A quantum walk control plane for distributed quantum computing in quantum networks

Matheus Guedes de Andrade[1], Wenhan Dai[1, 2], Saikat Guha[3], and Don Towsley[1]

[1]College of Information and Computer Science, University of Massachusetts Amherst
[2]Quantum Photonics Laboratory, Massachusetts Institute of Technology
[3]College of Optical Sciences, University of Arizona

Abstract—Quantum networks are complex systems formed by the interaction among quantum processors through quantum channels. Analogous to classical computer networks, quantum networks allow for the distribution of quantum computation among quantum computers. In this work, we describe a quantum walk protocol to perform distributed quantum computing in a quantum network. The protocol uses a quantum walk as a quantum control signal to perform distributed quantum operations. We consider a generalization of the discrete-time coined quantum walk model that accounts for the interaction between a quantum walker system in the network graph with quantum registers inside the network nodes. The protocol logically captures distributed quantum computing, abstracting hardware implementation and the transmission of quantum information through channels. Control signal transmission is mapped to the propagation of the walker system across the network, while interactions between the control layer and the quantum registers are embedded into the application of coin operators. We demonstrate how to use the quantum walker system to perform a distributed CNOT operation, which shows the universality of the protocol for distributed quantum computing. Furthermore, we apply the protocol to the task of entanglement distribution in a quantum network.

I. INTRODUCTION

Quantum networking is an innovative, multidisciplinary field of research that promises revolutionary improvements in communications, enabling tasks and applications that are impossible to achieve with the exclusive exchange of classical information [1], [2]. Similar to a classical computer network, a quantum network is a distributed system composed of quantum computers and quantum repeaters that exchange quantum information across physical channels. Among applications supported by quantum networks, distributed quantum computing is of particular interest as it leverages the power of interconnected quantum computers to create a virtual quantum machine with processing capabilities that surpass its physical constituents

alone [3]–[5]. Distributed quantum computing becomes even more interesting in the noise intermediate scale quantum machines (NISQ) scenario where there is a clear tradeoff between the size of quantum computers, in terms of number of qubits, and the fidelity of quantum operations, given the fact that physical separation directly reduces cross talk among qubits [6]. When the quantum network scenario is considered, the complexity of distributed quantum computing extends in at least two dimensions. First, physical quantum channels have a well known depleting effect in the exchange of quantum data, e.g the exponential decrease in channel entanglement rate with distance [7]. Second, there is a demand for a quantum network protocol capable of performing a desired distributed quantum operation while accounting for network connectivity. Generic quantum computation with qubits in distinct quantum processors demands either the application of remote controlled gates [8] or the continuous exchange of quantum information. For both cases, a network protocol is necessary to orchestrate the communication between nodes that are not directly connected with one another.

One challenge in the design of a control protocol is the need for it to be agnostic to hardware implementations. There is a plethora of physical systems suited for quantum computation under investigation, superconducting qubits [9], trapped ions [10], [11] and Silicon-vacancy color centers in diamond [12], [13] to name a few. In addition, there is a diverse investigation in the architectural description of quantum interconnecting devices capable of exchanging quantum information encoded in distinct quantum physical quantities [14]. This diverse ecosystem of quantum network technologies indicates that distributed quantum computing network protocols need to abstract physical implementations of quantum switches and network connectivity while maintaining universality requirements.

The goal of this article is to propose a control protocol for distributed quantum computing based on discrete time quantum walks [15]. Quantum walks are universal for quantum computing [16]–[18] and have been successfully employed in the quantum network scenario to perform perfect state transfer (PST) between network nodes [19], [20], teleportation [21], [22] and quantum key distribution (QKD) [23]. Previous works describe ways of distributing entanglement between nodes on a quantum network using the coin space of the walker to propel entanglement generation between qubits [20], [21]. In addition, the formalism of quantum walks with multiple coins enabled the description of an entanglement routing protocol, which interprets qubits within a network node as vertices of an abstract graph used by a quantum walker to generate entanglement [24]. In spite of their relevance, the quantum walk approaches defined in the literature are suited to particular network structures and quantum operations. In particular, they consider the case of regular lattices, describing walker dynamics on regular structures and do not address how the quantum walk can be used to perform generic distributed quantum operators. In this context, this work adds to the literature of both quantum walks and quantum networks with the description of a quantum network control protocol that can be applied to arbitrary graphs and perform universal quantum computing in a quantum network.

A. Contributions

The contributions of this article are three-fold.

