Efficient Quantum Walk on a Quantum Processor

Xiaogang Qiang,[1, *] Thomas Loke,[2, *] Ashley Montanaro,[3] Kanin Aungskunsiri,[1] Xiaoqi
Zhou,[1, 4] Jeremy L. O'Brien,[1] Jingbo Wang,[2, †] and Jonathan C. F. Matthews[1, ‡]

[1]*Centre for Quantum Photonics, H. H. Wills Physics Laboratory and Department of
Electrical and Electronic Engineering, University of Bristol, Bristol BS8 1UB, UK.*
[2]*School of Physics, The University of Western Australia, WA6009, Australia.*
[3]*School of Mathematics, University of Bristol, Bristol BS8 1TW, UK.*
[4]*State Key Laboratory of Optoelectronic Materials and Technologies and School of Physics and Engineering,
Sun Yat-sen University, Guangzhou 510275, China.*
(Dated: October 30, 2015)

The random walk formalism is used across a wide range of applications, from modelling share prices to predicting population genetics. Likewise quantum walks have shown much potential as a framework for developing new quantum algorithms. In this paper, we present explicit efficient quantum circuits for implementing continuous-time quantum walks on the circulant class of graphs. These circuits allow us to sample from the output probability distributions of quantum walks on circulant graphs efficiently. We also show that solving the same sampling problem for arbitrary circulant quantum circuits is intractable for a classical computer, assuming conjectures from computational complexity theory. This is a new link between continuous-time quantum walks and computational complexity theory and it indicates a family of tasks which could ultimately demonstrate quantum supremacy over classical computers. As a proof of principle we have experimentally implemented the proposed quantum circuit on an example circulant graph using a two-qubit photonics quantum processor.

Quantum walks are the quantum mechanical analogue to the well-known classical random walk and they have established roles in quantum information processing [1–3]. In particular, they are central to quantum algorithms created to tackle database search [4], graph isomorphism [5–7], network analysis and navigation [8, 9], and quantum simulation [10–12], as well as modelling biological processes [13, 14]. Meanwhile, physical properties of quantum walks have been demonstrated in a variety of systems, such as nuclear magnetic resonance [15, 16], bulk [17] and fiber [18] optics, trapped ions [19–21], trapped neutral atoms [22], and photonics [23, 24]. Almost all physical implementations of quantum walk so far followed an analog approach as for quantum simulation [25, 26], whereby the apparatus is dedicated to implement specific instances of Hamiltonians without translation onto quantum logic. However, there is no existing method to implement analog quantum simulations with error correction or fault tolerance, and they do not scale efficiently in resources when simulating broad classes of large graphs.

In this paper, we present efficient quantum circuits for implementing continuous time quantum walks (CTQWs) on circulant graphs with an eigenvalue spectrum that can be classically computed efficiently. These quantum circuits provide the time-evolution states of CTQWs on circulant graphs exponentially faster than best previously known methods [27]. We report a proof-of-principle experiment, where we implement CTQWs on an example circulant graph (namely the complete graph of four vertices) using a two-qubit photonics quantum processor to sample the probability distributions and perform state tomography on the output state of a CTQW. We also

provide evidence from computational complexity theory that the probability distributions that are output from the circuits of this circulant form are hard to sample from using a classical computer, implying our scheme also provides an exponential speedup for sampling.

Efficient quantum circuit implementations of CTQWs have been presented for sparse and efficiently row-computable graphs [28, 29], and specific non-sparse graphs [30, 31]. However, the design of quantum circuits for implementing CTQWs is in general difficult, since the time-evolution operator is time-dependent and non-local [1]. A subset of circulant graphs have the property that their eigenvalues and eigenvectors can be classically computed efficiently [27, 32]. This enables us to construct a scheme that efficiently outputs the quantum state $|\psi(t)\rangle$, which corresponds to the time evolution state of a CTQW on corresponding graphs. One can then either perform direct measurements on $|\psi(t)\rangle$ or implement further quantum circuit operations to extract physically meaningful information. For example the "SWAP test" [33] can be used to estimate the similarity of dynamical behaviors of two circulant Hamiltonians operating on two different initial states, as shown in Figure 1(A). This procedure can also be adapted to study the stability of quantum dynamics of circulant molecules (for example, the DNA Möbius strips [34]) in a perturbational environment [35, 36].

On the other hand, when measuring $|\psi(t)\rangle$ in the computational basis we can sample the probability distribution

$$p(x) := |\langle x \mid \psi(t)\rangle|^2 \tag{1}$$

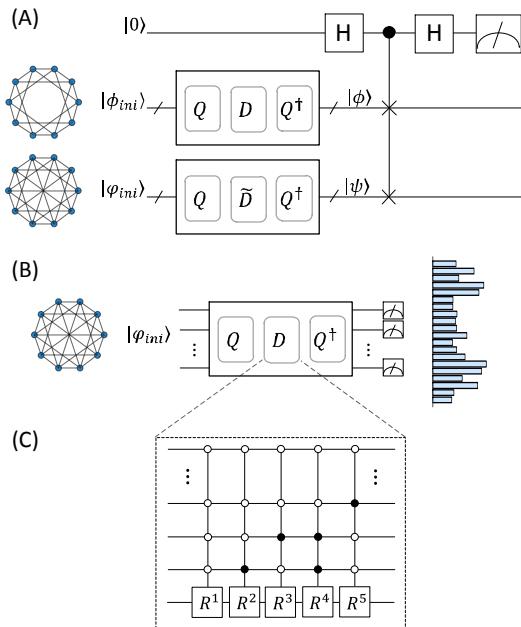that describes the probability of observing the quantum

FIG. 1: **Applications for generating the time evolution state of circulant Hamiltonians.** (A) The SWAP test [33] can be used to estimate the similarity of two evolution states of two similar circulant systems, or when one of the Hamiltonians is non-circulant but efficiently implementable. In brief, an ancillary qubit is entangled with the output states ψ and ϕ of two compared processes according to $\frac{1}{2}|0\rangle\left[|\phi\rangle|\psi\rangle+|\psi\rangle|\phi\rangle\right]+\frac{1}{2}|1\rangle\left[|\phi\rangle|\psi\rangle-|\psi\rangle|\phi\rangle\right]$. On measuring the ancillary qubit we obtain outcome "1" with probability $\frac{1}{2}(1-|\langle\phi|\psi\rangle|^2)$—the probability of observing "1" indicates the similarity of dynamical behaviors of the two processes. (B) Probability distributions are sampled by measuring the evolution state in a complete basis, such as the computational basis. (C) An example of the quantum circuit for implementing diagonal unitary operator $D=\exp(-it\Lambda)$, where the circulant Hamiltonian has 5 non-zero eigenvalues. The open and solid circles represent the control qubits as "if $|0\rangle$" and "if $|1\rangle$" respectively. $R^i=[1,0;0,\exp(-it\lambda_i)]$ $(i=1,\cdots,5)$, where λ_i is the corresponding eigenvalue.

walker at position $x\in\{0,1\}^n$—an n-bit string, corresponding to the vertices of the given graph, as shown in Figure 1(B). Sampling of this form is sufficient to solve various search and characterization problems [4, 9], and can be used to deduce critical parameters of the quantum walk, such as mixing time [2]. It is unlikely for a classical computer to be able to efficiently sample from $p(x)$. We adapt the similar methodology of refs. [37–39] to show that if there did exist a classical sampler for a somewhat more general class of circuits, this would have the following unlikely complexity-theoretic implication: the infinite tower of complexity classes known as the polynomial hierarchy would collapse. This evidence of hardness exists despite the classical efficiency with which properties of the CTQW, such as the eigenvalues of circulant graphs, can be computed on a classical machine.

