

SCIENTIFIC REPORTS



OPEN

The multi-level and multi-dimensional quantum wavelet packet transforms

Received: 7 June 2018

Accepted: 6 September 2018

Published online: 17 September 2018

Hai-Sheng Li¹, Ping Fan², Hai-ying Xia¹, Shuxiang Song¹ & Xiangjian He^{1,3}

The classical wavelet packet transform has been widely applied in the information processing field. It implies that the quantum wavelet packet transform (QWPT) can play an important role in quantum information processing. In this paper, we design quantum circuits of a generalized tensor product (GTP) and a perfect shuffle permutation (PSP). Next, we propose multi-level and multi-dimensional (1D, 2D and 3D) QWPTs, including a Haar QWPT (HQWPT), a D4 QWPT (DQWPT) based on the periodization extension and their inverse transforms for the first time, and prove the correctness based on the GTP and PSP. Furthermore, we analyze the quantum costs and the time complexities of our proposed QWPTs and obtain precise results. The time complexities of HQWPTs is at most 6 on 2^n elements, which illustrates high-efficiency of the proposed QWPTs. Simulation experiments demonstrate that the proposed QWPTs are correct and effective.

With the rapid development in the fields of optical imaging, Internet technology, high performance calculation etc., the amount of data is increasing explosively, so that it is necessary to find new ways to store and process information. Quantum information processing (QIP)¹ as new technology of information processing, offers a potential solution to store and process massive visual data efficiently. QIP has two outstanding merits: (1) the unique computing performance of quantum coherence, entanglement and superposition [1], and (2) quantum storage capacity increasing exponentially. Models of quantum image representation^{2–8} have displayed the enormous storage capacity of QIP. Other popular quantum algorithms, such as the Shor's discrete logarithms and integer-factoring algorithms⁹, the Deutsch's parallel computing algorithm¹⁰ and the Grover's quadratic speed up algorithm¹¹, have further shown that QIP is more efficient than its classical counterparts. In addition, many algorithms of QIP emerge continually, and these algorithms include quantum geometric transformation^{12–14}, quantum image encryption and decryption algorithms^{15,16}, quantum watermarking¹⁷, quantum image compression⁶, quantum edge detection¹⁸, and quantum image filtering¹⁹.

The classical wavelet packet transform (WPT) has been widely spread to the information processing field for image coding²⁰, pattern matching²¹ and fractional brownian motion decorrelation²². It indicates that the quantum wavelet packet transform (QWPT) plays an important role in QIP. Unfortunately, the research on QWPT is rare and still preliminary. For example, two important QWPTs, namely the Haar QWPT (HQWPT) and the D4 QWPT (DQWPT) proposed in^{23–26}, are still single level quantum wavelet transforms. Up to now, we have not yet found any implementation of a multi-level and multi-dimensional QWPT. Therefore, we believe that QWPTs deserve further research.

In this paper, we introduce the generalized tensor product (GTP) and the perfect shuffle permutation (PSP), and design quantum circuits for them. Then, we propose the iterations and implementation circuits of the multi-level and multi-dimensional QWPT and inverse QWPT (IQWPT). QWPTs and the inverse QWPTs being considered include HQWPT, DQWPT based on a periodization extension, the inverse HQWPT (IHQWPT), the inverse DQWPT (IDQWPT). In addition, we analyze the quantum costs and time complexities of the proposed circuits and prove that the multi-level and multi-dimensional HQWPT can be implemented with a complexity of O(1). Simulation experiments demonstrate that the proposed QWPTs are correct and effective.

The contributions of this paper are listed as follows.

¹Guangxi Normal University, College of Electronic Engineering, Guilin, 541004, China. ²East China JiaoTong University, College of Information Engineering, Nanchang, 330013, China. ³University of Technology, Sydney, School of Electrical and Data Engineering, Sydney, NSW, 2000, Australia. Correspondence and requests for materials should be addressed to H.-S.L. (email: lhs1974@mailbox.gxnu.edu.cn)

Notation	Gate	Notation
I_2		V
$(I_2)^{\otimes n}$		V^+
X		
H		
U		

Figure 1. Notations for some base gates with their corresponding symbols.

- We analyze precisely the complexities of the simulated networks of controlled NOT gates with multi-control qubits. Comparing with the methods proposed in the reference²⁷, our proposed simulated networks are reduced by 50% approximately.
- We design the simplified circuits of the PSP and reduce time complexity to 6 for 2^n elements.
- We present the multi-level and multi-dimensional QWPTs, including HQWPT, IHQWPT, DQWPT and IDQPT for the first time, and prove the correctness by theoretical derivations and simulation experiments.
- We design the circuits of the multi-level and multi-dimensional HQWPT with the complexity $O(1)$, which has the overwhelming advantage over the classic Haar WPT.

The Quantum Implementation of GTP

Let A be an $n \times n$ matrix and B be an $m \times m$ matrix, then the tensor product $A \otimes B$ is an $mn \times mn$ block matrix in the following equation,

$$A \otimes B = \begin{bmatrix} A_{0,0}B & \cdots & A_{0,n-1}B \\ \vdots & \ddots & \vdots \\ A_{n-1,0}B & \cdots & A_{n-1,n-1}B \end{bmatrix}. \quad (1)$$

Thus, the tensor product of quantum states are defined as the tensor product of matrices: $|u\rangle \otimes |v\rangle = [u_0 \dots u_{2^n-1}]^T \otimes [v_0 \dots v_{2^n-1}]^T$, which is also written simply as $|uv\rangle$ or $|uv\rangle$.

Then, n fold tensor product $U \otimes U \otimes \dots \otimes U$ is abbreviated as $U^{\otimes n}$. Similarly, the abbreviation of $|u\rangle \otimes |u\rangle \otimes \dots \otimes |u\rangle$ is $|u\rangle^{\otimes n}$.

A larger vector space can be formed by putting vector spaces together. For instance, suppose that $|i\rangle$ is a basic state in a 2^n dimensional Hilbert space for $i=0, 1, \dots, 2^n-1$, the state $|i\rangle$ consists of the tensor products of the n computation basis states:

$$|i\rangle = |i_n\rangle \otimes |i_{n-1}\rangle \otimes \dots \otimes |i_1\rangle = |i_n\rangle |i_{n-1}\rangle \dots |i_1\rangle = |i_n i_{n-1} \dots i_1\rangle, \quad (2)$$

where $i = \sum_{j=1}^n i_j \times 2^{j-1}$ and $i_1, i_2, \dots, i_n \in \{0, 1\}$. Its dual state is

$$\langle i| = \langle i_n| \otimes \langle i_{n-1}| \otimes \dots \otimes \langle i_1| = \langle i_n| \langle i_{n-1}| \dots \langle i_1| = \langle i_n i_{n-1} \dots i_1|. \quad (3)$$

There are some base gates and their corresponding symbols shown in Fig. 1. In the figure, the identity (I_2), Hadamard (H), Pauli-X (X) and Swap gates are well-known and can be found in the reference²⁸. The $2^n \times 2^n$ identity matrix $(I_2)^{\otimes n} = I_{2^n}$ denotes the circuit of n qubits. V and V^+ are two specific examples of U gates where U corresponds to a unitary matrix and

$$V = \frac{1+i}{2} \begin{bmatrix} 1 & -i \\ -i & 1 \end{bmatrix}, \quad V^+ = \frac{1-i}{2} \begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix}. \quad (4)$$

A controlled gate is one of the most useful gates in quantum computing, and we define two controlled gates of $(n+m)$ -qubits.

Definition 1. Let U_{2^m} be a $2^m \times 2^m$ unitary matrix, I_{2^m} be a $2^m \times 2^m$ identity matrix. Then, controlled gates $C_n^j(U_{2^m})$ and $V_n^j(U_{2^m})$ with n control qubits and m target qubits are defined by

$$C_n^j(U_{2^m}) = (|j\rangle\langle j|) \otimes U_{2^m} + \sum_{i=0, i \neq j}^{2^n-1} ((|i\rangle\langle i|) \otimes I_{2^m}), \quad (5)$$

$$V_n^j(U_{2^m}) = (U_{2^m} \otimes |j\rangle\langle j|) + \sum_{i=0, i \neq j}^{2^n-1} (I_{2^m} \otimes (|i\rangle\langle i|)), \quad (6)$$

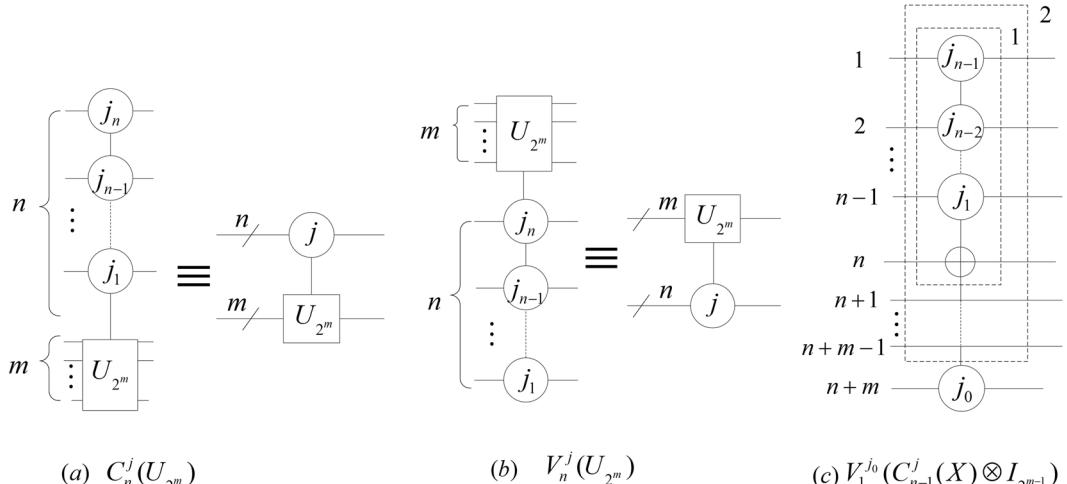


Figure 2. The $(n+m)$ qubit controlled gates and the N_{n+m}^n gate. The abbreviation notations are in the right parts of (a,b). The dashed box 1 and 2 in (c) implement $C_{n-1}^j(X)$ and $C_{n-1}^j(X) \otimes I_{2^{m-1}}$ where $j = j_{n-1}j_{n-2} \dots j_1$ and $j_{n-1}, j_{n-2}, \dots, j_1, j_0 \in \{0, 1\}$.

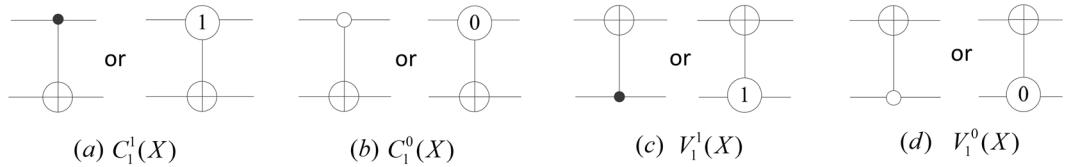


Figure 3. The four N_2^1 gates. The numbers 1 and 0 can be replaced by black and white points on control qubits.

where $|i\rangle = |i_1 \cdots i_n i_1\rangle$ and $|j\rangle = |j_1 \cdots j_2 j_1\rangle$ are the basic states in a 2^n dimensional Hilbert space shown in Eq. (2), and $j \in \{0, 1, \dots, 2^n - 1\}$. The Notations of $C_n^j(U_{2^m})$ and $V_n^j(U_{2^m})$ are shown in (a) and (b) of Fig. 2. Furthermore, $C_2^j(X)$ and $V_2^j(X)$ are called Toffoli gates.

Definition 2. An $(n+m)$ qubit controlled gate with n control qubits is named as an N_{n+m}^n gate, when the X gate is in the target qubit of the controlled gate. An instance of an N_{n+m}^n gate is shown in (c) of Fig. 2. In addition, the four N_2^1 gates shown in Fig. 3 are called controlled-NOT gates.

A Swap gate can be simulated by three N_2^1 gates, that is, $\text{Swap} = C_1^0(X)V_1^0(X)C_1^0(X)$.

Next, we introduce a perfect shuffle permutation. Let $P_{n,m}$ be the $mn \times mn$ matrix of a perfect shuffle permutation, then $P_{n,m}$ satisfies that $(P_{n,m})_{k,l} = \delta_{v,z} \delta_{z,v'}$ where $k = vn + z$, $l = v'm + z'$, $0 \leq v, z' < m$, $0 \leq v', z < n$, $\delta_{x,y}$ is the Kronecker delta function, that is, $\delta_{x,y} = 0$ if $x \neq y$, otherwise $\delta_{x,y} = 1$. Therefore, $P_{n,m}$ shuffles n packs of m cards into m packs of n cards.

As a useful tool for wavelet transforms, the GTP is defined as follows²⁹. Suppose that $\mathcal{A} = \{A^0, A^1, \dots, A^{m-1}\}$ and $\mathcal{B} = \{B^0, B^1, \dots, B^{n-1}\}$ are two sets of matrices, where A^i is an $n \times n$ matrix, $0 \leq i < m$, and B^j is an $m \times m$ matrix, $0 \leq j < n$. Then, the generalized tensor product $C = \mathcal{A} \otimes \mathcal{B}$ is an $mn \times mn$ matrix and can be calculated by

$$C = \mathcal{A} \otimes \mathcal{B} = P_{m,n} Diag(\mathcal{A}) P_{n,m} Diag(\mathcal{B}), \quad (7)$$

where $\text{Diag}(\mathcal{A}) = \text{Diag}(A^0, A^1, \dots, A^{m-1})$ and $\text{Diag}(\mathcal{B}) = \text{Diag}(B^0, B^1, \dots, B^{n-1})$ are block diagonal matrices.

Definition 3. Let $\mathcal{A} = \{A^0, A^1, \dots, A^{m-1}\}$ and $\mathcal{D} = \{D^0, D^1, \dots, D^{m-1}\}$ be two sets of matrices where A^i and D^i are $n \times n$ matrices. Then, the generalized product is defined as $\mathcal{A} \times \mathcal{D} = \mathcal{AD} = \{A^0 \times D^0, A^1 \times D^1, \dots, A^{m-1} \times D^{m-1}\}$.

Definition 4. The transpose, conjugate transpose and inverse of the matrix set \mathcal{A} are defined as follows:

$$\begin{cases} \mathcal{A}^T = \{(A^0)^T, (A^1)^T, \dots, (A^{m-1})^T\}, \\ \mathcal{A}^+ = \{(A^0)^+, (A^1)^+, \dots, (A^{m-1})^+\}, \\ \mathcal{A}^{-1} = \{(A^0)^{-1}, (A^1)^{-1}, \dots, (A^{m-1})^{-1}\}, \end{cases}$$

where $(A^i)^T$, $(A^i)^+$ and $(A^i)^{-1}$ denote the transpose, conjugate transpose and inverse of matrix A^i , respectively.