- We propose a quantum walk protocol for distributed quantum computing in a quantum network. The protocol uses a quantum walker system as a quantum control signal to perform computations among quantum processors that are physically separated. We assume that each processor dedicates part of its internal quantum register to represent the walker control signal and describe how the control subsystem interacts with the data subsystem. The interaction between data and control is specified by unitary operations that nodes need to implement in order to realize the quantum walk control plane.
- We show the universality of the protocol by describing how a 2-qubit controlled X (CNOT) operation between qubits in distinct nodes of the network can be performed with the quantum walk. The protocol is universal in the sense that it allows for any quantum operation in the Hilbert space formed by all qubits in the data subsystem of the

nodes. Furthermore, it is generic in the sense that it abstracts hardware implementation and channel transmissions, while being well-defined for any network topology

- We demonstrate how the protocol can be used to recover the behavior of entanglement distribution protocols previously described in the literature [25], [26].

The description of the protocol is carried in the logical setting under the assumption that quantum error correction processes ensure unitary operations for both control and data qubits.

The remainder of this article is structured as follows. In Section 2, we present the mathematical background needed for the description of the quantum walk protocol. We describe the quantum walk protocol and demonstrate its universality in Section 3. The description is extended to case of multiple walkers in Section 4. In section 5, we apply the protocol to recover the behavior of entanglement distribution protocols. Finally, the manuscript is concluded in Section 6.

II. BACKGROUND

Consider the graph $G = (V, E)$. Let $\delta(v)$ denote the set of neighbors of vertex $v \in V$ and $d(v) = |\delta(v)|$ denote the degree of v. Let $\Delta(u, v)$ denote the hop-distance between u and v in G. Throughout this work we refer to the inverse of a binary string $x \in \{0, 1\}^*$ as $\overline{x}$. A quantum network is a set of quantum hosts (quantum processors) interconnected by a set of quantum channels that allow for the exchange of quantum information [2]. A host is either a quantum repeater, a quantum router or a quantum computer with a fixed number of qubits, which performs generic quantum operations. A quantum network can be represented as a symmetric directed graph $N = (V, E)$. Each node $v \in V$ represents a quantum host that has a set $\mathcal{M}_v$ of qubits that can be processed together at any time and a set $\mathcal{N}_v$ of qubits that is used to exchange quantum information with nodes in its neighborhood $\delta(v)$ through a set of quantum channels. More precisely, each edge $(u, v) \in E$ represents a quantum channel connecting the qubits in $\mathcal{N}_u$ and $\mathcal{N}_v$ which can interact through operations mediated by the channel. We will refer to $\mathcal{N}_v$ and $\mathcal{M}_v$ as the networking and data registers of node v, respectively. The sets $\mathcal{M} = \bigcup_v \mathcal{M}_v$ and $\mathcal{N} = \bigcup_v \mathcal{N}_v$ are respectively referred to as the *network control plane* and the *network data plane*.

This network model separates the qubit registers in the nodes into control and data registers. This choice differs from previous works where the set of

314

qubits $\mathcal{N}$ that can interact with quantum channels are considered alone [25], [26]. Note that considering only networking qubits suffices to define entanglement routing protocols. Nonetheless, it is straightforward for the goal of describing a quantum network control plane protocol to consider the separation between control and data registers in the nodes.

A. Quantum network protocols

Consider the system formed by two quantum processors u and v connected by a channel and their respective qubits. A *local operation* is a quantum transformation represented as a separable operator of the form

$$O = O_u \otimes O_v, \qquad (1)$$

where O_u and O_v acts on the state space of the qubits at processors u and v, respectively. A *local operation assisted by two-way classical communication* (LOCC) is a local operation that depends on classical information exchanged between nodes [7], *e.g* the unitaries of quantum teleportation. The classical information is used to select what operators are to be applied to the system, which is embedded in the formalism by averaging non-unitary measurement operators into trace-preserving operators. A quantum network protocol for N is an algorithm that operates on the qubits at the nodes, transforming their joint state by an LOCC $\bar{\Lambda}_t$ as

$$\rho(t) = \bar{\Lambda}_t \rho(0) \bar{\Lambda}_t^\dagger, \qquad (2)$$

where t is the number of rounds in the protocol, ρ is a density operator defined on the Hilbert space $\mathcal{H}_\mathcal{M}$ formed by all qubits on the network, $\rho(0)$ is a density matrix where registers in distinct nodes are in a separable state. We express $\bar{\Lambda}_t = \prod_{k=0}^t \Lambda_k$ as a product of other LOCCs each applied at different time steps [27]. It suffices for $\rho(0)$ to be a separable density operator because the preparation of any non-separable state between registers in different network nodes can be represented by an LOCC. In fact, it is also possible to model a protocol as a sequence of external (mediated by channels) and internal (local to vertices) time-dependent superoperators A_t and B_t such that the state of the network is described as