For an undirected graph G of N vertices, a quantum particle (or "quantum walker") placed on G evolves into a superposition $|\psi(t)\rangle$ of states in the orthonormal basis

$\{|1\rangle,|2\rangle,\ldots,|N\rangle\}$ that correspond to vertices of G. The exact evolution of the CTQW is governed by connections between the vertices of G: $|\psi(t)\rangle=\exp(-itH)|\psi(0)\rangle$ where the Hamiltonian is given by $H=\gamma A$ for hopping rate per edge per unit time γ and where A is the N-by-N symmetric adjacency matrix, whose entries are $A_{jk}=1$, if vertices j and k are connected by an edge in G, and $A_{jk}=0$ otherwise [1].

Circulant graphs are defined by symmetric circulant adjacency matrices for which each row j when right-rotated by one element, equals the next row $j+1$—for example complete graphs, cycle graphs and Mobius ladder graphs are all subclasses of circulant graphs. It follows that Hamiltonians for CTQWs on any circulant graph have a symmetric circulant matrix representation, which can be diagonalized by the unitary Fourier transform [32], i.e. $H=Q^\dagger\Lambda Q$, where

$$Q_{jk}=\frac{1}{\sqrt{N}}\omega^{jk},\ \omega=\exp(2\pi i/N) \tag{2}$$

and Λ is a diagonal matrix containing eigenvalues of H, which are all real and whose order is determined by the order of the eigenvectors in Q. Consequently, we have $\exp(-itH)=Q^\dagger\exp(-it\Lambda)Q$, where the time dependence of $\exp(-itH)$ is confined to the diagonal unitary operator $D=\exp(-it\Lambda)$.

The Fourier transformation Q can be implemented efficiently by the well-known QFT quantum circuit [40]. For a circulant graph that has $N=2^n$ vertices, the required QFT of N dimension can be implemented with $O((\log N)^2)=O(n^2)$ quantum gates acting on $O(n)$ qubits. To implement the inverse QFT, the same circuit is used in reverse order with phase gates of opposite sign. D can be implemented using at most N controlled-phase gates with phase values being a linear function of t, because an arbitrary phase can be applied to an arbitrary basis state, conditional on at most $n-1$ qubits. Given a circulant graph that has $O(\text{poly}(n))$ non-zero eigenvalues, only $O(\text{poly}(n))$ controlled-phase gates are needed to implement D. If the given circulant graph has $O(2^n)$ distinct eigenvalues, which can be characterised efficiently (such as the cycle graphs and Mobius ladder graphs), we are still able to implement the diagonal unitary operator D using polynomial quantum resources. A general construction of efficient quantum circuits for D was given by Childs [6], and is shown in the Appendix for completeness. Thus, the quantum circuit implementations of CTQWs on circulant graphs can be constructed, which have an overall complexity of $O(\text{poly}(n))$, and act on at most $O(n)$ qubits. Compared with the best known classical algorithm based on fast Fourier transform, that has the computational complexity of $O(n2^n)$ [27], the proposed quantum circuit implementation generates the evolution state $|\psi(t)\rangle$ with an exponential advantage in speed.

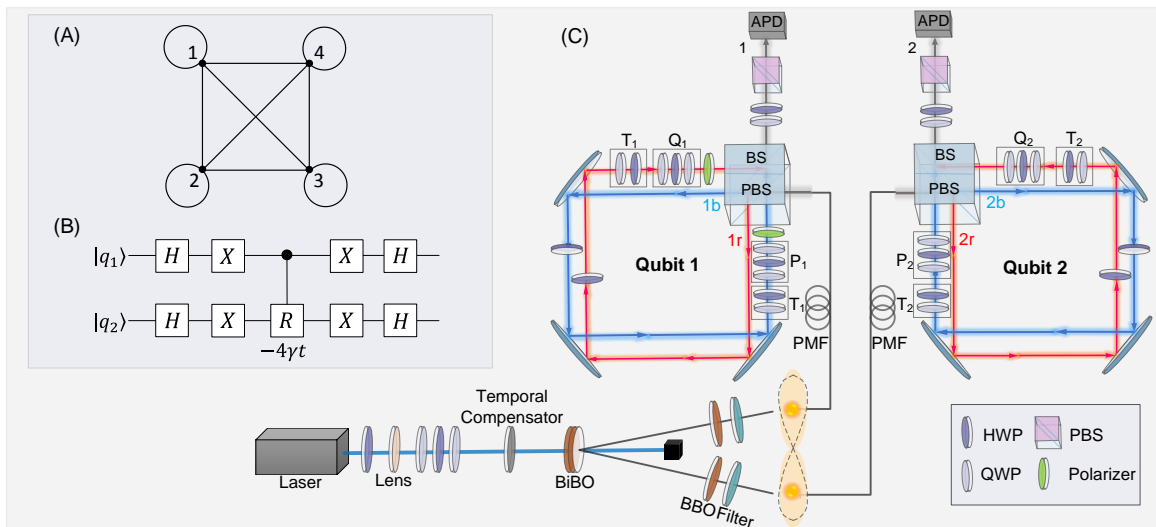Consider a circuit of the form $Q^\dagger DQ$, where D is a di-

FIG. 2: **The schematic diagram and setup of experimental demonstration.** (A) The K_4 graph. (B) The quantum circuit for implementing CTQW on the K_4 graph. This can also be used to implement CTQW on the K_4 graph without self-loops, up to a global phase factor $\exp(i\gamma t)$. H and X represent the Hadamard and Pauli-X gate respectively. $R = [1, 0; 0, \exp(-i4\gamma t)]$ is a phase gate. (C) The experimental setup for a reconfigurable two-qubit photonics quantum processor, consisting of a polarization-entangled photon source using paired type-I BiBO crystal in the sandwich configuration and displaced Sagnac interferometers. See further details in Methods.

agonal matrix made up of poly(n) controlled-phase gates. Define

$$p_D := |\langle 0|^{\otimes n} Q^\dagger D Q |0\rangle^{\otimes n}|^2$$
$$= |\langle +|^{\otimes n} D |+\rangle^{\otimes n}|^2$$
$$= |\langle 0|^{\otimes n} H^{\otimes n} D H^{\otimes n} |0\rangle^{\otimes n}|^2. \quad (3)$$

$H^{\otimes n} D H^{\otimes n}$ represents a family of circuits having the following structure: each qubit line begins and ends with a Hadamard (H) gate, and, in between, every gate is diagonal in the computational basis. This class of circuits is known as instantaneous quantum polynomial time (IQP) [38, 42]. It is known that computing p_D for arbitrary diagonal unitaries D made up of circuits of poly(n) gates, even if each acts on $O(1)$ qubits, is #P-hard [39, 43, 44]. This hardness result even holds for approximating p_D up to any multiplicative error strictly less than $1/2$ [43, 44], where $\widetilde{p_D}$ is said to approximate p_D up to multiplicative error ϵ if

$$|\widetilde{p_D} - p_D| \leq \epsilon\, p_D. \quad (4)$$

Towards a contradiction, assume that there exists a polynomial-time randomized classical algorithm which samples from p. Then a classic result of Stockmeyer [45] states that there is an algorithm in the complexity class FBPP^{NP} which can approximate any desired probability $p(x)$ to within multiplicative error $O(1/\text{poly}(n))$. This complexity class FBPP^{NP}—described as polynomial-time randomized classical computation equipped with an oracle to solve arbitrary NP problems—sits within the infinite tower of complexity classes known as the polynomial(-time) hierarchy [46]. Combining with the

above hardness result of approximating p_D, we find that the assumption implies that an FBPP^{NP} algorithm solves a #P-hard problem, so $\text{P}^{\#\text{P}}$ would be contained within FBPP^{NP}, and therefore the polynomial hierarchy would collapse to its third level. This consequence is considered very unlikely in computational complexity theory [46].