The following equations hold by using equation (7) and definitions 3 and 4.

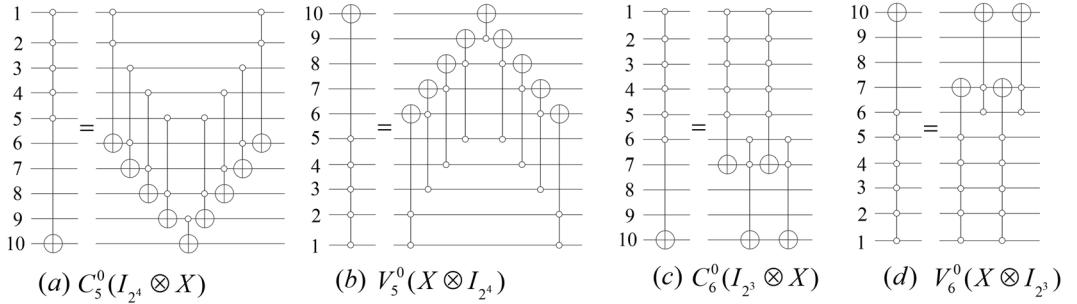


Figure 4. The controlled gate N_n^m illustrated for $n = 10$ and $m \in \{5, 6\}$.

$$\begin{cases} (\mathcal{A} \otimes \mathcal{B})^T = P_{m,n}(\mathcal{B}^T \otimes \mathcal{A}^T)P_{n,m}, \\ (\mathcal{A} \otimes \mathcal{B})^+ = P_{m,n}(\mathcal{B}^+ \otimes \mathcal{A}^+)P_{n,m}, \\ (\mathcal{A} \otimes \mathcal{B})^{-1} = P_{m,n}(\mathcal{B}^{-1} \otimes \mathcal{A}^{-1})P_{n,m}. \end{cases} \quad (8)$$

Let \mathcal{A} and \mathcal{C} be two sets of matrices containing m matrices with size $n \times n$, \mathcal{B} and \mathcal{D} be two sets of matrices containing n matrices with size $m \times m$, and I_m and I_n be $m \times m$ and $n \times n$ identity matrices, respectively. Then, the following equation holds²⁴:

$$(\mathcal{A} \times \mathcal{C}) \otimes (\mathcal{B} \times \mathcal{D}) = (\mathcal{A} \otimes I_m) \times (\mathcal{C} \otimes \mathcal{B}) \times (I_n \otimes \mathcal{D}), \quad (9)$$

and implies

$$\begin{cases} \mathcal{A} \otimes \mathcal{D} = (\mathcal{A} \times I_n) \otimes (I_m \times \mathcal{D}) = (\mathcal{A} \otimes I_m) \times (I_n \otimes \mathcal{D}), \\ (\mathcal{A} \times \mathcal{C}) \otimes I_m = (\mathcal{A} \otimes I_m) \times (\mathcal{C} \otimes I_m). \end{cases} \quad (10)$$

Furthermore, calculating by the definition of a GTP, we can implement the following four GTPs using controlled gates:

$$\begin{cases} I_2 \otimes \{I_{2^n}, U_{2^n}\} = (|0\rangle\langle 0|) \otimes I_{2^n} + (|1\rangle\langle 1|) \otimes U_{2^n} = C_1^1(U_{2^n}), \\ I_2 \otimes \{U_{2^n}, I_{2^n}\} = (|0\rangle\langle 0|) \otimes U_{2^n} + (|1\rangle\langle 1|) \otimes I_{2^n} = C_1^0(U_{2^n}) \\ \{I_{2^n}, U_{2^n}\} \otimes I_2 = I_{2^n} \otimes (|0\rangle\langle 0|) + U_{2^n} \otimes (|1\rangle\langle 1|) = V_1^1(U_{2^n}), \\ \{U_{2^n}, I_{2^n}\} \otimes I_2 = U_{2^n} \otimes (|0\rangle\langle 0|) + I_{2^n} \otimes (|1\rangle\langle 1|) = V_1^0(U_{2^n}). \end{cases} \quad (11)$$

The Complexity Analysis of Quantum Circuits

The complexity analysis of quantssum circuits. Since a quantum circuit can be simulated by basic operations referring to single-qubit gates, controlled-NOT gates, controlled-V and controlled- V^+ gates^{12,27,28,30}, we introduce some definitions and lemmas. Furthermore, $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ are the symbols of round down and round up respectively, which are used in the following definitions and lemmas.

Definition 5. The quantum cost of a quantum circuit can be regarded as the total number of basic operations which simulate the circuit, marked by $C()$.

Definition 6. The time complexity of a quantum circuit is defined by the total number of time steps. In a time step, only one basic operation is executed serially, but multiple ones can be performed in parallel. It is marked by $C^p()$.

Lemma 1. When $n \geq 6$ and $m \in \{3, 4, \dots, \lfloor n/2 \rfloor\}$, an N_n^m gate can be simulated by a network consisting of $2(m-1)$ Toffoli gates and a basic operation.

For instance, $C_m^0(I_{2^{n-m-1}} \otimes X)$ and $V_m^0(X \otimes I_{2^{n-m-1}})$ gates can be simulated by $2(m-1)$ Toffoli gates and a basic operation, respectively. The form of the network is shown in (a) and (b) of Fig. 4.

Lemma 2. For any $n \geq 6$, $r = \lfloor n/2 \rfloor$ and $m \in \{r+1, r+2, \dots, n-2\}$, an N_n^m gate can be simulated by two N_n^r gates and two N_n^{m-r+1} gates.

For instance, the simulated networks of $C_6^0(I_{2^3} \otimes X)$ and $V_6^0(X \otimes I_{2^3})$ gates are shown in (c) and (d) of Fig. 4.

Lemma 3. When $n \geq 5$ and $m = \lceil n/2 \rceil$, an N_n^m gate can be simulated by a network consisting of $4(m-2)$ Toffoli gates.

For instance, the simulated networks of $C_m^0(I_{2^{n-m-1}} \otimes X)$ and $V_m^0(X \otimes I_{2^{n-m-1}})$ gates are shown in Fig. 5.

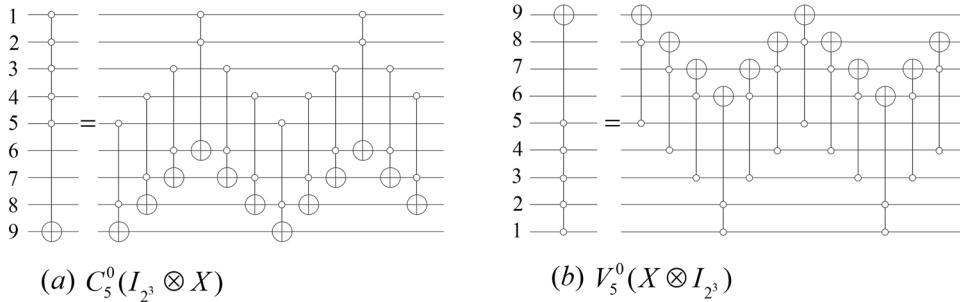


Figure 5. The controlled gate N_n^m illustrated for $n=9$ and $m=5$.

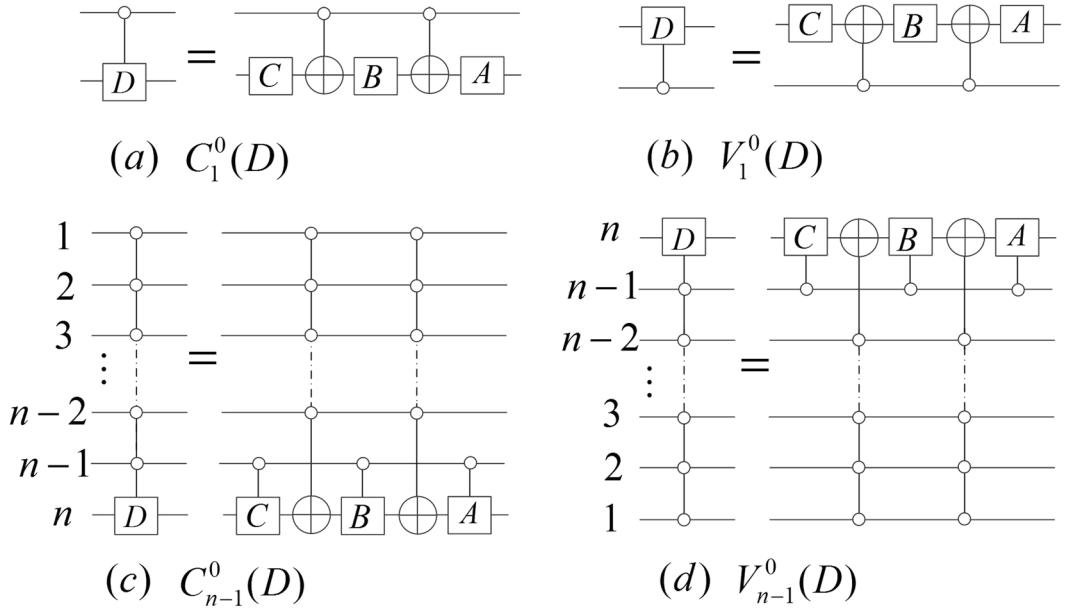


Figure 6. The simulated networks of the gates $C_{n-1}^0(D)$ and $V_{n-1}^0(D)$ with $n \geq 2$.

Lemma 4. For any $SU(2)$ matrix D , there exist $SU(2)$ matrices A , B , and C such that $ABC = I_2$ and $AXBXC = D$, and the gates $C_1^0(D)$ and $V_1^0(D)$ can be simulated by networks of the form shown in (a) and (b) Fig. 6. Here, $SU(2)$ is the Lie group of 2×2 unitary matrices with determinant 1.

More details of Lemmas 2, 3 and 4 are described in the reference²⁷. Next, we derive the following corollaries.

Corollary 1. For any $n \geq 7$, $r = \lfloor n/2 \rfloor$ and $m \in \{r+2, r+3, \dots, n-2\}$, an N_n^m gate can be simulated by $4(m-1)$ Toffoli gates and four basic operations.

Proof. Applying Lemma 2, an N_n^m gate can be simulated by two N_n^r gates and two N_n^{m-r+1} gates. Noting that $3 \leq r \leq \left\lfloor \frac{n}{2} \right\rfloor$ and $3 \leq m - r + 1 \leq \left\lfloor \frac{n}{2} \right\rfloor$, we apply lemma 1 so that the corollary holds. \square

Corollary 2. For any $n \geq 6$ and $r = \lfloor n/2 \rfloor$, an N_n^{r+1} gate can be simulated by $(4r-2)$ Toffoli gates and two basic operations when n is even, and $4(r-1)$ Toffoli gates when n is odd.

Proof. When n is odd, $r + 1 = \lceil n/2 \rceil$. Then, applying Lemma 3, we have that an N_n^{r+1} gate can be simulated by a network consisting of $4(r-1)$ Toffoli gates.

When n is even, by applying Lemma 2, it is derived that an N_n^{r+1} gate can be simulated by two N_n^r gates and two Toffoli gates. Then, by applying Lemma 1, it is proved that one can use $(4r - 2)$ Toffoli gates and two basic operations to simulate the N_n^{r+1} gate. \square

From lemma 4, the following corollary holds.

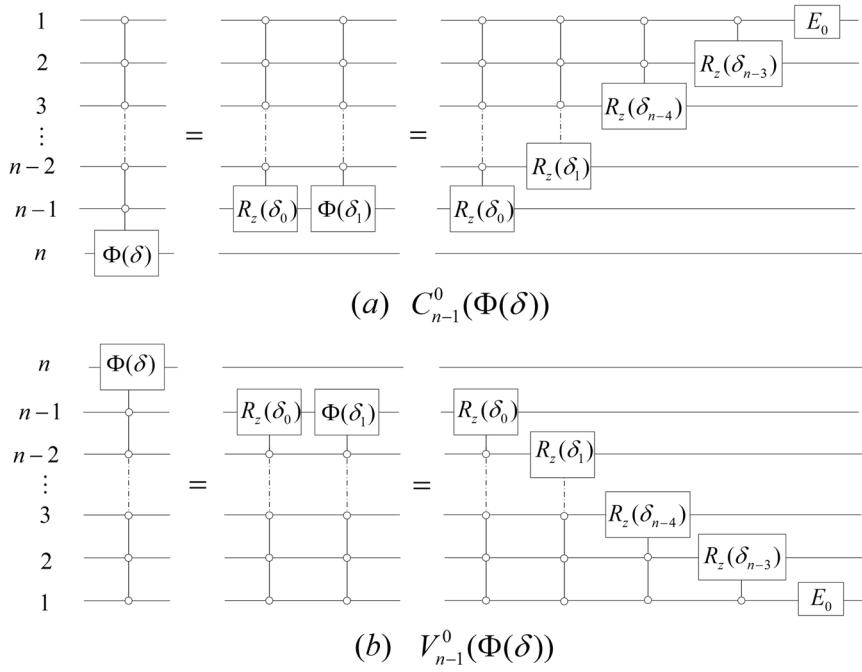


Figure 7. The simulated networks of the gates $C_{n-1}^0(\Phi(\delta))$ and $V_{n-1}^0(\Phi(\delta))$.

Corollary 3. For any $SU(2)$ matrix D , there exist $SU(2)$ matrices A , B , and C such that $ABC=I_2$ and $AXBXC=D$, and the gates $C_{n-1}^0(D)$ and $V_{n-1}^0(D)$ can be simulated by networks of the form shown in (c) and (d) of Fig. 6.

To analyze the complexities of the gates $C_{n-1}^0(X)$ and $V_{n-1}^0(X)$, we define three matrices:

$$\begin{cases} R_z(\theta) = \begin{bmatrix} e^{i\frac{\theta}{2}} & 0 \\ 0 & e^{-i\frac{\theta}{2}} \end{bmatrix}, \\ \Phi(\theta) = \begin{bmatrix} e^{i\theta} & 0 \\ 0 & e^{i\theta} \end{bmatrix}, \\ R_y(\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & \sin\frac{\theta}{2} \\ -\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}. \end{cases} \quad (12)$$

Lemma 5. Let $\delta_i = \frac{\delta}{2^i}$, $i \in \{0, 1, 2, \dots, n\}$, $E_0 = \Phi(\delta_{n-1})R_z(\delta_{n-2})$. Then, the gates $C_{n-1}^0(\Phi(\delta))$ and $V_{n-1}^0(\Phi(\delta))$ can be simulated by networks of the form shown in Fig. 7.