$$\rho(t+1) = B_t[A_t[\rho(t)]], \qquad (3)$$

where ρ is a density matrix characterizing the joint state of all network memories at discrete time t. As an example, it is usually the case for entanglement

distribution protocols that A_t represents channel entanglement protocols performed in all channels of the network and B_t represents either entanglement swapping operations or GHZ projections performed independently in multiple nodes. The LOCC formalism has proven useful since it allowed for the derivation of fundamental bounds for entanglement rate distribution [27]. In this article, however, our focus is to address network protocols for the transmission of quantum information in the logical perspective. In particular, we exploit the representation in (3) considering A and B as unitary operators provided by the network instead of generic superoperators and perform the analysis in the state vector formalism. It is worth emphasizing that the unitarity assumption for operators A and B translates to the assumption that quantum error correction is provided by the network.

B. Quantum walks on graphs

There are many ways to define a quantum walk on a graph and this article focuses on the discrete-time coined quantum walk model. Given a symmetric directed graph $G = (V, E)$, a coined quantum walk on G is a process of unitary evolution on the Hilbert space $\mathcal{H}_G = \mathcal{H}_V \otimes \mathcal{H}_\mathcal{C}$ formed by the edges of the graph, where $\mathcal{H}_V$ codifies vertices and $\mathcal{H}_\mathcal{C}$ is the coin space of the walker codifying the degrees of freedom the walker can move on. More precisely, every $(v, u) \in E$ defines a basis vector $|v, c\rangle$ for $\mathcal{H}_G$ through a mapping between the edges incident to v with the set of degrees of freedom $\mathcal{C}_v = \{0, 1, \ldots, d(v)-1\}$. As a convention, we map the degrees of freedom of the walker at a given vertex v to the order of labels of its neighbors, such that $|v, c\rangle$ refers to the edge (v, u), where u is the c-th smallest label in $\delta(v)$. Later, c_{vu} will be used to refer to the degree of freedom of v that represents the edge (v, u) and c_v to represent the self-loop (v, v). The generic state of the walker $|\Psi(t)\rangle = \sum \psi(v, c, t) |v, c\rangle$ is a superposition of the edges of G and the walker evolution is defined as

$$|\Psi(t+1)\rangle = S(t)C(t) |\Psi(t)\rangle, \qquad (4)$$

where C and S are respectively referred to as the coin and shift operators. The coin is a unitary operator of the form

$$C(t) = \sum_v |v\rangle\langle v| \otimes C_v(t), \qquad (5)$$

where $C_v : \mathcal{H}_\mathcal{C} \to \mathcal{H}_\mathcal{C}$. The shift can be defined as any permutation operator on the edges of the graph that maps an edge (v_1, u) to an edge (u, v_2). This mapping of edges represents a permutation between states $|v_1, c\rangle$

and $|u, c'\rangle$, where u is the c-th neighbor of v_1. Two shift operators are used throughout this work: the identity operator, which is a trivial permutation of the edges, and the flip-flop shift operator given by

$$S_f = \sum_{v \in V} \sum_{u \in \delta(v)} |v, c_{vu}\rangle\langle u, c_{uv}|, \qquad (6)$$

which applies, for every $(v, u) \in E$, the permutation $(v, u) \rightarrow (u, v)$. S_f reverses edges in the walker wavefunction and is well defined for every symmetric directed graph. The label of nodes and degrees of freedom are numerical and expressed as binary strings. In this setting, the degree of freedom $\bar{c}$ represents the bit-wise negation of label c. Note that c_{vu} is not necessarily $\overline{c_{uv}}$.

III. QUANTUM WALK NETWORK CONTROL PLANE

A direct way of controlling operations in a quantum network with a quantum walk is to consider a coupled system between a walker and the qubits in the nodes, using supersposition in the walker system to implement controlled unitary operations. We describe this joint system in the logical setting. Let $N' = (V, E \cup \{(v, v), \forall v \in V\})$ be the graph obtained by adding self-loops to a network N. $\mathcal{H}_W = \mathcal{H}_V \otimes \mathcal{H}_C$ denotes the space of a walker system on N' and $\mathcal{H}_g = \mathcal{H}_W \otimes \mathcal{H}_\mathcal{M}$ denotes the global Hilbert space spanned by the systems. The walker system is assumed to be implemented in the networking control plane, such that $\mathcal{H}_W$ is a subspace of $\mathcal{H}_\mathcal{N}$. Since the Hilbert space of the walker system represents an edge of N' as a basis vector, the dimension of the Hilbert space $\mathcal{H}_\mathcal{N}$ must be at least $|E|$ for $\mathcal{H}_\mathcal{W}$ to be a subspace. We consider $\mathcal{H}_g$ as a subspace of $\mathcal{H}_\mathcal{N} \otimes \mathcal{H}_\mathcal{M}$.