We therefore conclude that a polynomial-time randomized classical sampler from the distribution p is unlikely to exist. Further, this even holds for classical algorithms which sample from any distribution $\widetilde{p}$ which approximates p up to multiplicative error strictly less than $1/2$ in each probability $p(x)$. It is worth noting that if the output distribution results from measurements on only $O(\text{poly}\log n)$ qubits [47], or obeys the sparsity promise that only a poly(n)-sized, and a priori unknown, subset of the measurement probabilities are nonzero [48], it could be classically efficiently sampled. The proof of hardness here does not hold for the approximate sampling from p up to small *additive* error. It is an interesting open question whether similar techniques to [37, 39] can be used to prove hardness of the approximate case, perhaps conditioned on other conjectures in complexity theory.

As an experimental demonstration, we used a photonic quantum logic to simulate CTQWs on the K_4 graph—a complete graph with self loops on four vertices (Figure 2(A)). Complete graphs are a special kind of circulant graph, with an adjacency matrix A where $A_{jk} = 1$ for all j, k. The Hamiltonian of a complete graph on N vertices has only 2 distinct eigenvalues, 0 and $N\gamma$. Therefore, the diagonal matrix of eigenvalues of K_4 is $\Lambda = \text{diag}(\{4\gamma, 0, 0, 0\})$. Following the aforementioned discussion, we can readily construct the quantum circuit
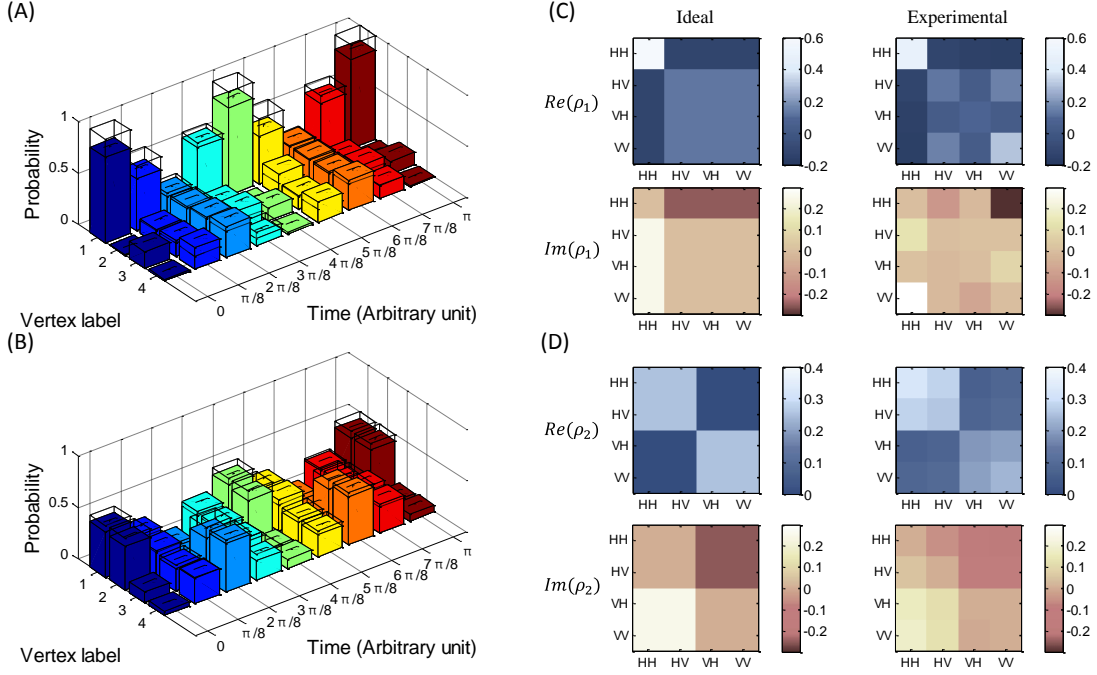
FIG. 3: **Experimental results for simulating CTQWs on K_4.** (A)-(B) The experimental sampled probability distributions with ideal theoretical distributions overlaid, for CTQWs on K_4 graph with initial states $|\varphi_{ini}\rangle_1 = [1, 0, 0, 0]'$ and $|\varphi_{ini}\rangle_2 = \frac{1}{\sqrt{2}}[1, 1, 0, 0]'$. The standard deviation of each individual probability is also plotted, which is calculated by propagating error assuming Poissonian statistics. (C)-(D) The ideal theoretical and experimentally reconstructed density matrices for the states $|\varphi_{out}\rangle_1 = [0.75 + 0.25i, -0.25 + 0.25i, -0.25 + 0.25i, -0.25 + 0.25i]'$ (corresponding to ρ_1) and $|\varphi_{out}\rangle_2 = [0.3536 + 0.3536i, 0.3536 + 0.3536i, -0.3536 + 0.3536i, -0.3536 + 0.3536i]'$ (corresponding to ρ_2). Both of the real and imaginary parts of the density matrices are obtained through the maximum likelihood estimation technique, and is shown as $Re(\rho)$ and $Im(\rho)$ respectively. Further results are shown in Appendix.

for implementing CTQWs on K_4 graph based on diagonalization using the QFT matrix. However, the choice of using the QFT matrix as the eigenbasis of Hamiltonian is not strictly necessary – any equivalent eigenbasis can be selected. Through the diagonalization using Hadamard eigenbasis, an alternative efficient quantum circuit for implementing CTQWs on K_4 graph is shown in Figure 2(B), which can be easily extended to the complete graph on N vertices.