Proof. Note that

$$C_1^0(\Phi(\delta)) = \left(\Phi\left(\frac{\delta}{2}\right) R_z(\delta) \right) \otimes I_2 = (\Phi(\delta_1) R_z(\delta_0)) \otimes I_2, \quad (13)$$

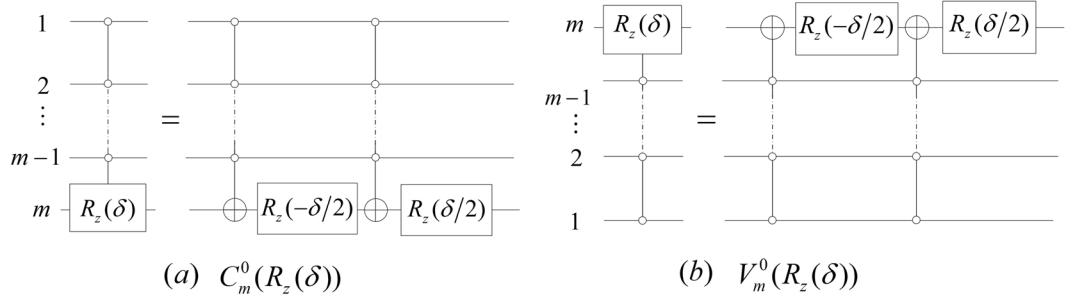
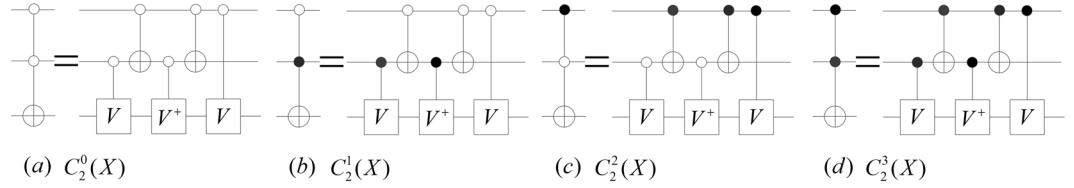
$$V_1^0(\Phi(\delta)) = I_2 \otimes \left(\Phi\left(\frac{\delta}{2}\right) R_z(\delta) \right) = I_2 \otimes (\Phi(\delta_1) R_z(\delta_0)). \quad (14)$$

Then, we have that

$$\begin{aligned} C_{n-1}^0(\Phi(\delta)) &= C_{n-2}^0(C_1^0(\Phi(\delta))) \\ &= C_{n-2}^0((\Phi(\delta_1) R_z(\delta_0)) \otimes I_2) \\ &= [C_{n-2}^0((\Phi(\delta_1)) C_{n-2}^0(R_z(\delta_0))] \otimes I_2. \end{aligned} \quad (15)$$

Similarly,

$$V_{n-1}^0(\Phi(\delta)) = I_2 \otimes [V_{n-2}^0((\Phi(\delta_1)) V_{n-2}^0(R_z(\delta_0)))] \quad (16)$$

**Figure 8.** The simulated networks of the gates $C_m^0(R_z(\delta))$ and $V_m^0(R_z(\delta))$.**Figure 9.** The simulated networks of the gates $C_2^i(X)$ ($i=0, 1, 2, 3$).

Therefore, we have the simulated networks of the gates $C_{n-1}^0(\Phi(\delta))$ and $V_{n-1}^0(\Phi(\delta))$ as shown in Fig. 7. \square

Lemma 6. The gates $C_m^0(R_z(\delta))$ and $V_m^0(R_z(\delta))$ can be simulated by networks of the form shown in Fig. 8.

Proof. Due to $R_z\left(\frac{\delta}{2}\right)R_z\left(-\frac{\delta}{2}\right) = I_2$ and $R_z\left(\frac{\delta}{2}\right)X R_z\left(-\frac{\delta}{2}\right)X = R_z(\delta)$, the conclusion is obvious. \square

$C_2^i(X)$ ($i=0, 1, 2, 3$) can be simulated by five basic operations shown in Fig. 9, i.e., $C(C_2^i(X)) = C^P(C_2^i(X)) = 5$.

Similarly, $C(V_2^i(X)) = C^P(V_2^i(X)) = 5$. Therefore, the complexity of Toffoli gates is 5. Thus, we obtain the complexity of $C_{n-1}^0(X)$ and $V_{n-1}^0(X)$ as described in theorem 1 below.

Theorem 1. For any $n \geq 7$, the gates $C_{n-1}^0(X)$ and $V_{n-1}^0(X)$ can be simulated by $(3.5n^2 - 13n - 4)$ Toffoli gates and $7n - 4$ basic operations when n is even, and by $(3.5n^2 - 12n - 5.5)$ Toffoli gates and $7n - 3$ basic operations when n is odd.

Proof. Let $\delta = \pi/2$, $D = R_z(-\pi)R_y(\pi)$, $A = R_z(-\pi)R_y(\pi)$, $B = R_y(-\pi/2)R_z(\pi/2)$ and $C = R_z(\pi/2)$. Then, $D, A, B, C \in SU(2)$, $ABC = I_2$, $AXBXC = D$ and $\Phi(\delta)D = X$.

Note that $C_{n-1}^0(X) = C_{n-1}^0(\Phi(\delta))C_{n-1}^0(D)$. Then,

$$C(C_{n-1}^0(X)) = C(C_{n-1}^0(\Phi(\delta))) + C(C_{n-1}^0(D)). \quad (17)$$

From lemma 4 and corollary 3, we obtain

$$C(C_{n-1}^0(D)) = 2C(N_n^{n-2}) + 15. \quad (18)$$

By lemma 5 and lemma 6, $C(C_{n-1}^0(\Phi(\delta)))$ can be computed by

$$C(C_{n-1}^0(\Phi(\delta))) = \sum_{i=1}^{n-2} C(C_i^0(R_z(\delta_{n-2-i}))) + 1 = 2 \sum_{i=1}^{n-2} C(N_n^i) + 2n - 3. \quad (19)$$

Therefore,

$$C(C_{n-1}^0(X)) = \sum_{i=3}^r 2C(N_n^i) + \sum_{i=r+2}^{n-3} 2C(N_n^i) + 4C(N_n^{n-2}) + 2C(N_n^{r+1}) + 2C(N_3^2) + 2n + 14, \quad (20)$$

where $r = \lfloor n/2 \rfloor$ and N_3^2 is a Toffoli gate.

Applying lemma 1, corollary 1 and corollary 2, we obtain

$$C(C_{n-1}^0(X)) = \begin{cases} (3.5n^2 - 13n - 4)C(N_3^2) + 7n - 4, & \text{when } n \text{ is even}, \\ (3.5n^2 - 12n - 5.5)C(N_3^2) + 7n - 3, & \text{when } n \text{ is odd}. \end{cases} \quad (21)$$

Similarly, we obtain that $C(V_{n-1}^0(X)) = C(C_{n-1}^0(X))$. \square

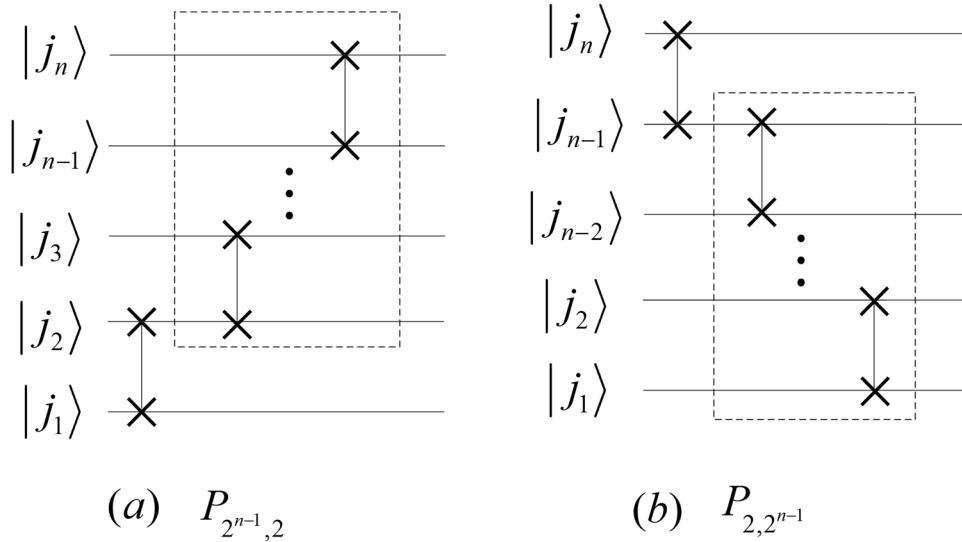


Figure 10. The implement circuits of $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$. The dotted boxes in (a,b) are the circuits of $P_{2^{n-2},2}$ and $P_{2,2^{n-2}}$, respectively.

Comparing with the methods proposed in²⁷, the complexities of our proposed simulated networks of these gates $N_n^i (3 \leq i \leq n - 1)$ are reduced by 50% approximately.

The Quantum Circuits of PSP

The perfect shuffle permutation $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$ can be expressed as

$$\begin{cases} P_{2^{n-1},2} = (P_{2^{n-2},2} \otimes I_2) (I_{2^{n-2}} \otimes P_{2,2}), \\ P_{2,2^{n-1}} = (I_2 \otimes P_{2,2^{n-2}}) (P_{2,2} \otimes I_{2^{n-2}}), \end{cases} \quad (22)$$

where $P_{2,2}$ is a Swap gate, and their implementation circuits are shown in Fig. 10.

Applying $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$ to the state $|j_n j_{n-1} \cdots j_2 j_1\rangle$, we have

$$\begin{cases} P_{2^{n-1},2} |j_n j_{n-1} \cdots j_2 j_1\rangle = |j_1 j_n j_{n-1} \cdots j_2\rangle, \\ P_{2,2^{n-1}} |j_n j_{n-1} \cdots j_2 j_1\rangle = |j_{n-1} \cdots j_2 j_1 j_n\rangle. \end{cases} \quad (23)$$

Let $\Gamma_{2^n} = P_{2^{n-1},2} (P_{2^{n-2},2} \otimes I_2) \dots (P_{2^2,2} \otimes I_{2^{n-3}}) (P_{2,2} \otimes I_{2^{n-2}})$, we have that $(\Gamma_{2^n})^{-1} = (P_{2,2} \otimes I_{2^{n-2}}) (P_{2,2} \otimes I_{2^{n-3}}) \dots (P_{2,2} \otimes I_{2^2}) (P_{2,2} \otimes I_2) P_{2,2^{n-1}}$ and

$$\Gamma_{2^n} |j_n j_{n-1} \cdots j_2 j_1\rangle = |j_1 j_2 \cdots j_{n-2} j_{n-1} j_n\rangle = (\Gamma_{2^n})^{-1} |j_n j_{n-1} j_{n-2} \cdots j_2 j_1\rangle \quad (24)$$

Therefore, we conclude that $\Gamma_{2^n} = (\Gamma_{2^n})^{-1}$ and design quantum circuits shown in Fig. 11.

The costs of the circuits of Γ_{2^n} and $(\Gamma_{2^n})^{-1}$ are

$$C(\Gamma_{2^n}) = C((\Gamma_{2^n})^{-1}) = 3 \times \left\lfloor \frac{n}{2} \right\rfloor. \quad (25)$$

By parallel computing, we redesign the circuits of Γ_{2^n} and $(\Gamma_{2^n})^{-1}$ shown in Fig. 12 and calculate time complexities by

$$C^p(\Gamma_{2^n}) = C^p((\Gamma_{2^n})^{-1}) = C(Swap) = 3. \quad (26)$$

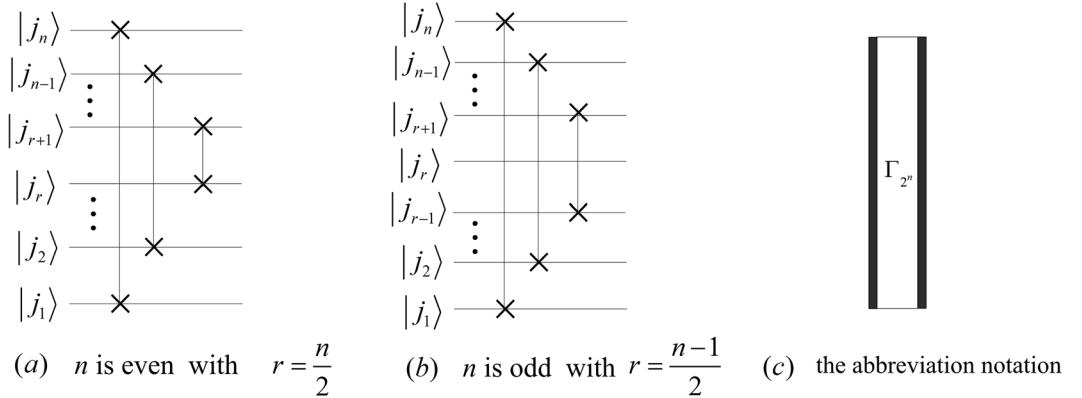
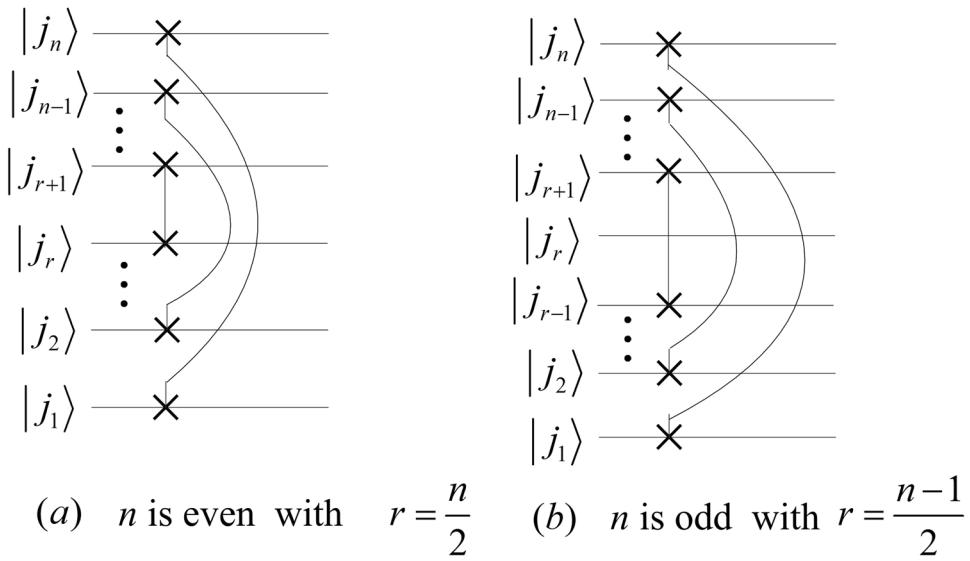
i.e., complexities of Γ_{2^n} and $(\Gamma_{2^n})^{-1}$ are $O(1)$.