A. Control-data interactions

We prescribe two laws of interaction between the quantum walk system and the data qubits in the network. The first generalizes a coin operator to the global Hilbert space $\mathcal{H}_g$ following

$$C(t) = \sum_v |v\rangle\langle v| \otimes C_v(t) \otimes U_v(t), \qquad (7)$$

where $C_v(t) : \mathcal{H}_C \rightarrow \mathcal{H}_C$ is a unitary operator on the coin space of the walker and $U_v(t) : \mathcal{H}_\mathcal{M} \rightarrow \mathcal{H}_\mathcal{M}$ is a unitary operator on the data space of the network written as

$$U_v(t) = K_v(t) \bigotimes_{u \neq v} I_{\mathcal{M}_v}, \qquad (8)$$

where $I_{\mathcal{M}_v}$ is the identity operator for the Hilbert space spanned by all qubits of vertex v. Essentially, $C(t)$

applies the operator $K_u(t)$ on qubits in v if, and only if, the walker has a non-zero wavefunction component in v. Note that $U_v(t)$ is the extension of $K_v(t)$, which acts in $\mathcal{H}_{\mathcal{M}_v}$, to $\mathcal{H}_\mathcal{M}$ with the identity operator. Note that the operator described in (7) extends the definition in (5) to perform unitary operations in the data qubit space controlled by vertex position.

The second law of interaction defines controlled operations between the data qubits and the coin space of the walker. In particular, we consider the case where only a single data qubit in a given node is used to control a unitary operation in the coin space of the walker. Let

$$U_{vq}(t) = \sum_{s \in \{0,1\}} U_{vqs}(t) \otimes |s\rangle\langle s| \bigotimes_{q' \neq q} I_{q'}, \qquad (9)$$

be a unitary operator acting on the coin space of vertex v controlled by the qubit $q \in \mathcal{M}_v$, defined for the joint space $\mathcal{H}_C \otimes \mathcal{H}_\mathcal{M}$. The complete interaction has the form

$$O(t) = \sum_v |v\rangle\langle v| \otimes U_{vq}(t). \qquad (10)$$

The operators correspond to a distributed implementation of a quantum walk system in the network. In essence, each node v contributes with $\mathcal{N}_v$ to describe the space $\mathcal{H}_W$ in $\mathcal{H}_\mathcal{N}$. Thus, (7) and (10) are operations controlled by the states of $\mathcal{H}_\mathcal{N}$ that represent walker position since $|v\rangle$ is implemented by an entangled state among qubits in $\mathcal{N}$. This encapsulates the necessary entanglement required for distributed controlled operations between the qubits of the network. As previously mentioned, we assume quantum error correction and describe the walker protocol in terms of logical qubits and operations. In the context of error correcting codes, a single logical qubit is implemented by a set of physical qubits [28], [29]. An implementation of the walker demands $\mathcal{O}(\log(|E|))$ logical qubits in $\mathcal{N}$ since each edge of N' is a basis vector of $\mathcal{H}_W$ and the dimension of the space spanned by qubits is exponential in the number of qubits. For practical implementations, network models that consider physical networking qubits in the order of $\mathcal{O}(|E|)$ have exponentially many physical qubits as the required number of logical qubits for the walker, which is interesting for the purpose of quantum error correction.

B. Universal distributed quantum computing

The interaction behavior described by (7) and (10) suffices for universal quantum computing. Our protocol trivially allows for the application of any separable qubit operator in the network nodes, since there is no need for the interaction between data qubits and the

walker in this case. Thus, we demonstrate universality by demonstrating how a CNOT gate between any pair of qubits a and b, respectively at arbitrary nodes A and B, can be performed using the quantum walk control plane. In fact, the demonstration works for any two-qubit controlled operation by substituting X with the desired single-qubit operator. Without loss of generality, assume that a is the control qubit. To simplify notation, we only define the operators $C(t)$ in (7) and $O(t)$ in (10) for the subspaces spanned by the qubits in nodes A and B, considering undefined operators to be identities, and omit the qubits in $\mathcal{M} \setminus \{a, b\}$. The subscripts following degrees of freedom inside a ket specifies edges while the ones following qubits are considered as indices, $e.g$ $|A, c_A, 0_a, 1_b\rangle \in \mathcal{H}_W \otimes \mathcal{H}_4$ represents the walker in the self-loop (A, A), a in the $|0\rangle$ state and b in the $|1\rangle$ state, where $\mathcal{H}_4$ is the Hilbert space spanned by 2 qubits.