We built a configurable two-qubit photonics quantum processor (Figure 2(C)), adapting the entanglement-based technique presented in [49], and implemented CTQWs on K_4 graph with various evolving times and initial states. Specifically, we prepared two different initial states $|\varphi_{ini}\rangle_1 = [1, 0, 0, 0]'$ and $|\varphi_{ini}\rangle_2 = \frac{1}{\sqrt{2}}[1, 1, 0, 0]'$, which represent the quantum walker starting from vertex 1, and the superposition of vertices 1 and 2 respectively. We chose the evolution time following the list $\left\{0, \frac{1}{8}\pi, \frac{2}{8}\pi, \frac{3}{8}\pi, \frac{4}{8}\pi, \frac{5}{8}\pi, \frac{6}{8}\pi, \frac{7}{8}\pi, \pi\right\}$, which covers the whole periodical characteristics of CTQWs on K_4 graph. For each evolution, we sampled the corresponding probability distribution with fixed integration time, shown in Figure 3(A) and (B). To measure how close the experimental and ideal probability distributions are, we calculated the average fidelities defined as $F_{average} =$

$\frac{1}{9}\sum_{n=1}^{9}\sum_{i=1}^{4}\sqrt{P_{ideal,n}(i)P_{exp,n}(i)}$. The achieved average fidelities for the samplings with two distinct initial states are 96.68±0.27% and 95.82±0.25% respectively. Through the proposed circuit implementation, we are also able to examine the evolution states using quantum state tomography, which is generally difficult for the analog simulations. For two specific evolution states $|\varphi_{out}\rangle_1 = \exp(-iH\frac{7}{8}\pi)|\varphi_{ini}\rangle_1$ and $|\varphi_{out}\rangle_2 = \exp(-iH\frac{7}{8}\pi)|\varphi_{ini}\rangle_2$, we performed quantum state tomography and reconstructed the density matrices using the maximum likelihood estimation technique. The two reconstructed density matrices achieve fidelities of 85.81±1.08% and 88.44±0.97% respectively, shown in Figure 3(C) and (D).

In this paper, we have described how CTQWs on circulant graphs can be efficiently implemented on a quantum computer, if the eigenvalues of the graphs can be characterised efficiently classically. In fact, we can construct an efficient quantum circuit to implement CTQWs on any graph whose adjacency matrix is efficiently diagonalisable, in other words, as long as the matrix of column eigenvectors Q and the diagonal matrix of the eigenvalue exponentials D can be implemented efficiently. We have shown that the problem of sampling from the output probability distributions of quantum circuits of

the form $Q^{\dagger}DQ$ is hard for classical computers, based on a highly plausible conjecture that the polynomial hierarchy does not collapse. This observation is particularly interesting from both perspectives of CTQW and computational complexity theory, as it provides new insights into the CTQW framework and also helps to classify and identify new problems in computational complexity theory. For the CTQWs on the circulant graphs of poly(n) non-zero eigenvalues, the proposed quantum circuit implementations do not need a fully universal quantum computer, and thus can be viewed as an intermediate model of quantum computation. Although the hardness of the approximate case of the sampling problem is unknown, the evidence we provided for the exact case indicates a promising candidate for experimentally establishing quantum supremacy over classical computers, and further evidence against the extended Church-Turing thesis. Compared with other intermediate models such as the one clean qubit model [50], IQP and boson sampling [37], the quantum circuit implementation of CTQWs is also more appealing due to available methods in fault tolerance and error correction, which are difficult to implement for other models [51]. This may also lead onto other practical applications through the use of CTQWs for quantum algorithm design.

Experimental Setup A diagonally polarized, 120 mW, continuous-wave laser beam with central wavelength of 404 nm is focused at the centre of paired type-I BiBO crystals with their optical axes orthogonally aligned to each other, to create the polarization entangled photon-pairs [52]. Through the spontaneous parametric down-conversion process, the photon pairs are generated in the state of $\frac{1}{\sqrt{2}}\left(|H_1 H_2\rangle + |V_1 V_2\rangle\right)$, where H and V represent horizontal and vertical polarization respectively. The photons pass through the polarization beam-splitter (PBS) part of the dual PBS/beam-splitter (BS) cubes on both arms to generate two-photon four-mode state of the form $\frac{1}{\sqrt{2}}\left(|H_{1b} H_{2b}\rangle + |V_{1r} V_{2r}\rangle\right)$ (r and b label red and blue paths shown in Figure 2(C)). Rotations T_1 and T_2 on each path, consisting of half wave-plate (HWP) and quarter wave-plate (QWP), convert the state into $\frac{1}{\sqrt{2}}\left(|\phi_{1b}\phi_{2b}\rangle + |\phi_{1r}\phi_{2r}\rangle\right)$, where $|\phi_1\rangle$ and $|\phi_2\rangle$ can be arbitrary single-qubit states. The four spatial modes $1b$, $2b$, $1r$ and $2r$ pass through four single-qubit quantum gates P_1, P_2, Q_1 and Q_2 respectively, where each of the four gates is implemented through three wave plates: QWP, HWP and QWP. The spatial modes $1b$ and $1r$ ($2b$ and $2r$) are then mixed on the BS part of the cube. By post-selecting the case where the two photons exit at ports 1 and 2, we obtain the state $(P_1 \otimes P_2 + Q_1 \otimes Q_2)|\phi_1\phi_2\rangle$. In this way, we implement a two-qubit quantum operation of the form $P_1 \otimes P_2 + Q_1 \otimes Q_2$ on the initialized state $|\phi_1\phi_2\rangle$.

As shown in Figure 2(B), the quantum circuit for implementing CTQW on the K_4 graph consists of Hadamard gates (H), Pauli-X gates (X) and controlled-phase gate (CP). CP is implemented by configuring $P_1 = |H\rangle\langle H|$, $P_2 = I$, $Q_1 = |V\rangle\langle V|$, $Q_2 = R(= \left[1, 0; 0, e^{-i4\gamma t}\right])$, where P_1 and Q_1 are implemented by polarizers. Together with combining the operation $(H \cdot X) \otimes (H \cdot X)$ before CP with state preparation and the operation $(X \cdot H) \otimes (X \cdot H)$ after CP with measurement setting, we implement the whole quantum circuit on the experimental setup. The evolution time of CTQW is con-

trolled by the phase value of R, which is determined by setting the three wave plates of Q_2 in Figure 2(C) to QWP($\frac{\pi}{4}$), HWP(ω), QWP($\frac{\pi}{4}$), where the angle ω of HWP equals to the phase θ of R. The evolution time t is then given by $t = -\omega/(4\gamma)$.

---

* Contributed equally to this work.
† Electronic address: jingbo.wang@uwa.edu.au
‡ Electronic address: jonathan.matthews@bristol.ac.uk

[1] Farhi, E. & Gutmann, S. Quantum computation and decision trees. *Phys. Rev. A* **58,** 915 (1998).

[2] Kempe, J. Quantum random walks: an introductory overview. *Contemp. Phys.* **44,** 307-327 (2003).

[3] Childs, A. M., Gosset, D. & Webb, Z. Universal computation by multiparticle quantum walk. *Science* **339,** 791-794 (2013).

[4] Childs, A. M. & Goldstone, J. Spatial search by quantum walk. *Phys. Rev. A* **70,** 022314 (2004).

[5] Douglas, B. L. & Wang, J. B. A classical approach to the graph isomorphism problem using quantum walks. *J. Phys. A* **41,** 075303 (2008).

[6] Gamble, J. K., Friesen, M., Zhou, D., Joynt, R. & Coppersmith, S. N. Two-particle quantum walks applied to the graph isomorphism problem. *Phys. Rev. A* **81,** 052313 (2010).

[7] Berry, S. D. & Wang, J. B. Two-particle quantum walks: Entanglement and graph isomorphism testing. *Phys. Rev. A* **83,** 042317 (2011).

[8] Berry, S. D. & Wang, J. B. Quantum-walk-based search and centrality. *Phys. Rev. A* **82,** 042333 (2010).

[9] Sánchez-Burillo, E., Duch, J., Gómez-Gardeñes, J. & Zueco, D. Quantum navigation and ranking in complex networks. *Sci. Rep.* **2,** 605 (2012).

[10] Lloyd, S. Universal quantum simulators. *Science* **273,** 1073-1078 (1996).