The iterations of Γ_{2^n} and $(\Gamma_{2^n})^{-1}$ are given by

$$\begin{cases} \Gamma_{2^n} = P_{2^{n-1},2} (\Gamma_{2^{n-1}} \otimes I_2), \\ (\Gamma_{2^n})^{-1} = ((\Gamma_{2^{n-1}})^{-1} \otimes I_2) P_{2,2^{n-1}}. \end{cases} \quad (27)$$

Then, we obtain

$$\begin{cases} P_{2^{n-1},2} = \Gamma_{2^n} (\Gamma_{2^{n-1}} \otimes I_2), \\ P_{2,2^{n-1}} = (\Gamma_{2^{n-1}} \otimes I_2) \Gamma_{2^n}. \end{cases} \quad (28)$$

**Figure 11.** The simplified quantum circuits of Γ_{2^n} and $(\Gamma_{2^n})^{-1}$.**Figure 12.** The parallel quantum circuits of Γ_{2^n} and $(\Gamma_{2^n})^{-1}$.

Therefore, we design the simplified circuits of $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$ as shown in Fig. 13. The complexities of $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$ are

$$\begin{cases} C(P_{2^{n-1},2}) = C(P_{2,2^{n-1}}) = 3(n-1), \\ C^P(P_{2^{n-1},2}) = C^P(P_{2,2^{n-1}}) = 6. \end{cases} \quad (29)$$

The reason that the abbreviation notations in Fig. 13 are the same except for the positions of black boxes is due to the fact that the circuit in Fig. 13(b) consists of the gates in Fig. 13(a) but rearranged in reverse order. We also adopt similar abbreviation notations to denote the circuits that are composed of the same quantum gates with reverse order in the following sections.

The iterations of $P_{2^n,2^{m-1}}$ and $P_{2^{m-1},2^n}$ are given by

$$\begin{cases} P_{2^n,2^{m-1}} = (P_{2,2^{m-1}} \otimes I_{2^{n-1}}) (I_2 \otimes P_{2^{n-1},2^{m-1}}), \\ P_{2^{m-1},2^n} = (I_2 \otimes P_{2^{m-1},2^{n-1}}) (P_{2^{m-1},2} \otimes I_{2^{n-1}}). \end{cases} \quad (30)$$

According to (30), we design the implementation circuits of $P_{2^{m-1},2^n}$ and $P_{2^n,2^{m-1}}$ in Fig. 14. The complexities of the circuits in Fig. 14 are

$$\begin{cases} C(P_{2^{m-1},2^n}) = C(P_{2^n,2^{m-1}}) = 3n(m-1), \\ C^P(P_{2^{m-1},2^n}) = C^P(P_{2^n,2^{m-1}}) = 6n. \end{cases} \quad (31)$$

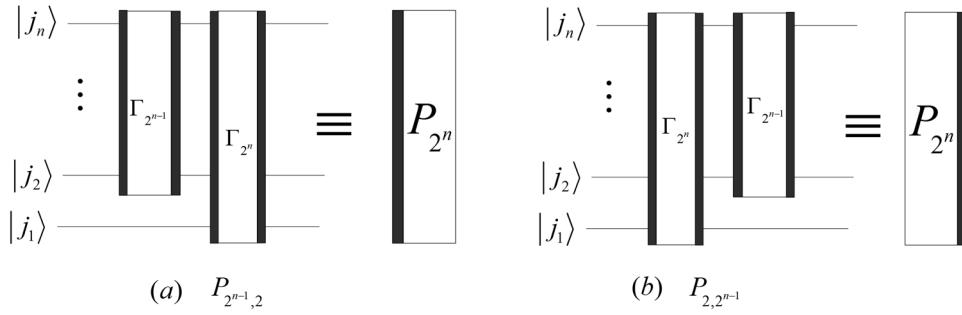


Figure 13. The simplified circuits of $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$. The rights of (a,b) correspond to the abbreviation notation of $P_{2^{n-1},2}$ and $P_{2,2^{n-1}}$, respectively.

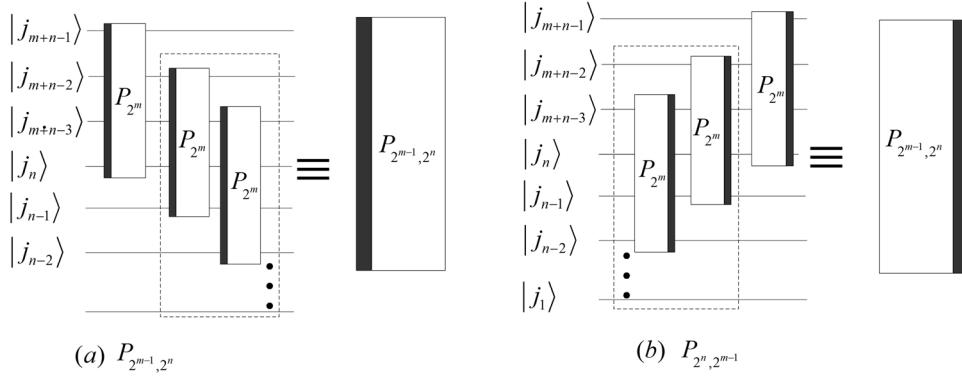


Figure 14. The quantum circuits of $P_{2^{m-1},2^n}$ and $P_{2^n,2^{m-1}}$. The dotted boxes in (a,b) are the implement circuits of $P_{2^{m-1},2^{n-1}}$ and $P_{2^n,2^{m-1}}$, respectively.

The Implementation of QWPT

Let $W_{2^n}^0 = W_{2^n}$ be a wavelet kernel matrix. Then, the $(k+1)$ -th iteration of a discrete wavelet packet transform is defined by

$$\begin{cases} Z_{2^n}^k = W_{2^n}^k W_{2^n}^{k-1} \dots W_{2^n}^1 W_{2^n}^0, \\ W_{2^n}^j = \text{Diag}(W_{2^{n-j}}, W_{2^{n-j}}, \dots, W_{2^{n-j}}), \end{cases} \quad (32)$$

where $j=1, \dots, k$ and $\text{Diag}(W_{2^{n-j}}, W_{2^{n-j}}, \dots, W_{2^{n-j}})$ is a matrix with 2^j blocks of $W_{2^{n-j}}$ on the main diagonal and zeros elsewhere.

The following equations

$$\begin{cases} W_{2^n}^j = \text{Diag}(W_{2^{n-j}}, W_{2^{n-j}}), \\ Z_{2^n}^k = \text{Diag}(Z_{2^{n-j}}^{k-1}, Z_{2^{n-j}}^{k-1}) W_{2^n}^0 \end{cases} \quad (33)$$

can be derived by (32).

Since

$$Z_{2^{n-1}}^{k-1} \otimes I_2 = P_{2,2^{n-1}} \text{Diag}(Z_{2^{n-1}}^{k-1}, Z_{2^{n-1}}^{k-1}) P_{2^{n-1},2}, \quad (34)$$

the iteration equation of the QWPT is given by

$$Z_{2^n}^k = P_{2^{n-1},2} (Z_{2^{n-1}}^{k-1} \otimes I_2) P_{2,2^{n-1}} W_{2^n}^0 \quad (35)$$

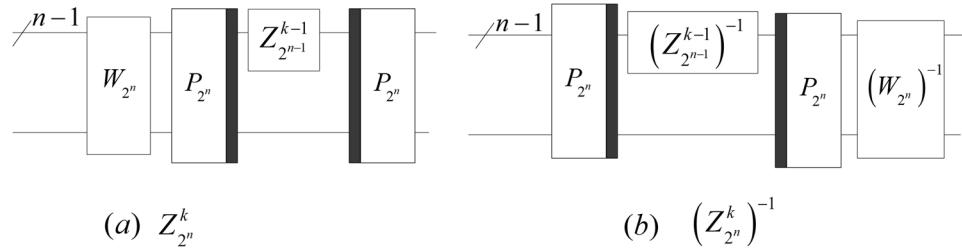
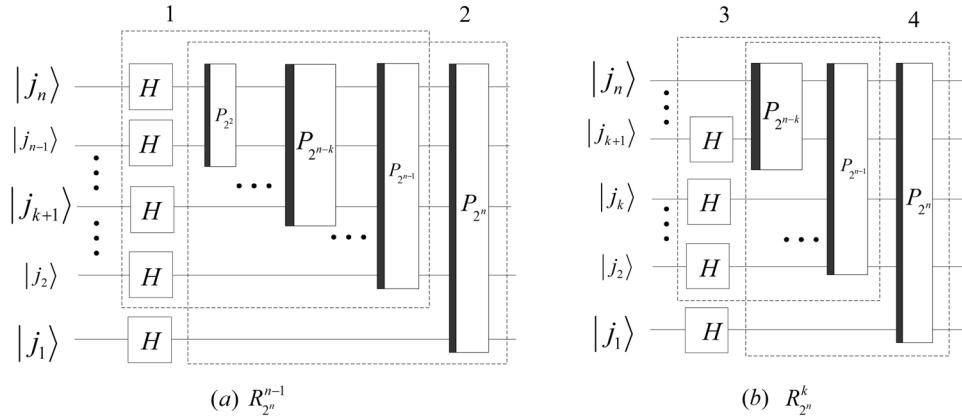
with the initial value $Z_{2^{n-k}}^0 = W_{2^{n-k}}$ and the implementation circuit shown in (a) of Fig. 15.

Similarly, the inverse of $Z_{2^n}^k$ is

$$(Z_{2^n}^k)^{-1} = (W_{2^n})^{-1} P_{2^{n-1},2} ((Z_{2^{n-1}}^{k-1})^{-1} \otimes I_2) P_{2,2^{n-1}} \quad (36)$$

with the initial value $(Z_{2^{n-k}}^0)^{-1} = (W_{2^{n-k}})^{-1}$ and the implementation circuit of $(Z_{2^n}^k)^{-1}$ shown in (b) of Fig. 15.

Next, we describe the implementations of the Haar QWPT (HQWPT) and the D4 QWPT (DQWPT) in detail.

**Figure 15.** The implementation circuits of $Z_{2^n}^k$ and $(Z_{2^n}^k)^{-1}$.**Figure 16.** The implementation circuits of $R_{2^n}^{n-1}$ and $R_{2^n}^k$ ($1 \leq k < n - 1$). The dashed box 1 and box 3 implement $R_{2^n}^{n-2}$ and $R_{2^{n-1}}^{k-1}$, respectively.

The implementation of HQWPT. Substituting the kernel matrix $W_{2^n} = P_{2^{n-1},2}(I_{2^{n-1}} \otimes H)$ into equations (35) and (36), the ($k + 1$)-th iteration of HQWPT and its inverse are

$$\begin{cases} R_{2^n}^k = P_{2^{n-1},2}(R_{2^{n-1}}^{k-1} \otimes I_2)(I_{2^{n-1}} \otimes H), \\ (R_{2^n}^k)^{-1} = (I_{2^{n-1}} \otimes H)((R_{2^{n-1}}^{k-1})^{-1} \otimes I_2)P_{2,2^{n-1}} \end{cases} \quad (37)$$

with the initial values

$$\begin{cases} R_{2^{n-k}}^0 = P_{2^{n-k-1},2}(I_{2^{n-k-1}} \otimes H), 1 \leq k < n - 1, \\ R_2^0 = H, k = n - 1, \\ (R_{2^{n-k}}^0)^{-1} = (H \otimes I_{2^{n-k-1}})P_{2,2^{n-k-1}}, 1 \leq k < n - 1, \\ (R_2^0)^{-1} = H, k = n - 1. \end{cases}$$

The quantum circuits of $R_{2^n}^{n-1}$ and $R_{2^n}^k$ ($1 \leq k < n - 1$) are designed in Fig. 16.

Since $P_{2^{n-1},2}(P_{2^{n-2},2} \otimes I_2) \dots (P_{2,2} \otimes I_{2^{n-2}}) = \Gamma_{2^n}$ and $P_{2^{n-1},2}(P_{2^{n-2},2} \otimes I_2) \dots (P_{2^{n-k-1},2} \otimes I_{2^k}) = \Gamma_{2^n}((\Gamma_{2^{n-k-1}})^{-1} \otimes I_{2^{k+1}})$ with $1 \leq k < n - 2$, the quantum circuit of $R_{2^n}^k$, $R_{2^n}^{n-2}$ and $R_{2^n}^{n-1}$ can be simplified and shown in Fig. 17.

Similarly, the quantum circuits of the inverses of $R_{2^n}^k$, $R_{2^n}^{n-2}$ and $R_{2^n}^{n-1}$ can be designed as shown in Fig. 18. The costs of HQWPT are

$$\begin{cases} C(R_{2^n}^k) = C((R_{2^n}^k)^{-1}) = 3\left\lfloor \frac{n}{2} \right\rfloor + 3\left\lfloor \frac{n-k-1}{2} \right\rfloor + k + 1, \\ C(R_{2^n}^{n-2}) = C((R_{2^n}^{n-2})^{-1}) = 3\left\lfloor \frac{n}{2} \right\rfloor + n - 1, \\ C(R_{2^n}^{n-1}) = C((R_{2^n}^{n-1})^{-1}) = 3\left\lfloor \frac{n}{2} \right\rfloor + n, \end{cases} \quad (38)$$

where $1 \leq k < n - 2$. Since $C^p(R_{2^n}^k) = C^p((R_{2^n}^k)^{-1}) = 6$, $1 \leq k < n - 2$ and $C^p(R_{2^n}^k) = C^p((R_{2^n}^k)^{-1}) = 4$, $n - 2 \leq k \leq n - 1$, the time complexity of the HQWPT is $O(1)$.

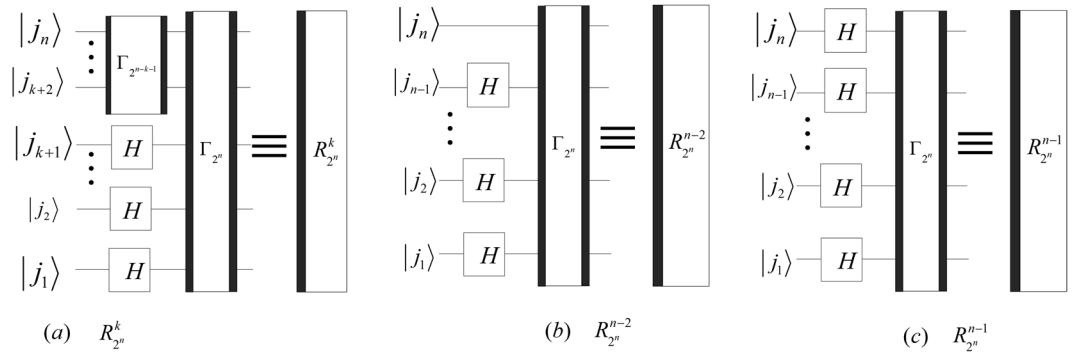


Figure 17. The simplified circuits of $R_{2^n}^k$, $1 \leq k < n - 2$, $R_{2^n}^{n-2}$ and $R_{2^n}^{n-1}$.