We consider the initial state of the walker to be $|A, c_A\rangle$, which corresponds to the self-loop in A. Thus, the global system is described by the state vector

$$|\Psi(0)\rangle = |A, c_A\rangle \otimes (\alpha |0_a\rangle + \beta |1_a\rangle) \otimes |\Psi_b\rangle, \quad (11)$$

where $|\Psi_b\rangle$ is the state of b. The interaction operator O described in (10) is applied with the CNOT operation

$$U_{Aa} = I_{\mathcal{C}_A} \otimes |0\rangle\langle0| + X_{\mathcal{C}_A} \otimes |1\rangle\langle1|, \quad (12)$$

which trivially generates the entangled state

$$O|\Psi(0)\rangle = (\alpha |A, c_A, 0\rangle + \beta |A, \overline{c_A}, 1\rangle) \otimes |\Psi_b\rangle. \quad (13)$$

This entangled state can be propagated in the network through the evolution of the walker system such that the state after propagation becomes

$$|\Psi(t)\rangle = (\alpha |A, c_A, 0\rangle + \beta |B, c_B, 1\rangle) \otimes |\Psi_b\rangle. \quad (14)$$

We refer to the propagation of the walker as the transmission of the control signal through the network, which will be further explained in detail. The last step refers to the application of the extended coin operator defined in (7), with $K_b = X$ and all other operators defined as identity. The final state obtained is

$$|\Psi(t)\rangle = \alpha |A, c_A, 0, \Psi_b\rangle + \beta |B, c_B, 1, \overline{\Psi_b}\rangle, \quad (15)$$

which clearly shows the application of a CNOT controlled by a with b as a target.

Note that the state obtained in (15) is entangled with the walker subsystem. We will later demonstrate how to separate the data qubits from the walker.

C. Protocol execution in the network

In the context of quantum network protocols, the operators defined by (7) and (10) do not capture the propagation of control information across the network. The propagation of quantum control information between neighbors in the network is embedded in the formalism through the walker's shift operator. The propagation in the case of non-neighboring nodes needs both coin and shift operators as defined in (5) and (6). Note that generic computation on the data qubits can be performed with (7) and (10) given states in the superposition (15), although it is not possible to evolve the walker to that superposition from $|A, c_A\rangle$ without shift operators.

We define coin and shift operators that evolve the global state from (13) to (15). The quantum walk evolution is restricted to neighbor locality, such that, to have the state given in (15) at time t, all of the wavefunction at time $t - 1$ must exclusively be a superposition of edges incident to A, B and its neighbors. There are many ways to define coin and shift operators with this desired behavior and we consider the case where the quantum walk traverses a single path connecting A and B. Some auxiliary definitions and assumptions are required to describe the operators in context. Let p be a minimum path of the network connecting A and B with hop distance $\Delta(A, B)$. We assume that every node knows the network topology and that classical information can be transmitted across the nodes. Recall that the edges of N' are mapped to walker states following the relation

$$(v, u) \rightarrow |v, c_{vu}\rangle, \quad (16)$$

and that the self-loop of node v is mapped to the degree of freedom c_v, for all v. We refer to the edge that connects A to its neighbor in p as $|A, c_A^p\rangle$ and the reverse edge that connects B to its precedent vertex in p as $|B, c_B^p\rangle$. This notation is depicted in Figure 1 for a 2D-grid network. If some quantum operation is performed between A and B at a given time t, the propagation process starts at time $t - \Delta(A, B)$.

The extended shift operator used to route information is fixed and given by

$$S = S_f \otimes I_{\mathcal{M}}, \quad (17)$$

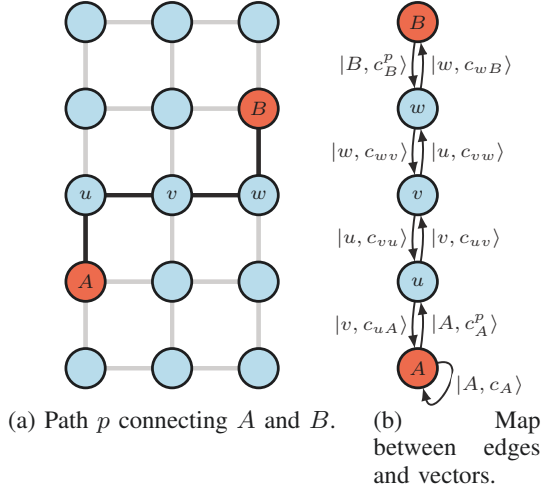where S_f is defined in (6). The coin operator definition is also time-independent, although it depends on the

(a) Path p connecting A and B.

