantum walks," *Int. J. Theor. Phys.*, vol. 58, no. 6, pp. 1861–1873, Mar. 2019, doi: 10.1007/s10773-019-04081-z.

[8] Y. G. Yang, J. L. Bi, X. B. Chen, Z. Yuan, Y. H. Zhou, and W. M. Shi, "Simple hash function using discrete-time quantum walks," *Quantum Inf. Process.*, vol. 17, no. 8, Jun. 2018, Art. no. 189, doi: 10.1007/s11128-018-1954-2.

[9] Y. G. Yang, Y. C. Zhang, G. Xu, X. B. Chen, Y. H. Zhou, and W. M. Shi, "Improving the efficiency of quantum hash function by dense coding of coin operators in discrete-time quantum walk," *Sci. China-Phys. Mech. Astron.*, vol. 61, no. 3, Jan. 2018, Art. no. 030312, doi: 10.1007/s11433-017-9132-y.

[10] D. Li, Y. G. Yang, J. L. Bi, J. B. Yuan, and J. Xu, "Controlled alternate quantum walks based quantum hash function," *Sci. Rep.*, vol. 8, no. 1, Jan. 2018, Art. no. 225, doi: 10.1038/s41598-017-18566-6.

[11] W. F. Cao, Y. C. Zhang, Y. G. Yang, D. Li, Y. H. Zhou, and W. M. Shi, "Constructing quantum hash functions based on quantum walks on Johnson graphs," *Quantum Inf. Process.*, vol. 17, no. 7, May 2018, Art. no. 156, doi: 10.1007/s11128-018-1923-9.

[12] Y. G. Yang, P. Xu, R. Yang, Y. H. Zhou, and W. M. Shi, "Quantum hash function and its application to privacy amplification in quantum key distribution, pseudo-random number generation and image encryption," *Sci. Rep.*, vol. 6, Jan. 2016, Art. no. 19788, doi: 10.1038/srep19788.

[13] D. Li, J. Zhang, F. Z. Guo, W. Huang, Q. Y. Wen, and H. Chen, "Discrete-time interacting quantum walks and quantum hash schemes," *Quantum Inf. Process.*, vol. 12, no. 3, pp. 1501–1513, Mar. 2013, doi: 10.1007/s11128-012-0421-8.

[14] M. Mc Gettrick, "One dimensional quantum walks with memory," *Quantum Inf. Comput.*, vol. 10, no. 5, pp. 509–524, May 2010, doi: 10.5555/2011362.2011371.

[15] Q. Zhou and S. F. Lu, "One-dimensional quantum walks with two-step memory," *Quantum Inf. Process.*, vol. 18, no. 12, Oct. 2019, Art. no. 359, doi: 10.1007/s11128-019-2475-3.

[16] D. Li, M. Mc Gettrick, Y. G. Yang, J. Xu, and Y. Wang, "Quantum walks with memory provided by parity of memory," *Int. J. Theor. Phys.*, vol. 59, no. 6, pp. 1934–1943, Jun. 2020, doi: 10.1007/s10773-020-04466-5.

[17] D. Li, Y. Liu, Y. G. Yang, J. Xu, and J. B. Yuan, "Szegedy quantum walks with memory on regular graphs," *Quantum Inf. Process.*, vol. 19, pp. 1–12, 2020, doi: 10.1007/s11128-019-2534-9.

[18] W. Dai, J. Yuan, and D. Li, "Discrete-time quantum walk with memory on the Cayley graph of the dihedral group," *Int. J. Theor. Phys.*, vol. 59, no. 1, pp. 10–28, Jan. 2020, doi: 10.1007/s10773-019-04257-7.

[19] W. J. Dai, J. B. Yuan, and D. Li, "Discrete-time quantum walk on the Cayley graph of the dihedral group," *Quantum Inf. Process.*, vol. 17, pp. 1–21, 2018, doi: 10.1007/s11128-018-2101-9.

[20] G. D. Molfetta, D. O. Soares-Pinto, and S. M. Duarte Queiros, "Elephant quantum walk," *Phys. Rev. A.*, vol. 97, no. 6, Jun. 2018, Art. no. 062112, doi: 10.1103/PhysRevA.97.062112.

[21] D. Li, M. Mc Gettrick, F. Gao, J. Xu, and Q. Y. Wen, "Generic quantum walks with memory on regular graphs," *Phys. Rev. A.*, vol. 93, no. 4, Apr. 2016, Art. no. 042323, doi: 10.1103/PhysRevA.93.042323.

[22] M. Mc Gettrick and J. A. Miszczak, "Quantum walks with memory on cycles," *Phys. A.*, vol. 399, pp. 163–170, Apr. 2014, doi: 10.1016/j.physa.2014.01.002.

[23] N. Konno and T. Machida, "Limit theorems for quantum walks with memory," *Quantum Inf. Comput.*, vol. 10, no. 11, pp. 1004–1017, Nov. 2010, doi: 10.48550/arXiv.1004.0443.

[24] N. Konno, "A new type of limit theorems for the one-dimensional quantum random walk," *J. Math. Soc. Jpn.*, vol. 57, no. 4, pp. 1179–1195, 2005, doi: 10.2969/jmsj/1150287309.

[25] A. Ambainis, E. Bach, A. Nayak, A. Vishwanath, and J. Watrous, "One-dimensional quantum walks," in *Proc. 33th Annu. ACM Symp. Theory Comput.*, 2001, pp. 37–49, doi: 10.1145/380752.380757.

[26] B. Fuglede and F. Topsøe, "Jensen-Shannon divergence and Hilbert space embedding," in *Proc. Int. Symp. Inf. Theory*, 2004, p. 30, doi: 10.1109/ISIT.2004.1365067.