[11] Berry, D. W. & Childs, A. M. Black-box hamiltonian simulation and unitary implementation. *Quantum Inf. Comput.* **12,** 29-62 (2012).

[12] Schreiber, A. *et al.* A 2D quantum walk simulation of two-particle dynamics. *Science* **336,** 55-58 (2012).

[13] Engel, G. S. *et al.* Evidence for wavelike energy transfer through quantum coherence in photosynthetic systems. *Nature* **446,** 782-786 (2007).

[14] Rebentrost, P. *et al.* Environment-assisted quantum transport. *New J. Phys.* **11,** 033003 (2009).

[15] Du, J. *et al.* Experimental implementation of the quantum random-walk algorithm. *Phys. Rev. A* **67,** 042316 (2003).

[16] Ryan, C. A. *et al.* Experimental implementation of a discrete-time quantum random walk on an NMR quantum-information processor. *Phys. Rev. A* **72,** 062317 (2005).

[17] Do, B. *et al.* Experimental realization of a quantum quincunx by use of linear optical elements. *J. Opt. Soc. Am. B* **22,** 499-504 (2005).

[18] Schreiber, A. *et al.* Photons walking the line: a quantum walk with adjustable coin operations. *Phys. Rev. Lett.* **104,** 050502 (2010).

[19] Xue, P., Sanders, B. C. & Leibfried, D. Quantum walk on a line for a trapped ion. *Phys. Rev. Lett.* **103,** 183602

(2009).

[20] Schmitz, H. *et al.* Quantum walk of a trapped ion in phase space. *Phys. Rev. Lett.* **103,** 090504 (2009).

[21] Zähringer, F. *et al.* Realization of a quantum walk with one and two trapped ions. *Phys. Rev. Lett.* **104,** 100503 (2010).

[22] Karski, M. *et al.* Quantum walk in position space with single optically trapped atoms. *Science* **325,** 174-177 (2009).

[23] Perets, H. B. *et al.* Realization of quantum walks with negligible decoherence in waveguide lattices. *Phys. Rev. Lett.* **100,** 170506 (2008).

[24] Carolan, J. *et al.* On the experimental verification of quantum complexity in linear optics. *Nat. Photonics* **8,** 621 (2014).

[25] Manouchehri, K. & Wang, J. B. *Physical Implementation of Quantum Walks.* Springer-Verlag, Berlin, (2014).

[26] Exceptions, such as [15], adopted the qubit model, but there was no discussion on potentially efficient implementation of quantum walks.

[27] Ng, M. K. *Iterative Methods for Toeplitz Systems.* Oxford University Press New York, (2004).

[28] Aharonov, D. & Ta-Shma, A. Adiabatic quantum state generation and statistical zero knowledge. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing* 20-29. ACM (2003).

[29] Berry, D. W., Ahokas, G., Cleve, R. & Sanders, B. C. Efficient quantum algorithms for simulating sparse hamiltonians. *Commun. Math. Phys.* **270,** 359-371 (2007).

[30] Childs, A. M. & Kothari, R. Limitations on the simulation of non-sparse hamiltonians. *Quantum Inf. Comput.* **10,** 669-684 (2009).

[31] Childs, A. M. On the relationship between continuous- and discrete-time quantum walk. *Commun. Math. Phys.* **294,** 581-603 (2010).

[32] Gray, R. M. *Toeplitz and Circulant Matrices: A Review.* Now Publishers Inc., (2006).

[33] Buhrman, H., Cleve, R., Watrous, J. & De Wolf, R. Quantum fingerprinting. *Phys. Rev. Lett.* **87,** 167902 (2001).

[34] Han, D. *et al.* Folding and cutting DNA into reconfigurable topological nanostructures. *Nat. Nanotechnol* **5,** 712-717 (2010).

[35] Peres, A. Stability of quantum motion in chaotic and regular systems. *Phys. Rev. A* **30,** 1610 (1984).

[36] Prosen, T. & Znidaric, M. Stability of quantum motion and correlation decay. *J. Phys. A: Math. Gen.* **35,** 1455 (2002).

[37] Aaronson, S. & Arkhipov, A. The computational complexity of linear optics. In *Proceedings of the forty-third annual ACM symposium on Theory of computing* 333–342. ACM (2011).

[38] Bremner, M. J., Jozsa, R. & Shepherd, D. J. Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, rspa.2010.0301 (2010).

[39] Bremner, M. J., Montanaro, A. & Shepherd, D. J. Average-case complexity versus approximate simulation of commuting quantum computations, Preprint at `http://arxiv.org/abs/1504.07999` (2015).

[40] Nielsen, N. A. & Chuang, I. L. *Quantum Computation and Quantum Information.* Cambridge University Press, (2010).

[41] Childs, A. M. *Quantum information processing in continuous time.* PhD Thesis, Massachusetts Institute of Technology (2004).

[42] Shepherd, D. & Bremner, M. J. Temporally unstructured quantum computation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science* **465,** 1413-1439 (2009).

[43] Fujii, K. & Morimae, T. Quantum commuting circuits and complexity of Ising partition functions, Preprint at `http://arxiv.org/abs/1311.2128` (2013).

[44] Goldberg, L. A. & Guo, H. The complexity of approximating complex-valued Ising and Tutte partition functions, Preprint at `http://arxiv.org/abs/1409.5627` (2014).

[45] Stockmeyer, L. J. On approximation algorithms for #P. *SIAM J. Comput.* **14,** 849-861 (1985).

[46] Papadimitriou, C. *Computational Complexity.* Addison-Wesley, (1994).

[47] Nest, M. Simulating quantum computers with probabilistic methods. *Quantum Inf. Comput.* **11,** 784-812 (2011).

[48] Schwarz, M. & Nest, M. Simulating quantum circuits with sparse output distributions, Preprint at `http://arxiv.org/abs/1310.6749` (2013).

[49] Zhou, X. Q. *et al.* Adding control to arbitrary unknown quantum operations. *Nat. Commun.* **2,** 413 (2011).

[50] Knill, E., & Laflamme, R. Power of one bit of quantum information. *Phys. Rev. Lett.* **81,** 5672 (1998).

[51] Rohde, P. P. & Ralph, T. C. Error tolerance of the boson-sampling model for linear optics quantum computing. *Phys. Rev. A* **85,** 022332 (2012).

[52] Rangarajan, R., Goggin, M. & Kwiat, P. Optimizing type-I polarization-entangled photons. *Opt. Express* **17,** 18920-18933 (2009).

Appendix

FURTHER DETAILS ON CIRCULANT GRAPHS AND OTHER EXAMPLES

A circulant graph of N vertices is fully described by an N-by-N symmetric circulant adjacency matrix C defined as follows.

$$C = \begin{bmatrix} c_0 & c_1 & c_2 & \dots & c_{N-1} \\ c_{N-1} & c_0 & c_1 & \dots & c_{N-2} \\ c_{N-2} & c_{N-1} & c_0 & \dots & c_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_1 & c_2 & c_3 & \dots & c_0 \end{bmatrix} \qquad (A1)$$

where $c_j = c_{N-j}, j = 1, 2, \cdots, N - 1$. Obviously, every circulant matrix can be generated given any row of the matrix – conventionally we use the first row of the matrix, denoted as r_C. It is clear that C has at most N distinct eigenvalues which are given by $\lambda_m = \sum_{k=0}^{N-1} c_k \omega^{-mk}$, where $\omega = \exp(2\pi i / N)$ and $m = 0, 1, \ldots, N - 1$ [1]. If C is singular, some of the eigenvalues of C are zeros. The complete graph and complete bipartite graph are straightforward examples of circulant graphs with few distinct eigenvalues.