(b) Map between edges and vectors.

Fig. 1: Notation for edges exemplified in a grid graph. Consider that A and B are two nodes connected in a 2D grid network. (a) p is a minimum path connecting A and B with hop-distance 4 traversed by the walker. (b) Each edge on the path corresponds to a vector in $\mathcal{H}_W$, which appear in the walker wavefunction throughout movement. The degrees of freedom are defined such that $|x, c_{xy}\rangle$ represents edge (x, y). As an example, the flip-flop operator specified in (6) maps $|v, c_{vu}\rangle \rightarrow |u, c_{uv}\rangle$, while the operator C_u defined in terms of (18) maps $|u, c_{uA}\rangle \rightarrow |u, c_{uv}\rangle$.

minimum path p chosen. All operators C_v in (7) have the form

$$C_v^p = |c_1\rangle\langle c_2| + |c_2\rangle\langle c_1| + \sum_{\substack{c \in \mathcal{C}_v \\ c \neq c_1, c_2}} |c\rangle\langle c| \qquad (18)$$

where c_1 and c_2 refer to the degrees of freedom that represent the edges incident to v in p. Thus, we define C_v^p specifying c_1 and c_2 for the vertices of interest. The unitary C_A has $c_1 = \overline{c_A}$ and $c_2 = c_A^p$. Let u and w be the neighbors of $v \in p \setminus \{A, B\}$ on the path. C_v has $c_1 = c_{vu}$ and $c_2 = c_{vw}$, representing a permutation between the edges (v, u) and (v, w) in p. Finally, the operator C_B has $c_1 = c_B^p$ and $c_2 = c_B$. It suffices to set $K_v(t) = I_{\mathcal{M}_v}$ in (8) to perform the desired controlled operation between a and b, although it is possible to perform operations controlled by a on the qubits in the intermediate nodes as the walker moves by choosing $K_v(t)$ accordingly.

The overall behavior of the walker is straightforward and is illustrated in Figure 2. It is assumed that the operator O defined in (13) is applied to the system at the beginning of execution. The first application of the coin operator in A creates a superposition between the

states $|A, c_A\rangle$ and $|A, c_A^p\rangle$ that are entangled with a. The flip-flop shift allows the wavefunction in $|A, c_A^p\rangle$ to propagate along p and ensures that the $|A, c_A\rangle$ component remains in the superposition throughout protocol execution. Each coin flip routes information on nodes internally, ensuring propagation. The net effect of $\Delta(A, B)$ successive applications of SC is the superposition specified in (15).

D. Separating data and control

The state prescribed in (15) is an entangled state between control and data. This implies that a partial trace operation in the walker system does not leave the state of a and b as it should be if the operation was performed without the walker. In order to overcome this problem, the walker evolution is reversed after the controlled operation takes place by applying the inverses of the unitary operators used for propagation. Since the coin and shift operators considered are permutation operators, they are Hermitian unitaries and, thus, are their own inverses. It takes $\Delta(A, B)$ time steps to reverse the walker back to A and to transform the joint state of the system to the form

$$\alpha |A, c_A, 0_a, \Psi_b\rangle + \beta |A, \overline{c_A}, 1_a, \overline{\Psi_b}\rangle . \qquad (19)$$

It should be clear that an extended coin operator with interaction of the form given in (7) and a measurement on the walker produces one of the following separable states

$$\begin{cases} |A, c_A\rangle \otimes (\alpha |0_a, \Psi_b\rangle + \beta |1_a, \overline{\Psi_b}\rangle), \\ |A, \overline{c_A}\rangle \otimes (\alpha |0_a, \Psi_b\rangle - \beta |1_a, \overline{\Psi_b}\rangle), \end{cases} \qquad (20)$$

with equal probability. It is straightforward from the separability between the walker system and the qubits a and b that the state after a partial trace operation in the walker system is now equivalent to a CNOT gate, up to a single-qubit Z gate conditioned on the measurement outcome.