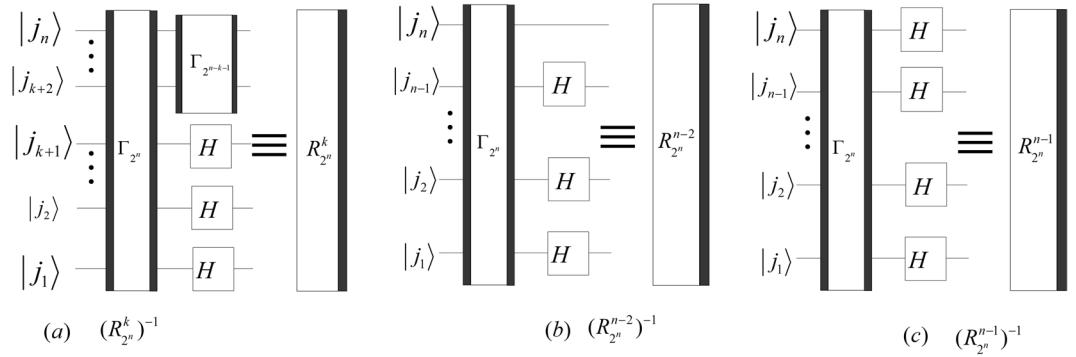


Figure 18. The simplified circuits of $(R_{2^n}^k)^{-1}$, $1 \leq k < n - 2$, $(R_{2^n}^{n-2})^{-1}$ and $(R_{2^n}^{n-1})^{-1}$.

The implementation of DQWPT. The kernel matrix of the D4 wavelet transform is defined by the reference³¹

$$D_{2^n} = \begin{bmatrix} h_0 & h_1 & h_2 & h_3 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ h_3 & -h_2 & h_1 & -h_0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & h_0 & h_1 & h_2 & h_3 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & h_3 & -h_2 & h_1 & -h_0 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & h_0 & h_1 & h_2 & h_3 \\ 0 & 0 & 0 & 0 & 0 & 0 & \cdots & h_3 & -h_2 & h_1 & -h_0 \\ h_2 & h_3 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & h_0 & h_1 \\ h_1 & -h_0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & h_3 & -h_2 \end{bmatrix}, \quad (39)$$

where $h_0 = \frac{1 + \sqrt{3}}{4\sqrt{2}}$, $h_1 = \frac{3 + \sqrt{3}}{4\sqrt{2}}$, $h_2 = \frac{3 - \sqrt{3}}{4\sqrt{2}}$ and $h_3 = \frac{1 - \sqrt{3}}{4\sqrt{2}}$. D_{2^n} and $(D_{2^n})^{-1}$ can be rewritten to

$$\begin{cases} D_{2^n} = (I_{2^{n-1}} \otimes S_1) Q_{2^n} (I_{2^{n-1}} \otimes S_0), \\ Q_{2^n} = (\{Q_{2^{n-1}}, I_{2^{n-1}}\} \otimes I_2), \\ (D_{2^n})^{-1} = (I_{2^{n-1}} \otimes S_0) (Q_{2^n})^{-1} (I_{2^{n-1}} \otimes S_1), \\ (Q_{2^n})^{-1} = (\{(Q_{2^{n-1}})^{-1}, I_{2^{n-1}}\} \otimes I_2) (I_{2^{n-1}} \otimes X), \end{cases} \quad (40)$$

where

$$\begin{cases} S_0 = \begin{bmatrix} \sin(2\pi/3) & \cos(2\pi/3) \\ \cos(2\pi/3) & -\sin(2\pi/3) \end{bmatrix} \\ S_1 = \begin{bmatrix} -\cos(\pi/12) & \sin(\pi/12) \\ \sin(\pi/12) & \cos(\pi/12) \end{bmatrix} \\ Q_2 = (Q_2)^{-1} = X, \end{cases}$$

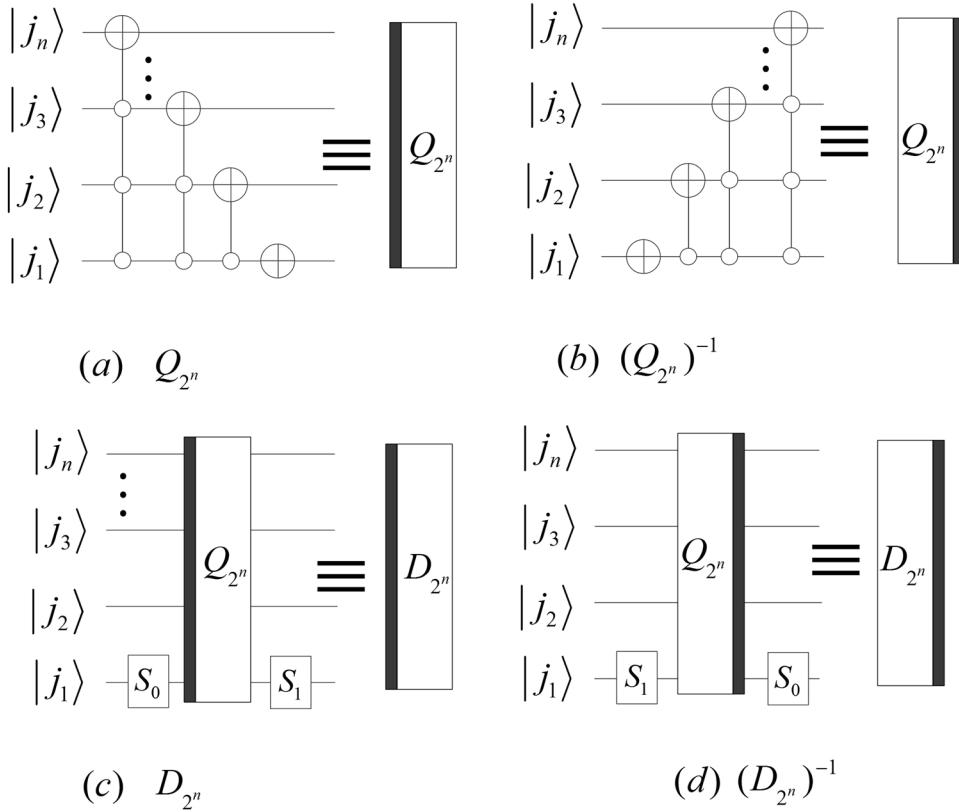


Figure 19. The quantum circuits of the kernel matrix of the D4 wavelet transform.

and the implementation circuits shown in Fig. 19.

In order to implement a multi-level DQWPT based on the periodization extension, a single-level DQWPT and its inverse are given by:

$$\begin{cases} T_{2^n} = P_{2^{n-1},2}D_{2^n}^P, \\ D_{2^n}^P = D_{2^n}(Q_{2^n})^{-1}, \\ (T_{2^n})^{-1} = (D_{2^n}^P)^{-1}P_{2,2^{n-1}}, \\ (D_{2^n}^P)^{-1} = Q_{2^n}(D_{2^n})^{-1}. \end{cases} \quad (41)$$

The implement circuits of the above DQWPT are shown in Fig. 20. Substituting the kernel matrix W_{2^n} with T_{2^n} in (35) and (36), we obtain that the $(k+1)$ -th iterations of the DQWPT and its inverse based on the periodization extension are

$$\begin{cases} A_{2^n}^k = P_{2^{n-1},2}(A_{2^{n-1}}^{k-1} \otimes I_2)D_{2^n}^P, \\ (A_{2^n}^k)^{-1} = Q_{2^n}(D_{2^n})^{-1}((A_{2^{n-1}}^{k-1})^{-1} \otimes I_2)P_{2,2^{n-1}} \end{cases} \quad (42)$$

with the initial values $A_{2^{n-k}}^0 = T_{2^{n-k}}, (A_{2^{n-k}}^0)^{-1} = (T_{2^{n-k}})^{-1}, 1 \leq k < n-1$ and their implementation circuits shown in Fig. 21.

Using Γ_{2^n} , the quantum circuit of $A_{2^n}^k$ and $(A_{2^n}^k)^{-1}$ can be simplified and shown in Fig. 22.

We analyze the complexity of the above DQWPT and suppose $r = \lfloor n/2 \rfloor$.

From Figs 19 and 20, we calculate the complexity of T_{2^n} by

$$C^P(T_{2^n}) = C^P((T_{2^n})^{-1}) = C^P(Q_{2^n}) + C^P(D_{2^n}) + C^P(P_{2^{n-1},2}) = \sum_{i=1}^{n-1} 2C(N_n^i) + 10. \quad (43)$$

Applying lemma 1, corollary 1, corollary 2 and theorem 1, we obtain

$$C^P(T_{2^n}) = C^P((T_{2^n})^{-1}) = \begin{cases} 52.5n^2 - 216n + 38, & n \text{ is even,} \\ 52.5n^2 - 201n + 16.5, & n \text{ is odd.} \end{cases} \quad (44)$$

We calculate the quantum cost of T_{2^n} by

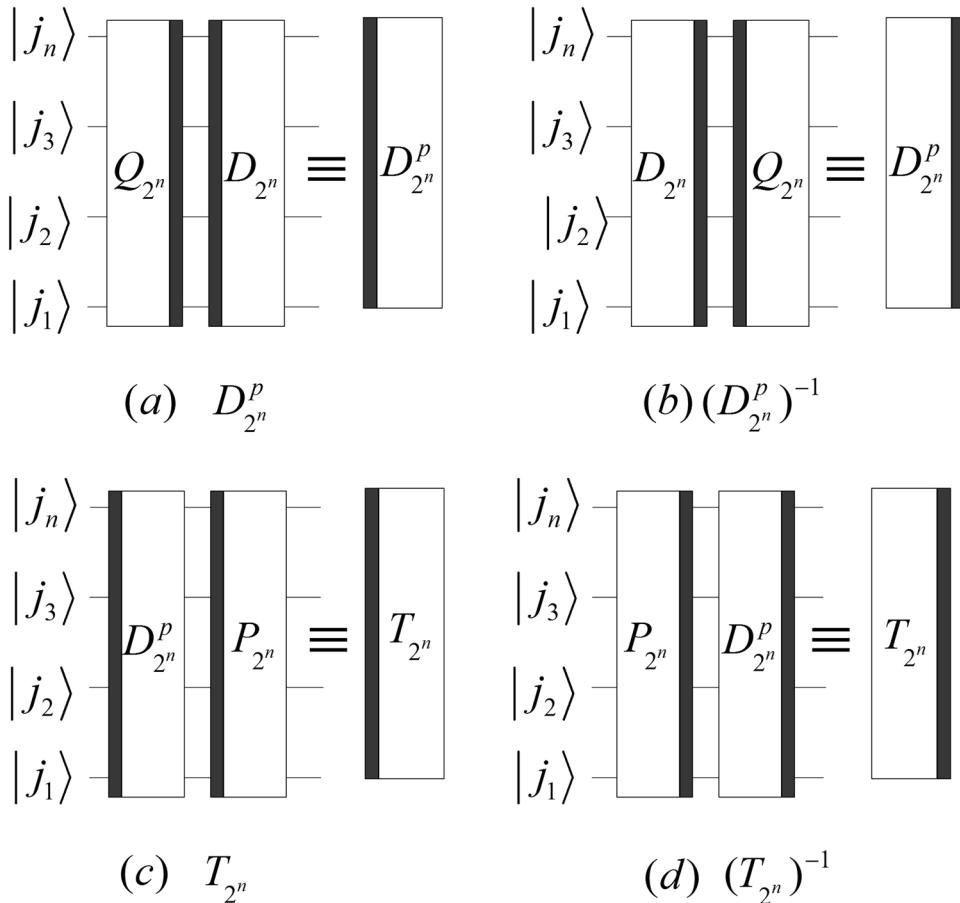


Figure 20. The quantum circuits of the single-level DQWPT and its inverse based on the periodization extension.

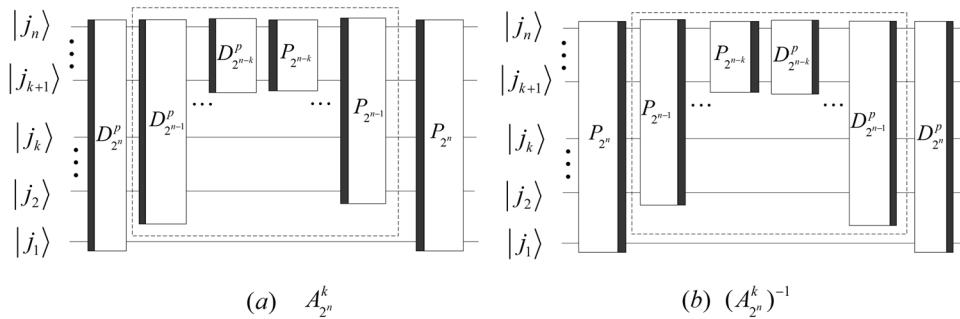


Figure 21. The quantum circuits of $A_{2^n}^k$ and $(A_{2^n}^k)^{-1}$. The dashed boxes in (a,b) implement $A_{2^{n-1}}^{k-1}$ and $(A_{2^{n-1}}^{k-1})^{-1}$, respectively.

$$C(T_{\lambda^n}) = C((T_{\lambda^n})^{-1}) = C(Q_{\lambda^n}) + C(D_{\lambda^n}) + C(P_{\lambda^{n-1}, 2}) = C^p(T_{\lambda^n}) + 3n - 9. \quad (45)$$

Let $\phi_2(k) = \begin{cases} 3, & k = n - 2 \\ 6, & 1 \leq k \leq n - 2, \end{cases}$ the time complexity of DQWPT is

$$C^p(A_{2^n}^k) = C^p((A_{2^n}^k)^{-1}) = \sum_{i=1}^{n-k-1} 2(k+1)C(N_n^i) + \sum_{i=n-k}^{n-1} 2(n-i)C(N_n^i) + 4(k+1) + \phi_2(k). \quad (46)$$