There are also some other interesting examples of circulant graph such as self-complementary circulant graphs and Paley graphs with prime order [2, 3]. Both of these two families of graphs are also strongly regular graphs which have only three distinct eigenvalues. For example, the Paley graph on 13 vertices has three distinct eigenvalues: 6 (with multiplicity 1) and $\frac{1}{2}\left(-1 \pm \sqrt{13}\right)$ (both with multiplicity 6), and thus the diagonal unitary $\exp(-it\Lambda)$ can be implemented efficiently. We note here it is required to implement QFT (and its inverse) for the dimension of 13, which does not have the form of $N = 2^n$. The QFT on general dimensions can be implemented by means of amplitude amplification with extra qubit registers to perform the computation [4]. Alternatively, approximate versions of the QFT on general dimensions have also been developed [5].

IMPLEMENTATION OF THE DIAGONAL UNITARY OPERATOR

We say that the eigenvalues of a circulant graph can be characterised efficiently, if they can be calculated efficiently classically. In other words, the eigenvalue matrix Λ of the given circulant Hamiltonian can be efficiently computed, and thus the diagonal unitary operator $\exp(-it\Lambda)$ can be efficiently implemented [6]. Specifically, there exists a quantum circuit shown in Figure A1, which transforms a computational basis state $|x\rangle$, to-
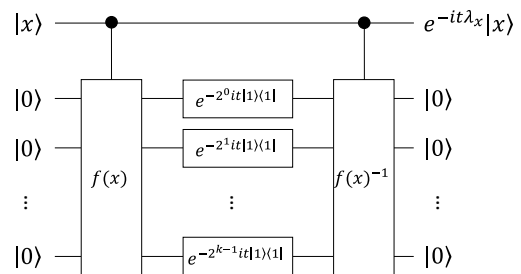

FIG. A1: The quantum circuit for implementing the diagonal unitary operator $\exp(-it\Lambda)$ of the given circulant Hamiltonian. The eigenvalues of given circulant can be calculated by the function $f(x)$ efficiently classically.

gether with a k-qubit ancilla $|0\rangle$ for $k = \text{poly}(n)$, as

$$\begin{aligned} |x\rangle|0\rangle &\to |x\rangle|\lambda_x\rangle \\ &\to e^{-it\lambda_x}|x\rangle|\lambda_x\rangle \\ &\to e^{-it\lambda_x}|x\rangle|0\rangle = e^{-it\Lambda}|x\rangle|0\rangle \qquad (A2) \end{aligned}$$

where $x = 0, 1, \ldots, N - 1$. Note that here we assume λ_x can be expressed exactly as a rational number with k bits of precision. If this is not the case, truncating λ_x to k bits of precision will introduce an error which can be made arbitrarily small by taking large enough $k = \text{poly}(n)$. The function $f(x)$ returns λ_x for any given x. λ_x is always a real number since the adjacency matrix is symmetric.

For example, for the case of the cycle graph of $N = 2^n$ vertices, there are essentially $N/2$ distinct eigenvalues simply given by $\lambda_x = 2\cos(2\pi x/N)$, where $x = 0, 1, \ldots, N - 1$. And then $f(x)$ will be the cosine function that can be computed with a number of operations polynomial in n, using a reversible equivalent of classical algorithms to compute trigonometric functions, e.g. the Taylor approximation. In general, given a sparse circulant graph which has only $\text{poly}(n)$ 1s in the first row r_C of its adjacency matrix, an efficient function $f(x)$ can be given as

$$f(x) = \sum_{y \in S} e^{2i\pi xy/N} \qquad (A3)$$

where S is the set of positions for which the first row in nonzero. $f(x)$ is a sum of $|S| = \text{poly}(n)$ numbers, taking $O(\text{poly}(n))$ time to compute. For a non-sparse circulant graph, its eigenvalues are still possible to be calculated efficiently classically. Some straightforward examples are complete graph, complete bipartite graph $K_{N,N}$ and cocktail party graph. Therefore, together with the quantum circuits of QFT and the inverse of QFT, we construct an efficient quantum circuit for implementing CTQW on the circulant graph whose eigenvalues can be computed efficiently classically.
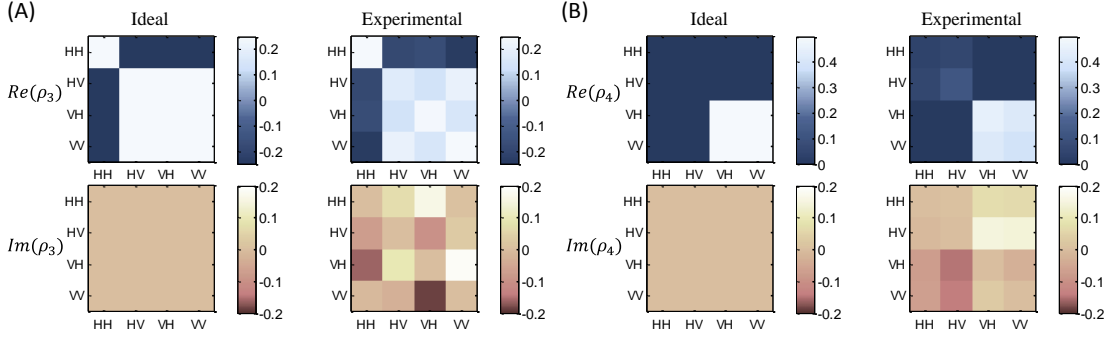
FIG. A2: (A) The ideal theoretical and experimentally reconstructed density matrices for the states $|\varphi_{out}\rangle_3 = \frac{1}{2}[1, -1, -1, -1]'$ (corresponding to ρ_3) and (B) $|\varphi_{out}\rangle_4 = \frac{1}{\sqrt{2}}[0, 0, -1, -1]'$ (corresponding to ρ_4). Both of the real and imaginary parts of the density matrices are obtained through the maximum likelihood estimation technique, and shown as $Re(\rho)$ and $Im(\rho)$ respectively. The achieved fidelities are 88.63±1.24% and 91.53±0.43% respectively.

COMPLEXITY ANALYSIS OF "SWAP TEST"

Unlike the sampling problem we discussed in the main text, the scenario of "SWAP test", where we compare two unitary processes $Q^\dagger D Q$ and $Q^\dagger \widetilde{D} Q$, could sometimes be easier for a classical computer. Imagine we start each process in the state $|\phi_{ini}\rangle = |\varphi_{ini}\rangle = |0\rangle^{\otimes n}$. Then the overlap O between the resulting output states approximated by the SWAP test satisfies

$$O = |\langle 0|^{\otimes n} (Q^\dagger D Q)(Q^\dagger \widetilde{D} Q) |0\rangle^{\otimes n}|^2$$
$$= |\langle +|^{\otimes n} D\widetilde{D} |+\rangle^{\otimes n}|^2 = \Big|\frac{1}{2^n} \sum_{x \in \{0,1\}^n} D_{xx}\widetilde{D}_{xx}\Big|^2,$$
$$(A4)$$

where D_{xx} is the value at position x on the diagonal of D. O can be approximated by a classical algorithm up to $O(1/\text{poly}(n))$ additive error. The algorithm simply takes the average of $\text{poly}(n)$ values of the product $D_{xx}\widetilde{D}_{xx}$ for uniformly random x. For each x, this value can be computed exactly in polynomial time.