Note that moving the walker backwards is only one of several ways to achieve a separable state between the walker and the qubits a and b. In a nutshell, any walker dynamics that concentrates the walker's part of the wavefunction into a single network node suffices for this purpose, *e.g* propagating the walker wavefunction from both A and B to an intermediate node. Under the assumption of quantum error correction, this backward propagation can occur simultaneously with any further quantum operation that nodes A and B may perform on the qubits and does not impact the latency of the protocol.

318

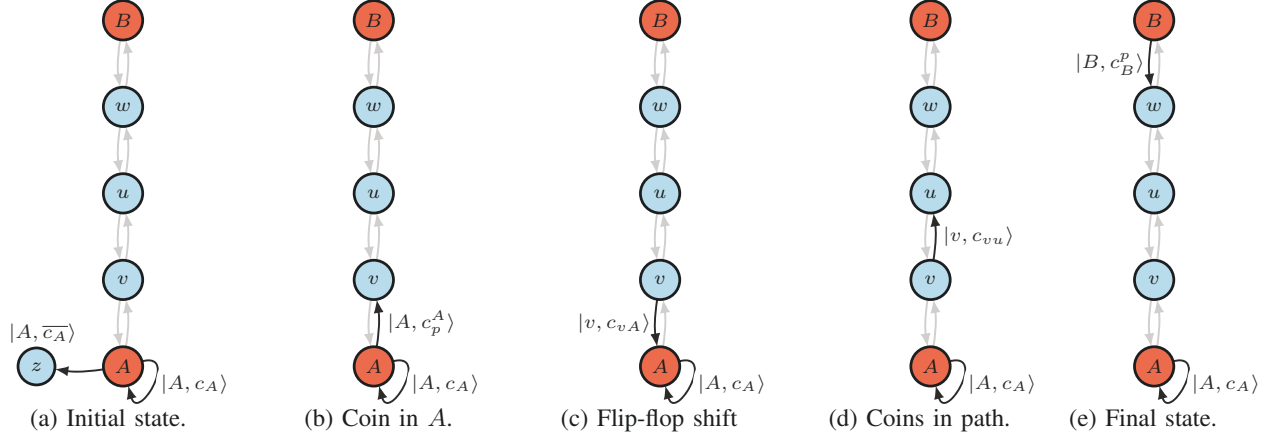| (a) Initial state. | (b) Coin in A. | (c) Flip-flop shift | (d) Coins in path. | (e) Final state. |

Fig. 2: Protocol execution in a 5-by-5 grid with a quantum walk through a single path. Dark edges depict vectors which have non-zero wavefunction component in a given step. (a) The initial state of execution is the state generated by the application of the controlled operation demonstrated in (13). The node z is shown for illustration purposes and is not for propagation. (b) The first coin flip permutes the wavefunction into edges (A, A) and (A, v). (c) The flip-flop shift exchanges edge (A, u) with edge (u, A), moving the walker while mapping the self-loop edge to itself. (d) After the first coin flip, all subsequent coin operators work as shift operators inside a node, mapping degrees of freedom in order to propel the walker towards B. (e) After $\Delta(A, B)$ steps, the final wavefunction is a uniform superposition between edges (A, A) and (B, w), which can be used to perform an operation controlled by qubit a located in A with target qubit b located in B.

IV. PROPAGATION OF MULTIPLE CONTROL SIGNALS

We demonstrated in the previous section how the quantum walk can propagate through a particular path of the network connecting nodes A and B. This is easily extended to the traversal of multiple paths as we will show next. In this setting, k walkers can be used to simultaneously perform operations controlled by a with k target qubits b_j located in nodes B_j, for $j \in \{0, \ldots, k-1\}$. This protocol execution translates to the parallel application of $k + 1$ 1-qubit gates controlled by a in terms of distributed quantum computation.

The analysis carried for a single walker system extends to this case by considering a set of paths $\mathcal{P}$ such that, for every j, the path $p_j \in \mathcal{P}$ connects A and B_j. The path p_j defines extended coin operators for the j-th walker following (18). By carefully choosing the qubits controlled by each walker, so that each walker interacts with a unique set of qubits, it is possible to route $k = |\mathcal{P}|$ walkers concurrently. This is a trivial extension to the single walker case because the separability of individual coins for each walker yield the k-walker coin operator for the set of paths $\mathcal{P}$ to be of the form

$$C^{\mathcal{P}}(t) = \bigotimes_{j=0}^{k-1} C^{p_j}(t). \tag{21}$$

Since the memories affected by walker j are unique, for every $j \in \{0, \ldots, T-1\}$, each walker can be seen as a parallel control signal working on distinct qubits. Note that, even though the walker evolution is performed by separable extended coin operators and shifts, the application of the controlled operation O described in (10) before propagation implies that the walkers are maximally entangled with each other and with qubit a. Furthermore, the flip-flop shift operator for each walker does not depend on a particular path and the joint operator for all walkers is described by

$$S^{\mathcal{P}} = \bigotimes_{j=0}^{k-1} S, \tag{22}$$

with S given in (17).