For instance,

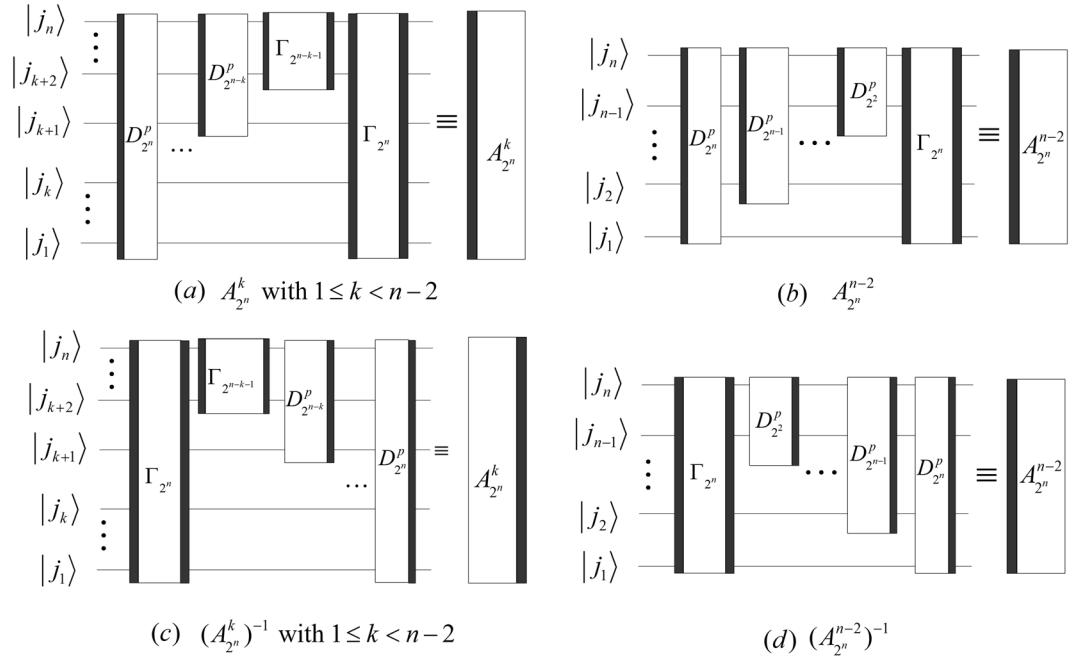


Figure 22. The simplified circuits of $A_{2^n}^k$ and $(A_{2^n}^k)^{-1}$ with $1 \leq k < n - 1$.

$$C^p(A_{2^n}^{n-2}) = C^p((A_{2^n}^{n-2})^{-1}) = \begin{cases} 5n^3 + 21n^2 - 163.5n + 63, & n \text{ is even,} \\ 5n^3 + 23.5n^2 - 168.5n + 51, & n \text{ is odd,} \end{cases} \quad (47)$$

and

$$C^p(A_{2^n}^1) = C^p((A_{2^n}^1)^{-1}) = \begin{cases} 70n^2 - 316n + 115, & n \text{ is even,} \\ 70n^2 - 296n + 85, & n \text{ is odd.} \end{cases} \quad (48)$$

The costs of $A_{2^n}^k$ and $(A_{2^n}^k)^{-1}$ are

$$C(A_{2^n}^k) = C((A_{2^n}^k)^{-1}) = C^p(A_{2^n}^k) - \phi_2(k) + \left\lfloor \frac{n-k-1}{2} \right\rfloor + \left\lfloor \frac{n}{2} \right\rfloor. \quad (49)$$

The 2D and 3D QWPTs

Firstly, we briefly describe NASS to represent 2D images and 3D videos. The NASS state $|\psi_2\rangle$ of an image can be represented by

$$|\psi_2\rangle = \sum_{x_m=0}^{2^m-1} \sum_{y_k=0}^{2^k-1} \theta_{x_m, y_k} |x_m\rangle |y_k\rangle, \quad (50)$$

where $|x_m\rangle = |i_n \dots i_{k+1}\rangle$ and $|y_k\rangle = |i_k \dots i_1\rangle$ are the X-axis and Y-axis of the image, θ_{x_m, y_k} represents the color of the pixel in the coordinate $|x_m\rangle |y_k\rangle$, and $n = m + k$.

The NASS state $|\psi_3\rangle$ of a video can be represented by

$$|\psi_3\rangle = \sum_{x_m=0}^{2^m-1} \sum_{y_k=0}^{2^k-1} \sum_{t_h=0}^{2^h-1} \theta_{x_m, y_k, t_h} |x_m\rangle |y_k\rangle |t_h\rangle, \quad (51)$$

where $|x_m\rangle = |i_n \dots i_{h+k+1}\rangle$, $|y_k\rangle = |i_{h+k} \dots i_{h+1}\rangle$ and $|t_h\rangle = |i_h \dots i_1\rangle$ are the X-axis, Y-axis and time-axis of a video, and $n = m + k + h$.

More details are shown in our previous work⁶. For instance, the NASS state

$$|\psi_2\rangle = \sum_{x_3=0}^{2^3-1} \sum_{y_2=0}^{2^2-1} \theta_{x_3, y_2} |x_3\rangle |y_2\rangle = \theta_{0,0} |000\rangle |00\rangle + \dots + \theta_{7,3} |111\rangle |11\rangle \quad (52)$$

represents the color image of 8×4 (height multiplies weight) as shown in (a) of Fig. 23.

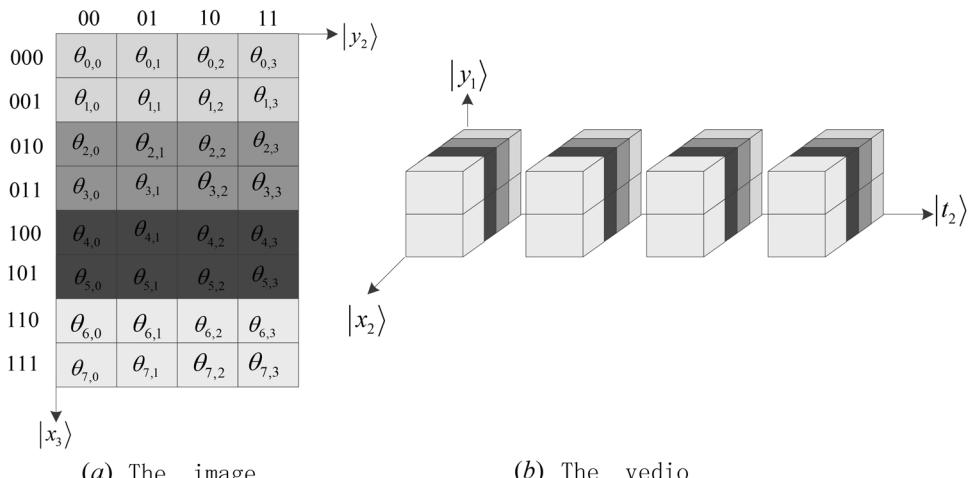


Figure 23. The image and the video.

The NASS state

$$|\psi_3\rangle = \sum_{x_2=0}^{2^2-1} \sum_{y_1=0}^1 \sum_{t_2=0}^{2^2-1} \theta_{x_m y_k t_h} |x_2\rangle |y_1\rangle |t_2\rangle = \theta_{0,0,0} |00\rangle |0\rangle |00\rangle + \dots + \theta_{3,1,3} |11\rangle |1\rangle |11\rangle \quad (53)$$

represents the video with four frames as shown in (b) of Fig. 23, where each frame is a 4×2 image.

The same string can have different meanings corresponding to different data types in classic computers. For instance, a binary string 0100001 can represent a char 'A' or a number 65. Similarly, using the circuit in⁶, we can store an image (shown in (a) of Fig. 23) or a video (shown in (b) of Fig. 23) in the following state

$$|\psi\rangle = \sum_{i=0}^{2^5-1} \theta_i |i\rangle. \quad (54)$$

Meanwhile, the priori knowledge ' x_3, y_2 ' or ' x_2, y_1, t_2 ' is equivalent to a data type, implying an image or a video stored in the state $|\psi\rangle$.

A natural image with size of $2^n \times 2^m$ can be expressed as an angle matrix

$$\Lambda_{2^n, 2^m} = \begin{bmatrix} \theta_{0,0} & \theta_{0,1} & \cdots & \theta_{0,2^m-1} \\ \theta_{1,0} & \theta_{1,1} & \cdots & \theta_{1,2^m-1} \\ \vdots & \vdots & \cdots & \vdots \\ \theta_{2^n-1,0} & \theta_{2^n-1,1} & \cdots & \theta_{2^n-1,2^m-1} \end{bmatrix}, \quad (55)$$

where $\theta_{x,y}$ is the color information of the pixel on the coordinate (x, y) and an example is shown in Fig. 23.

Thus, the 2D wavelet transform on $\Lambda_{2^n, 2^m}$ is defined as

$$wt2(\Lambda_{2^n} \gamma^m) = W_{2^n} \times \Lambda_{2^n} \gamma^m \times W_{2^m}^T, \quad (56)$$

where W_{2^n} and W_{2^m} are $2^n \times 2^n$ and $2^m \times 2^m$ wavelet transforms, respectively.

An image can be stored in the state $NASS|\psi_2\rangle$ in (50) by using a quantum circuit in the literature⁶. Suppose that the function $f(\cdot)$ is equivalent to the quantum circuit implementing the storage of the image $\Lambda_{\gamma^n \gamma^m}$, that is,

$$f(\Lambda_{2^n, 2^m}) = |\psi_2\rangle = \begin{bmatrix} B_0^T \\ \vdots \\ B_{2^n-1}^T \end{bmatrix}, \quad (57)$$

where $B_i = [\theta_{i,0} \ \theta_{i,1} \ \dots \ \theta_{i,2^m-1}]$ is the row vector of $\Lambda_{2^n,2^m}$ and $0 \leq i \leq 2^n - 1$.

Applying the function $f(\cdot)$ on $\Lambda_{2^n, 2^m} \times W_{2^m}^T$, the result is

$$f(\Lambda_{2^n} \times W_{2^m}) = (I_{2^n} \otimes W_{2^m})f(\Lambda_{2^n} \times W_{2^m}). \quad (58)$$

Using the perfect shuffle permutation $P_{2^m \cdot 2^n}$, we obtain

$$f(\Lambda_{2^n 2^m}) \equiv P_{2^m 2^n} f((\Lambda_{2^n 2^m})^T). \quad (59)$$

Then, we have

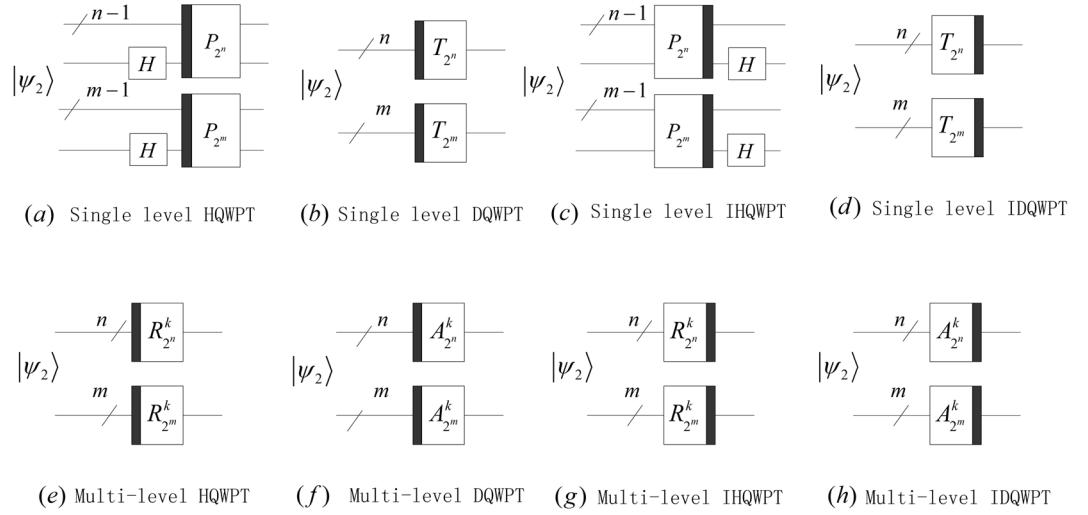


Figure 24. The quantum circuits of the 2D QWPT and IQWPT with $1 \leq k \leq \min(m, n) - 1$ in (e,f), $1 \leq k \leq \min(m, n) - 2$ in (f,h).

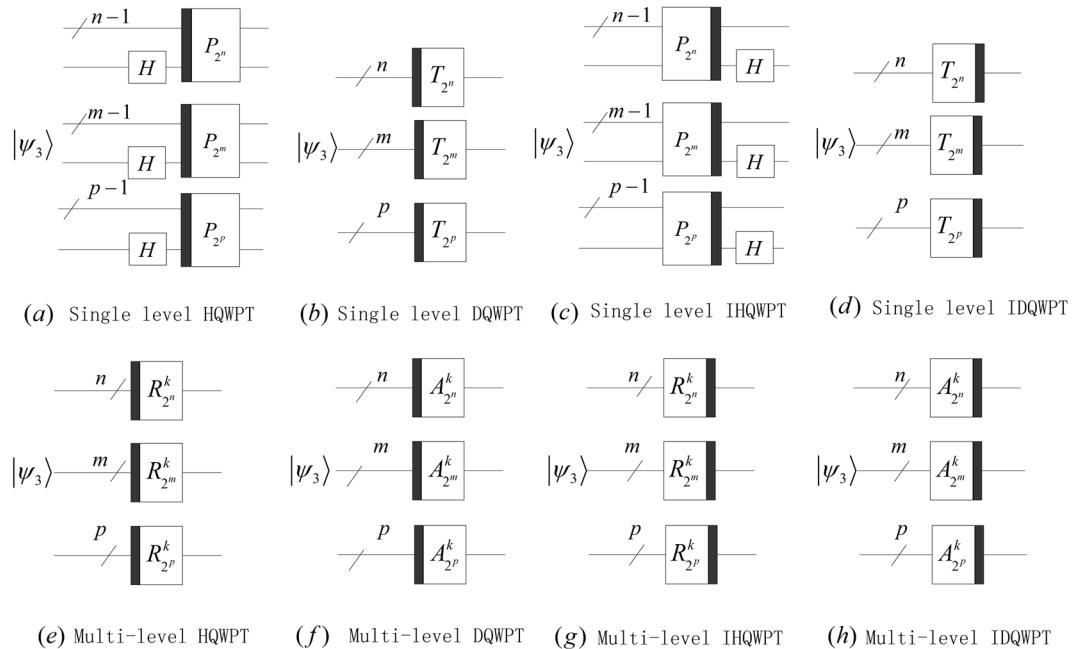


Figure 25. The quantum circuits of the 2D QWPT and IQWPT with $1 \leq k \leq \min(m, n, p) - 1$ in (e,f), $1 \leq k \leq \min(m, n, p) - 2$ in (f,h).

$$f(W_{2^n}\Lambda_{2^n,2^m}) = (W_{2^n} \otimes I_{2^m})f(\Lambda_{2^n,2^m}), \quad (60)$$

$$f(W_{2^n}\Lambda_{2^n,2^m}W_{2^m}^T) = (W_{2^n} \otimes W_{2^m})f(\Lambda_{2^n,2^m}). \quad (61)$$

Then, the 2D QWPT of $\Lambda_{2^n,2^m}$ is given by

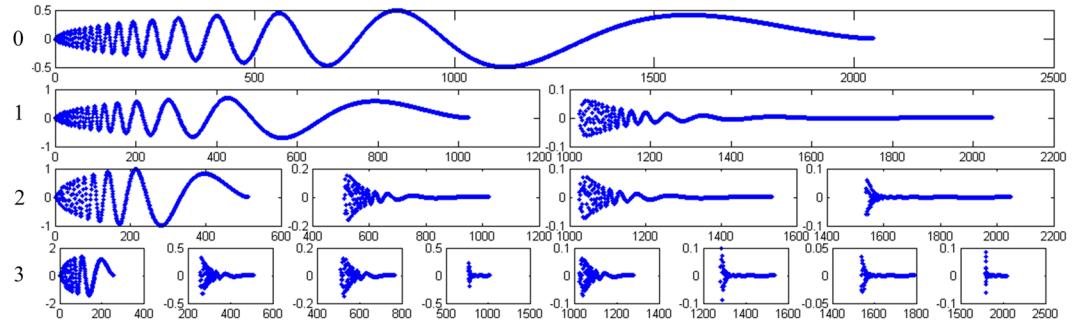
$$f(wt2(\Lambda_{2^n,2^m})) = (W_{2^n} \otimes W_{2^m})|\psi_2\rangle. \quad (62)$$

A video of 2^p frames of size $2^n \times 2^m$ corresponds to the following angle matrix.