This highlights that the complexity of comparing $Q^\dagger D Q |\phi_{ini}\rangle$ and $Q^\dagger \widetilde{D} Q |\varphi_{ini}\rangle$ depends on the choice of input states $|\phi_{ini}\rangle$ and $|\varphi_{ini}\rangle$. In full generality, one could allow these to be arbitrary states produced by a polynomial-time quantum computation; the state comparison problem would then be BQP-complete, but for rather trivial reasons. We expect that the problem would remain classically hard for choices of initial states relevant, for example, to quantum-chemistry applications.

On the other hand, the SWAP test can still be used as in Scenario (B) in the main text to compare the evolution of two Hamiltonians, one of which is not circulant but is efficiently implementable. In this case, the comparison problem is also BQP-complete, and hence expected to be hard for a classical computer.

FURTHER EXPERIMENTAL RESULTS

We present the ideal and experimentally sampled probability distributions of CTQW with initial states $|\varphi_{ini}\rangle_1 = [1, 0, 0, 0]'$, $|\varphi_{ini}\rangle_2 = \frac{1}{\sqrt{2}}[1, 1, 0, 0]'$ (mentioned in the main text), $|\varphi_{ini}\rangle_3 = \frac{1}{\sqrt{2}}[1, -1, 0, 0]'$, $|\varphi_{ini}\rangle_4 = \frac{1}{\sqrt{2}}[1, -i, 0, 0]'$, $|\varphi_{ini}\rangle_5 = \frac{1}{2}[1, i, 1, i]'$, $|\varphi_{ini}\rangle_6 = \frac{1}{2}[1, i, i, -1]'$, in Table I. The achieved average fidelities between ideal and experimental probability distributions are 96.68±0.27%, 95.82±0.25%, 92.61±0.21%, 96.36±0.16%, 98.76±0.17% and 97.27±0.24% respectively. In the main text, we reconstructed the density matrices for the two quantum states $|\varphi_{out}\rangle_1$ and $|\varphi_{out}\rangle_2$, through performing quantum state tomography. Here we also present the reconstructed density matrices for another two evolution states $|\varphi_{out}\rangle_3 = \exp(-iH\frac{3}{4}\pi)|\varphi_{ini}\rangle_1$ and $|\varphi_{out}\rangle_4 = \exp(-iH\frac{3}{4}\pi)|\varphi_{ini}\rangle_2$, with the achieved fidelities of 88.63±1.24% and 91.53±0.53% respectively. See in Figure A2. The four reconstructed density matrices ρ_1, ρ_2, ρ_3 and ρ_4 for quantum states $|\varphi_{out}\rangle_1$, $|\varphi_{out}\rangle_2$, $|\varphi_{out}\rangle_3$ and $|\varphi_{out}\rangle_4$ are shown as follows.

$$\rho_1 = \begin{bmatrix} 0.4763 & -0.1175 - 0.1281i & -0.1410 - 0.0112i & -0.1507 - 0.3104i \\ -0.1175 + 0.1281i & 0.1354 & 0.0257 + 0.0115i & 0.1620 + 0.0133i \\ -0.1410 + 0.0112i & 0.0257 - 0.0115i & 0.0841 & 0.0289 + 0.0797i \\ -0.1507 + 0.3104i & 0.1620 - 0.0133i & 0.0289 - 0.0797i & 0.3041 \end{bmatrix}$$

$$\rho_2 = \begin{bmatrix} 0.3207 & 0.2801 - 0.0479i & 0.0665 - 0.1666i & 0.0851 - 0.1810i \\ 0.2801 + 0.0479i & 0.2575 & 0.0812 - 0.1135i & 0.0989 - 0.1245i \\ 0.0665 + 0.1666i & 0.0812 + 0.1135i & 0.1899 & 0.2044 + 0.0088i \\ 0.0851 + 0.1840i & 0.0989 + 0.1245i & 0.2044 - 0.0088i & 0.2319 \end{bmatrix}$$

$$\rho_3 = \begin{bmatrix} 0.2779 & -0.1804 + 0.0749i & -0.1711 + 0.1613i & -0.2397 + 0.0091i \\ -0.1804 - 0.0749i & 0.1778 & 0.1376 - 0.0938i & 0.2083 + 0.0311i \\ -0.1711 - 0.1613i & 0.1376 + 0.0938i & 0.2411 & 0.1507 + 0.1825i \\ -0.2397 - 0.0091i & 0.2083 - 0.0311i & 0.1507 - 0.1825i & 0.3031 \end{bmatrix}$$

$$\rho_4 = \begin{bmatrix} 0.0418 & 0.0605 + 0.0067i & -0.0353 + 0.0698i & -0.0287 + 0.0671i \\ 0.0605 - 0.0067i & 0.1196 & -0.0260 + 0.1465i & -0.0155 + 0.1381i \\ -0.0353 - 0.0698i & -0.0260 - 0.1465i & 0.4481 & 0.4175 - 0.0263i \\ -0.0287 - 0.0671i & -0.0155 - 0.1381i & 0.4175 + 0.0263i & 0.3904 \end{bmatrix}$$

[*] Contributed equally to this work.
[†] Electronic address: jingbo.wang@uwa.edu.au
[‡] Electronic address: jonathan.matthews@bristol.ac.uk

[1] Ng, M. K. *Iterative Methods for Toeplitz Systems.* Oxford University Press New York, (2004).
[2] Zhou, H. On self-complementary of circulant graphs. In *Theoretical and Mathematical Foundations of Computer Science* 464-471, Springer (2011).
[3] Rajasingh, I. & Natarajan, P. Spanning trees in circulant networks. 2013.
[4] Mosca, M. & Zalka, C. Exact quantum fourier transforms and discrete logarithm algorithms. *Int. J. Quantum Inf.* **2** 91-100 (2004).
[5] Hales, L. & Hallgren, S. An improved quantum fourier transform algorithm and applications. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on* 515-525, IEEE (2000).
[6] Childs, A. M. *Quantum information processing in continuous time.* PhD Thesis, Massachusetts Institute of Technology (2004).