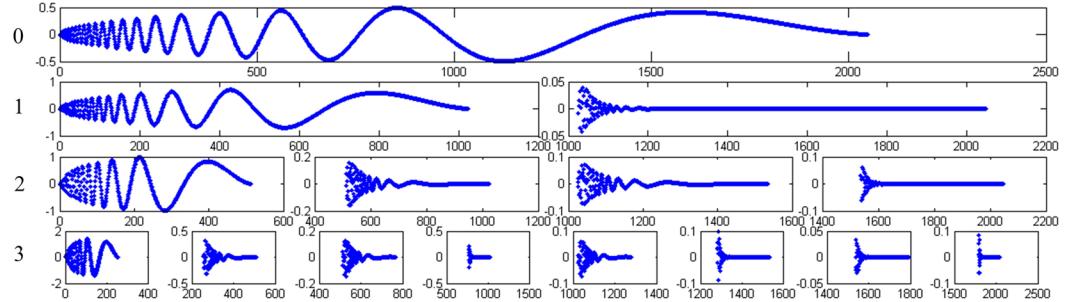
$$\Lambda_{2^n,2^m,2^p} = (\Lambda_{2^n,2^m}^1, \Lambda_{2^n,2^m}^2, \dots, \Lambda_{2^n,2^m}^{2^p}), \quad (63)$$

where the angle matrix $\Lambda_{2^n,2^m}^k$ is the k -th frame.

We firstly define the following DWPTs: $W^x(\cdot)$, $W^y(\cdot)$ and $W^t(\cdot)$.



(a) HQWPT



(b) DQWPT

Figure 26. The simulation results of the first 3 levels of HQWPT and DQWPT. The left number i refers to i -level QWT with $1 \leq i \leq 10$, and $i=0$ refers to the input signal.

level	$\text{norm}(M_1^k - M_3^k) \times 10^{13}$	$\text{norm}(S - M_2^k) \times 10^{13}$	$\text{norm}(M_3^k - M_6^k) \times 10^{10}$	$\text{norm}(S - M_4^k) \times 10^{13}$
1	0	0.0211	0.0057	0.0257
2	0.0228	0.0583	0.0128	0.0511
3	0.0252	0.0643	0.0230	0.0798
4	0.0361	0.0619	0.0374	0.1006
5	0.0311	0.0716	0.0574	0.1371
6	0.0693	0.0736	0.0842	0.1662
7	0.0505	0.1118	0.1168	0.2402
8	0.0960	0.1518	0.1564	0.2575
9	0.1325	0.1874	0.1963	0.3040
10	0.1982	0.2899	0.1560	0.3623
11	0.2497	0.3981	—	—

Table 1. The simulation results of the HQWT, IHQWT and HWT.

$$\begin{cases} W^x(A_{2^n, 2^m, 2^p}) = (W_{2^n} \Lambda_{2^n, 2^m}^1, \dots, W_{2^n} \Lambda_{2^n, 2^m}^{2^p}), \\ W^y(A_{2^n, 2^m, 2^p}) = (\Lambda_{2^n, 2^m}^1 W_{2^m}^T, \dots, \Lambda_{2^n, 2^m}^{2^p} W_{2^m}^T), \\ W^t(A_{2^n, 2^m, 2^p}) = (C_{2^n, 2^m}^1, C_{2^n, 2^m}^2, \dots, C_{2^n, 2^m}^{2^p}), \end{cases} \quad (64)$$

with the row vectors

$$\begin{cases} [C_{x,y}^1 \ C_{x,y}^2 \ \dots \ C_{x,y}^{2^p}] = u_{x,y} \times W_{2^m}^T, \\ u_{x,y} = [\theta_{x,y}^1 \ \theta_{x,y}^2 \ \dots \ \theta_{x,y}^{2^p}], \end{cases} \quad (65)$$

where $C_{x,y}^j$ and $\theta_{x,y}^j$ are the elements of the matrices $C_{2^n, 2^m}^j$ and $\Lambda_{2^n, 2^m}^j$ on the position (x, y) , respectively. Next, the 3D DFPT of $A_{2^n, 2^m, 2^p}$ can be defined as

$$wt3(A_{2^n, 2^m, 2^p}) = W^t(W^y(W^x(A_{2^n, 2^m, 2^p}))). \quad (66)$$

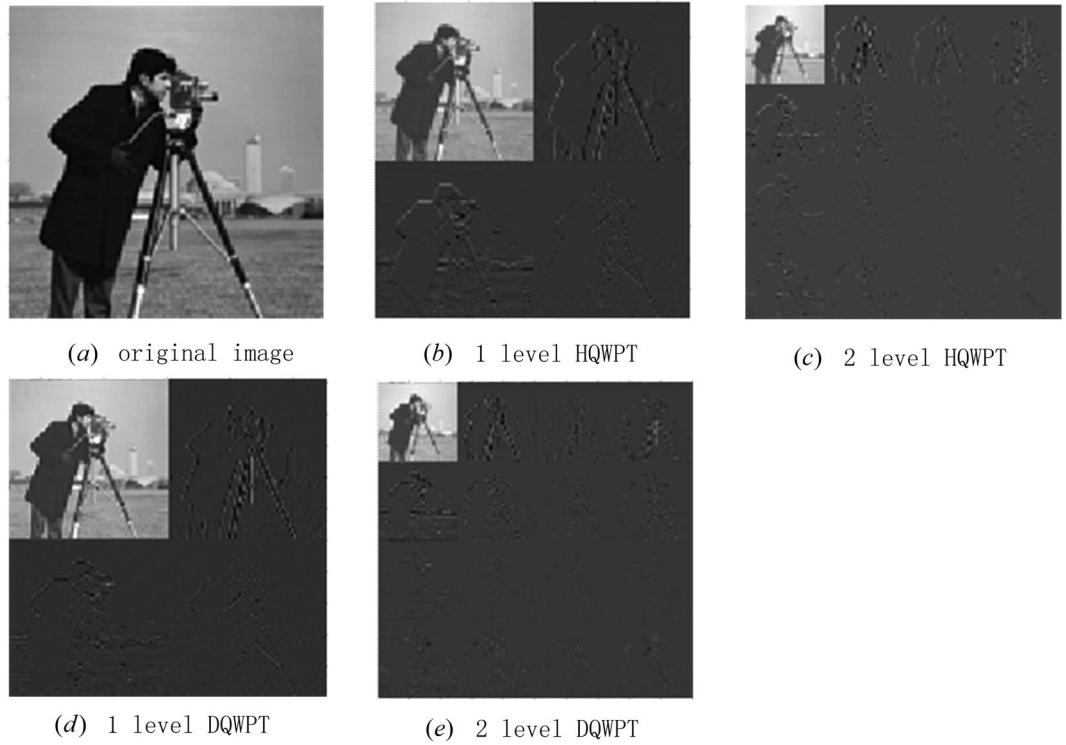


Figure 27. The simulation results of the first 2 levels of HQWPT and DQWPT.

Similarly, we utilize the equivalent function of the quantum circuit to create the NASS state of $A_{2^n, 2^m, 2^p}$

$$|\psi_3\rangle = f(A_{2^n, 2^m, 2^p}) = [u_{0,0} \cdots u_{0,2^m-1} u_{1,0} \cdots u_{1,2^m-1} \cdots u_{2^n-1,0} \cdots u_{2^n-1,2^m-1}]^T, \quad (67)$$

where the row vector $u_{x,y}$ is shown in equation (65).

Applying the function $f(\cdot)$ on $F^t(A_{2^n, 2^m, 2^p})$, $F^y(A_{2^n, 2^m, 2^p})$ and $F^x(A_{2^n, 2^m, 2^p})$ respectively, we have the following three equations.

$$\begin{cases} f(W^t(A_{2^n, 2^m, 2^p})) = (I_{2^{n+m}} \otimes W_{2^p})|\psi_3\rangle, \\ f(W^y(A_{2^n, 2^m, 2^p})) = (I_{2^n} \otimes W_{2^m} \otimes I_{2^p})|\psi_3\rangle, \\ f(W^x(A_{2^n, 2^m, 2^p})) = (W_{2^n} \otimes I_{2^m} \otimes I_{2^p})|\psi_3\rangle. \end{cases} \quad (68)$$

Therefore, we derive the 3D QWPT of $A_{2^n, 2^m, 2^p}$

$$f(wt3(A_{2^n, 2^m, 2^p})) = (W_{2^n} \otimes W_{2^m} \otimes W_{2^p})|\psi_3\rangle. \quad (69)$$

Substituting our proposed 1D QWPT into equations (62) and (69), we obtain 2D HQWPT, 2D DQWPT, 3D HQWPT and 3D DQWPT. Furthermore, their circuits can be designed in Figs 24 and 25.

Simulation Experiments

In the absence of a quantum computer to implement our proposed QWPTs, experiments of quantum signals are simulated on a classical computer. The quantum signals are stored in quantum states (i.e., column vectors) and the QWPTs are implemented using unitary matrices in Matlab (the R2010bversion).

Simulation experiments of the 1D HQWPT and DQWPT. Consider a quantum state

$$|\nu\rangle = \frac{1}{\sqrt{\sum_{i=0}^{2047} (\nu_i)^2}} [\nu_0 \ \nu_1 \ \cdots \ \nu_{2047}]^T \quad (70)$$

as an input signal of the QWT, where $\nu_k = d(k/2048)$, $k = 0, \dots, 2047$, and $d(t) = \sqrt{t(1-t)} \sin\left(\frac{2\pi * 1.05}{t + 0.05}\right)$.

For simply, we can take a vector

$$S = [\nu_0 \ \nu_1 \ \cdots \ \nu_{2047}]^T \quad (71)$$

as the input signal of simulation experiments, which is according with the state $|\nu\rangle$ without the normalized item.

level	$\text{norm} (Q_1^k - Q_5^k) \times 10^9$	$\text{norm} (\psi_2\rangle - Q_2^k) \times 10^{13}$	$\text{norm} (Q_3^k - Q_6^k) \times 10^8$	$\text{norm} (\psi_2\rangle - Q_4^k) \times 10^{14}$
1	0.0011	0.0041	0.0669	0.0427
2	0.0022	0.0093	0.0128	0.0880
3	0.0026	0.0104	0.2641	0.1837
4	0.0099	0.0125	0.3917	0.2904
5	0.0218	0.0242	0.5851	0.6436
6	0.0679	0.0518	0.9105	0.8180
7	0.1218	0.1079	—	—

Table 2. The simulation results of the 2D HQWT, IHQWT and HWT.

For convenience, let the single-level HQWPT and DQWPT be

$$R_{2^{11}}^0 = P_{2^{n-1}, 2}(I_{2^{n-1}} \otimes H), A_{2^{11}}^0 = T_{2^{11}}. \quad (72)$$

Applying multi-level HQWPT $R_{2^{11}}^k$, $k = 0, 1, \dots, 10$ and multi-level DQWPT $A_{2^{11}}^k$, $k = 0, 1, \dots, 9$ to the input signal S in Eq. (71), the simulation results of the first 3 levels are shown in Fig. 26 with multi-windows. Table 1 shows the comparison of simulation experiments of our proposed QWPT and the function of the WPT in Matlab using the 2-norm function $\text{norm}()$. The symbols in this table are listed as follows:

$$\begin{cases} M_1^k = R_{2^{11}}^k \times S, 0 \leq k \leq 10, \\ M_2^k = (R_{2^{11}}^k)^{-1} \times M_1^k, 0 \leq k \leq 10, \\ M_3^k = A_{2^{11}}^k \times S, 0 \leq k \leq 9, \\ M_4^k = (A_{2^{11}}^k)^{-1} \times M_3^k, 0 \leq k \leq 9. \end{cases} \quad (73)$$

The function $wpdec(S, k + 1, 'db1')$ in Matlab performs a $(k + 1)$ -level HWPT to return a wavelet packet tree. Next, we get the coefficients of the nodes of the wavelet packet tree using the function $wpcoef()$ to construct a vector M_5^k . Similarly, we obtain a vector M_6^k of a $(k + 1)$ -level DWPT based on the periodization extension by the $wpdec(S, k + 1, 'db2')$ and $wpcoef()$.

Simulation experiments of the 2D HQWPT and DQWPT. An angle matrix Λ_g is given by

$$\Lambda_g = \frac{\pi C_g}{2 \times 2^8 - 2} A_g, \quad (74)$$

where A_g is the 128×128 matrix of the gray-scale image shown in Fig. 27(a), and C_g is a constant corresponding to the image.

The NASS state $|\psi_2\rangle = f(\Lambda_g)$ can be regarded as a column vector, where the function $f(\cdot)$ is defined in equation (57). Applying the $k + 1$ level 2D HQWPT and DQWPT on the image A_g , respectively, the results are

$$\begin{cases} Q_1^k = \frac{2 \times 2^8 - 2}{\pi C_g} f^{-1}((R_{2^7}^k \otimes R_{2^7}^k)|\psi_2\rangle), \\ Q_3^k = \frac{2 \times 2^8 - 2}{\pi C_g} f^{-1}((A_{2^7}^k \otimes A_{2^7}^k)|\psi_2\rangle), \end{cases} \quad (75)$$

where $f^{-1}(\cdot)$ is the inverse function of $f(\cdot)$, which converts a column vector into a 2-dimension matrix.

The simulation results are shown in Fig. 27 and Table 2. The rest symbols in Table 2 are: $Q_2^k = |\psi_2\rangle - ((R_{2^7}^k)^{-1} \otimes (R_{2^7}^k)^{-1})[(R_{2^7}^k \otimes R_{2^7}^k)|\psi_2\rangle]$, $Q_4^k = |\psi_2\rangle - ((A_{2^7}^k)^{-1} \otimes (A_{2^7}^k)^{-1})[(A_{2^7}^k \otimes A_{2^7}^k)|\psi_2\rangle]$. Similarly with the 1D HQWPT and DQWPT, matrices Q_5^k and Q_6^k are created using the functions $wpdec2(A_g, k + 1, 'db1')$, $wpdec2(A_g, k + 1, 'db2')$ and $wpcoef()$, respectively.

Simulation experiments of the 3D HQWPT and DQWPT. An angle matrix Λ_c is given by

$$\Lambda_c = \frac{\pi C_c}{2 \times 2^8 - 2} V_t, \quad (76)$$

where V_t is the $64 \times 64 \times 4$ matrix of the video shown in (a) of Fig. 28, and C_c is a constant corresponding to the video.

The NASS state $|\psi_3\rangle = f(\Lambda_c)$ can be regarded as a column vector, where the function $f(\cdot)$ is defined in equation (67). Applying the $k + 1$ level 3D HQWPT and DQWPT on the video V_t , respectively, the results are

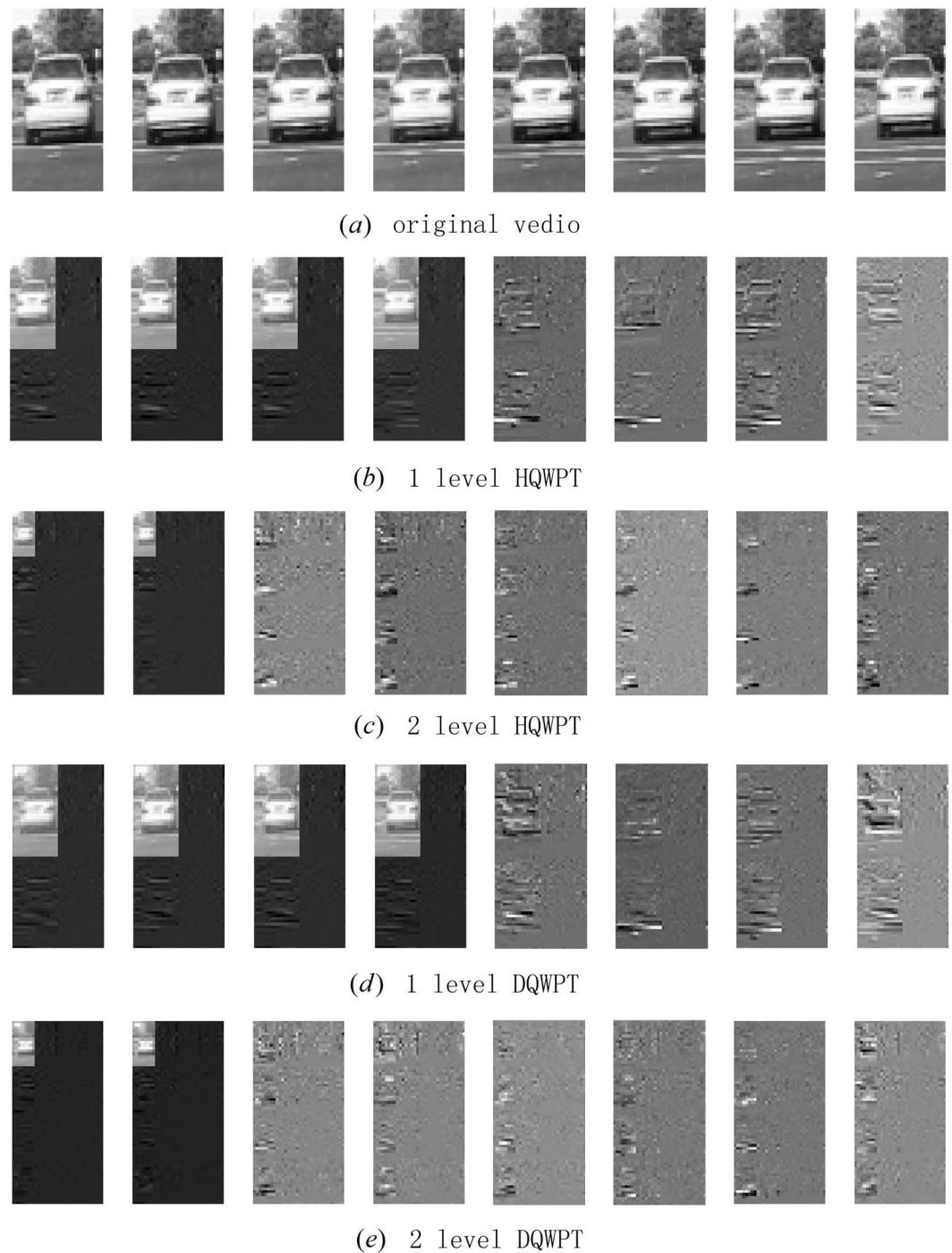


Figure 28. The simulation results of the first 2 levels of the 3D HQWPT and DQWPT.

$$\begin{cases} V_1^k = \frac{2 \times 2^8 - 2}{\pi C_g} f^{-1}((R_{2^6}^k \otimes R_{2^6}^k \otimes R_{2^6}^k) |\psi_3\rangle), \\ V_3^k = \frac{2 \times 2^8 - 2}{\pi C_g} f^{-1}((A_{2^6}^k \otimes A_{2^6}^k \otimes A_{2^6}^k) |\psi_3\rangle), \end{cases} \quad (77)$$

where $f^{-1}(\cdot)$ converts a column vector into a 3-dimension matrix.

The simulation results are shown in Fig. 28 and Table 3. Since there are no functions of the 3D WPT, we realize $wt3$ in (66) using the functions `wpdec2()` and `wpdec()` and note V_5^k and V_6^k as results of 3D HWPT and DWPT,

level	$\text{norm } (V_1^k - V_5^k) \times 10^{10}$	$\text{norm } (\psi_3\rangle - V_2^k) \times 10^{14}$	$\text{norm } (V_3^k - V_6^k) \times 10^8$	$\text{norm } (\psi_3\rangle - V_4^k) \times 10^{14}$
1	0.0164	0.0488	0.1787	0.0714
2	0.0702	0.1449	0.3328	0.2005
3	0.2171	0.2044	—	—

Table 3. The simulation results of the 3D HQWT, IHQWT and HWT.

respectively. The rest symbols in Table 2 are $V_2^k = |\psi_3\rangle - ((R_{2^6}^k)^{-1} \otimes (R_{2^5}^k)^{-1} \otimes (R_{2^3}^k)^{-1}) [(R_{2^6}^k \otimes R_{2^5}^k \otimes R_{2^3}^k) |\psi_3\rangle]$ and $V_4^k = |\psi_3\rangle - ((A_{2^6}^k)^{-1} \otimes (A_{2^5}^k)^{-1} \otimes (A_{2^3}^k)^{-1}) [(A_{2^6}^k \otimes A_{2^5}^k \otimes A_{2^3}^k) |\psi_3\rangle]$.

Analyzing the above simulation experiments, we conclude that our proposed HQWPT, IHQWPT, DQWPT and IDQWPT can implement decompositions and reconstructions of the Haar wavelet and D4 wavelet, respectively. The simulation results of our proposed HQWPT and DQWPT, which are equal to the corresponding WPTs without consideration of truncation error on machine computing, show our proposed QWPTs are correct.

Conclusion and Future Works

This article has constructed the iteration equations of multi-level and multi-dimensional QWPTs by GTP and PSP. The iteration equations include HQWPT, DQWPT based on the periodization extension and their inverse transforms for the first time, which ensure the theoretical correctness of our proposed QWPTs. Next, we have designed circuits of the proposed QWPTs. The precise analysis of the quantum costs and the time complexities of circuits prove that our proposed QWPTs are of high-efficiency. For instance, the time complexities of the multi-level HQWPT and DQWPT at most are 6 and $(5n^3 + O(n^2))$ on 2^n , respectively. In contrast, the classical fast WPTs need $O(n2^n)$ basic operations to implement the discrete wavelet transform^{21,32}. Thus, our proposed QWPT can exponentially speed up the computation of the wavelet transform in comparison to the one on a classical computer. The simulation results show that our proposed QWPTs are correct and effective. In summary, the proposed QWPTs and IQWPTs can implement effective decompositions and reconstructions of 1D signals, 2D images and 3D video, respectively. Therefore, the article provide a feasible scheme for the WPT to be applied in QIP.

Studies of quantum wavelet packet are still in their infancy. Multi-level and multi-dimension wavelet transforms play an important role in classical image and signal processing, therefore, their quantum versions will be significant and core tool algorithms for quantum image and signal processing. Our future works are how to use these wavelet transforms to implement some complex operations, such as quantum image and signal compression, and quantum image and signal denoising.

References

1. Stajic, J. The future of quantum information processing. *Sci.* **339**, 1163–1163 (2013).
2. Li, H.-S. *et al.* Image storage, retrieval, compression and segmentation in a quantum system. *Quantum Inf. Process.* **12**, 2269–2290 (2013).
3. Li, H.-S., Zhu, Q., Zhou, R.-G., Song, L. & Yang, X.-J. Multi-dimensional color image storage and retrieval for a normal arbitrary quantum superposition state. *Quantum inf. process* **13**, 991–1011 (2014).
4. Zhang, Y., Lu, K., Gao, Y. & Wang, M. Nqr: a novel enhanced quantum representation of digital images. *Quantum inf. process* **12**, 2283–2860 (2013).
5. Le, P. Q., Dong, F. & Hirota, K. A flexible representation of quantum images for polynomial preparation, image compression. *Quantum inf. process* **10**, 63–84 (2011).
6. Li, H.-S., Zhu, Q., Zhou, R.-G., Li, M.-C. & Ian, H. Multidimensional color image storage, retrieval, and compression based on quantum amplitudes and phases. *Inf. Sci.* **273**, 212–232 (2014).
7. Yan, F., Ilyasu, A. M. & Venegas-Andraca, S. E. A survey of quantum image representations. *Quantum Inf. Process.* **15**, 1–35 (2016).
8. Li, H.-S., Fan, P., Xia, H.-Y., Peng, H. & Song, S. Quantum implementation circuits of quantum signal representation and type conversion. *IEEE Transactions on Circuits Syst. I: Regul. Pap.* 1–14 (2018).
9. Shor, P. W. Foundations of computer science. *1994 Proceedings, 35th Annu. Symp. on IEEE* 124–134 (1994).
10. Deutsch, D. Quantum theory, the church-turing principle and the universal quantum computer. *Proc R Soc Lond. A* **400**, 97–117 (1985).
11. Grover, L. A fast quantum mechanical algorithm for database search. *Proc. 28th Annu. ACM Symp. on Theory Comput.* 212–219 (1996).
12. Fan, P., Zhou, R.-G., Jing, N. & Li, H.-S. Geometric transformations of multidimensional color images based on nass. *Inf. Sci.* **340**, 191–208 (2016).
13. Zhou, R.-G., Hu, W., Fan, P. & Ian, H. Quantum realization of the bilinear interpolation method for neqr. *Sci. Reports* **7**, 2511 (2017).
14. Zhou, R., Hu, W., Luo, G., Liu, X. & Fan, P. Quantum realization of the nearest neighbor value interpolation method for ineqr. *Quantum Inf. Process.* **17**, 166 (2018).
15. Zhou, R.-G., Wu, Q., Zhang, M.-Q. & Shen, C.-Y. Quantum image encryption and decryption algorithms based on quantum image geometric transformations. *Int. J. Theor. Phys.* **52**, 1802–1817 (2013).
16. Zhou, R.-G. *et al.* A novel quantum image steganography scheme based on lsb. *Int. J. Theor. Phys.* **57**, 1848–1863 (2018).
17. Yan, F. *et al.* A duple watermarking strategy for multi-channel quantum images. *Quantum Inf. Process.* **14**, 1675–1692 (2015).
18. Yao, X.-W. *et al.* Quantum image processing and its application to edge detection: Theory and experiment. *Phys. Rev. X* **7**, 031041 (2017).
19. Kai, L., Zhang, Y., Wang, X.-P. & Kai, L. A strategy of quantum image filtering in frequency domain. *DEStech Transactions on Eng. Technol. Res* (2016).
20. Ramchandran, K., Xiong, Z., Asai, K. & Vetterli, M. Adaptive transforms for image coding using spatially varying wavelet packets. *IEEE Transactions on Image Process.* **5**, 1197–1204 (1996).
21. Ouyang, W., Zhao, T., Cham, W.-K. & Wei, L. Fast full-search equivalent pattern matching using asymmetric haar wavelet packets. *IEEE Transactions on Circuits Syst. for Video Technol.* **28**, 819–833 (2016).
22. Yu, X. Wavelet packet transform for fractional brownian motion: Asymptotic decorrelation and selection of best bases. *IEEE Transactions on Inf. Theory* **63**, 4532–4550 (2017).

23. Klappenecker, A. Wavelets and wavelet packets on quantum computers. arXiv preprint quant-ph/9909014 (1999).
24. Hoyer, P. Efficient quantum transforms. arXiv preprint quant-ph/9702028 (1997).
25. Fijany, A. & Williams, C. P. Quantum wavelet transforms: Fast algorithms and complete circuits. *Lect. Notes Comput. Sci.* **1509**, 10–33 (1998).
26. Terraneo, M. & Shepelyansky, D. L. Imperfection effects for multiple applications of the quantum wavelet transform. arXiv preprint quant-ph/0303043 (2003).
27. Barenco, A. *et al.* Elementary gates for quantum computation. *Phys. Rev. A* **52**, 3457–3467 (1995).
28. Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information*. (Cambridge University Press, 2000).
29. Fino, B. J. & Algazi, V. R. A unified treatment of discrete fast unitary transforms. *SIAM J. on Comput.* **6**, 700–717 (1977).
30. Smolin, J. A. & DiVincenzo, D. P. Five two-bit quantum gates are sufficient to implement the quantum fredkin gate. *Phys. Rev. A* **53**, 2855 (1996).
31. Ruch, D. K. & Van Fleet, P. J. *Wavelet theory: an elementary approach with applications*. (John Wiley & Sons, 2011).
32. Noskoski, O. A., Bermudez, J. C. & de Almeida, S. J. Region-based wavelet-packet adaptive algorithm for identification of sparse impulse responses. *IEEE Transactions on Signal Process.* **61**, 3321–3333 (2013).

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant No. 61462026, No. 61762012, No. 61763014 and No. 61762014, Project of Science and Technology of Jiangxi province Grant No. 2016BAB202065, the key research project of Guangxi Normal University Grant No. 2016ZD008 and an award of China Scholarship Council, Science and technology research project of Jiangxi Provincial Education Department under Grant No. GJJ170382, Project of International Cooperation and Exchanges of Jiangxi Province under Grant No. 2016BBH80034, Project of Humanities and Social Sciences in colleges and universities of Jiangxi Province under Grant No. JC161023, the Fund for Distinguished Young Scholars of Jiangxi Province under Grant No.2018ACB2101.

Author Contributions

Li H.S. and Fan P. conceived the core theme, Xia H.Y. conducted experiments, Song S. drew many figures, He X. reviewed and corrected the manuscript.

Additional Information

Competing Interests: The authors declare no competing interests.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2018