Node/T	0	$\frac{1}{8}\pi$	$\frac{2}{8}\pi$	$\frac{3}{8}\pi$	$\frac{4}{8}\pi$	$\frac{5}{8}\pi$	$\frac{6}{8}\pi$	$\frac{7}{8}\pi$	π
$P_{1,ideal}$	1	0.625	0.25	0.625	1	0.625	0.25	0.625	1
$P_{2,ideal}$	0	0.125	0.25	0.125	0	0.125	0.25	0.125	0
$P_{3,ideal}$	0	0.125	0.25	0.125	0	0.125	0.25	0.125	0
$P_{4,ideal}$	0	0.125	0.25	0.125	0	0.125	0.25	0.125	0
$P_{1,exp}$	0.8225	0.5014	0.2139	0.5759	0.8482	0.4583	0.2414	0.549	0.858
$P_{2,exp}$	0.003	0.1388	0.2254	0.1455	0.0078	0.2167	0.2375	0.1324	0.0114
$P_{3,exp}$	0.1598	0.153	0.2659	0.2105	0.1284	0.1333	0.2299	0.1912	0.1193
$P_{4,exp}$	0.0148	0.2068	0.2948	0.0681	0.0156	0.1917	0.2912	0.1275	0.0114

Node/T	0	$\frac{1}{8}\pi$	$\frac{2}{8}\pi$	$\frac{3}{8}\pi$	$\frac{4}{8}\pi$	$\frac{5}{8}\pi$	$\frac{6}{8}\pi$	$\frac{7}{8}\pi$	π
$P_{1,ideal}$	0.5	0.25	0	0.25	0.5	0.25	0	0.25	0.5
$P_{2,ideal}$	0.5	0.25	0	0.25	0.5	0.25	0	0.25	0.5
$P_{3,ideal}$	0	0.25	0.5	0.25	0	0.25	0.5	0.25	0
$P_{4,ideal}$	0	0.25	0.5	0.25	0	0.25	0.5	0.25	0
$P_{1,exp}$	0.4386	0.2796	0.0682	0.2927	0.4375	0.2823	0.0669	0.2338	0.434
$P_{2,exp}$	0.4189	0.2607	0.0746	0.2717	0.4103	0.3008	0.0605	0.2	0.4415
$P_{3,exp}$	0.1031	0.2156	0.3945	0.2482	0.0679	0.2058	0.4108	0.2923	0.0566
$P_{4,exp}$	0.0395	0.2441	0.4627	0.1874	0.0842	0.2111	0.4618	0.2738	0.0679

Node/T	0	$\frac{1}{8}\pi$	$\frac{2}{8}\pi$	$\frac{3}{8}\pi$	$\frac{4}{8}\pi$	$\frac{5}{8}\pi$	$\frac{6}{8}\pi$	$\frac{7}{8}\pi$	π
$P_{1,ideal}$	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
$P_{2,ideal}$	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
$P_{3,ideal}$	0	0	0	0	0	0	0	0	0
$P_{4,ideal}$	0	0	0	0	0	0	0	0	0
$P_{1,exp}$	0.4147	0.3918	0.3865	0.4289	0.4156	0.382	0.4058	0.4356	0.4123
$P_{2,exp}$	0.4627	0.474	0.4679	0.4348	0.4359	0.4697	0.4507	0.4189	0.4416
$P_{3,exp}$	0.0601	0.0671	0.0722	0.0782	0.0898	0.0877	0.0792	0.0933	0.0823
$P_{4,exp}$	0.0625	0.0671	0.0734	0.0581	0.0587	0.0606	0.0642	0.0522	0.0639

Node/T	0	$\frac{1}{8}\pi$	$\frac{2}{8}\pi$	$\frac{3}{8}\pi$	$\frac{4}{8}\pi$	$\frac{5}{8}\pi$	$\frac{6}{8}\pi$	$\frac{7}{8}\pi$	π
$P_{1,ideal}$	0.5	0.125	0.25	0.625	0.5	0.125	0.25	0.625	0.5
$P_{2,ideal}$	0.5	0.625	0.25	0.125	0.5	0.625	0.25	0.125	0.5
$P_{3,ideal}$	0	0.125	0.25	0.125	0	0.125	0.25	0.125	0
$P_{4,ideal}$	0	0.125	0.25	0.125	0	0.125	0.25	0.125	0
$P_{1,exp}$	0.4178	0.1655	0.1969	0.4932	0.4729	0.1492	0.1977	0.4217	0.4332
$P_{2,exp}$	0.3541	0.548	0.2749	0.1504	0.3824	0.6258	0.2503	0.1085	0.4308
$P_{3,exp}$	0.1376	0.1015	0.211	0.1467	0.0594	0.1047	0.2316	0.2796	0.0734
$P_{4,exp}$	0.0904	0.185	0.3171	0.2096	0.0853	0.1203	0.3205	0.1902	0.0626

Node/T	0	$\frac{1}{8}\pi$	$\frac{2}{8}\pi$	$\frac{3}{8}\pi$	$\frac{4}{8}\pi$	$\frac{5}{8}\pi$	$\frac{6}{8}\pi$	$\frac{7}{8}\pi$	π
$P_{1,ideal}$	0.25	0.5	0.25	0	0.25	0.5	0.25	0	0.25
$P_{2,ideal}$	0.25	0	0.25	0.5	0.25	0	0.25	0.5	0.25
$P_{3,ideal}$	0.25	0.5	0.25	0	0.25	0.5	0.25	0	0.25
$P_{4,ideal}$	0.25	0	0.25	0.5	0.25	0	0.25	0.5	0.25
$P_{1,exp}$	0.2642	0.4163	0.2227	0.0172	0.2191	0.4136	0.2167	0.0591	0.2476
$P_{2,exp}$	0.2724	0.0156	0.3	0.4678	0.2367	0.0227	0.2808	0.4864	0.199
$P_{3,exp}$	0.2561	0.5525	0.2409	0.0043	0.3145	0.5545	0.2266	0.0227	0.3204
$P_{4,exp}$	0.2073	0.0156	0.2364	0.5107	0.2297	0.0091	0.2759	0.4318	0.233

Node/T	0	$\frac{1}{8}\pi$	$\frac{2}{8}\pi$	$\frac{3}{8}\pi$	$\frac{4}{8}\pi$	$\frac{5}{8}\pi$	$\frac{6}{8}\pi$	$\frac{7}{8}\pi$	π
$P_{1,ideal}$	0.25	0.625	0.5	0.125	0.25	0.625	0.5	0.125	0.25
$P_{2,ideal}$	0.25	0.125	0	0.125	0.25	0.125	0	0.125	0.25
$P_{3,ideal}$	0.25	0.125	0	0.125	0.25	0.125	0	0.125	0.25
$P_{4,ideal}$	0.25	0.125	0.5	0.625	0.25	0.125	0.5	0.625	0.25
$P_{1,exp}$	0.2056	0.4226	0.3307	0.1129	0.168	0.4883	0.3425	0.1321	0.3347
$P_{2,exp}$	0.1542	0.2469	0.0906	0.0887	0.127	0.1596	0.0827	0.0566	0.3431
$P_{3,exp}$	0.2477	0.1674	0.0354	0.1411	0.3238	0.1643	0.0276	0.1358	0.1715
$P_{4,exp}$	0.3925	0.1632	0.5433	0.6573	0.3811	0.1878	0.5472	0.6755	0.1506

TABLE I: The ideal theoretical and experimental probability distributions of CTQWs on K_4 graph with initial states $|\varphi_{ini}\rangle_1 = [1, 0, 0, 0]'$, $|\varphi_{ini}\rangle_2 = \frac{1}{\sqrt{2}}[1, 1, 0, 0]'$, $|\varphi_{ini}\rangle_3 = \frac{1}{\sqrt{2}}[1, -1, 0, 0]'$, $|\varphi_{ini}\rangle_4 = \frac{1}{\sqrt{2}}[1, -i, 0, 0]'$, $|\varphi_{ini}\rangle_5 = \frac{1}{2}[1, i, 1, i]'$, $|\varphi_{ini}\rangle_6 = \frac{1}{2}[1, i, i, -1]'$ shown in six sub-tables from top to bottom. For each sub-table, the first four rows are ideal results and the last four rows are experimental results.