In addition to simultaneous 2-qubit control operations, it is trivial to use multiple walkers departing from A to perform controlled operations in the nodes B_j, where the target qubit b_j in B_j is controlled by qubit a_j in A. The fundamental difference for this case is that the O operator described in (13) is controlled by qubit a_j when applied to the j-th walker, generating a separable state among the walkers.

319

V. Quantum Walk Propagation and Entanglement Distribution

As described previously, the operators defined in (7) and (10) allow for universal distributed quantum computation among nodes of the network. In this sense, the movement of the walker introduces entanglement across nodes. Given this is the case, it is suitable to consider entanglement distribution protocols in terms of the quantum walk control plane. In this section, we apply the quantum walk control protocol to recover the behavior of two entanglement distribution protocols. In particular, the recovered protocols distribute maximally entangled Greenberger-Horne-Zeillinger (GHZ) states across network nodes. We demonstrate, for both protocols, how the entanglement introduced by the walker can be used to produce the same entangled states.

Superposition of the walker wavefunction allows for the creation of a Bell state between a and b as follows. Assume that the joint state of the system is at time t described by

$$|\Psi(t)\rangle = \frac{1}{\sqrt{2}}(|A, c_A, 0_a, 0_b\rangle + |B, c_B, 0_a, 0_b\rangle), \quad (23)$$

which is a separable state between the walker and the qubits a and b. By setting $U_V = X_v \otimes I$ and $C_V(t) = I$ for $V \in \{A, B\}$ in (7), the extended coin operator $C(t)$ is

$$C(t)|\Psi(t)\rangle = \frac{1}{\sqrt{2}}(|A, c_A, 1_a, 0_b\rangle + |B, c_B, 0_a, 1_b\rangle), \quad (24)$$

which is an entangled state between a and b. The entanglement between a, b and the walker system can be removed by transmitting the walker back to A as explained in Section III-D. Note that only operators of the form prescribed in (7) are needed because there is no control dependent on the state of either a or b.

A GHZ generalizes Bell states to multiple qubits. A k-qubit GHZ projection is a von-Neumann measurement operation in the GHZ basis and a Bell state measurement (BSM) is a 2-qubit GHZ projection.

A. Multi-path entanglement distribution with BSMs

We start with the multi-path protocol defined in [25]. Assuming global link state knowledge, it suffices to choose the same paths that a complete round of the protocol uses to propagate entanglement. Consider that, for such round, the set of paths $P = \{p_0, \ldots, p_{k-1}\}$ is used. In this protocol, every path p_j used to distribute entanglement yields a set of $2(|p_j|-1)$ qubits in a GHZ state, which comes from the net effect of performing $|p_j| - 2$ BSMs in the intermediate nodes. Moreover,

the protocol assumes a network model where every quantum channel in the network is mapped in a 1-to-1 fashion with pairs of qubits in the nodes incident to that channel. The logical behavior of this protocol is recovered by sending one quantum walk from A through each path p_j such that, once it arrives at B, every qubit incident to the quantum channels in the path are in the same GHZ state. Thus, consider that all qubits in the network start in the $|0\rangle$ state. Initialize k walkers in state $|A, c_A\rangle$. The operator $C_A^j(0)$ for the j-th walker can be chosen as any unitary that maps

$$|A, c_A\rangle \rightarrow \frac{1}{\sqrt{2}}(|A, c_A\rangle + |A, c_A^{p_j}\rangle), \quad (25)$$

while the interaction operator $K_A^j(0)$ is taken to be the identity operator. The state of the system after the application of the extended operator $C(0)$ is

$$C(0)|\Psi(0)\rangle = \bigotimes_j \frac{1}{\sqrt{2}}(|A, c_A\rangle + |A, c_A^{p_j}\rangle). \quad (26)$$

Protocol execution is performed by the operators defined in (22) and (21) as specified in the previous section, although the operators K_v are now defined to generate entanglement in the nodes as the walker passes by. Let q_{vj0} and q_{vj1} denote the pair of qubits inside node $v \in p_j \setminus \{A, B\}$ that need to be entangled together by walker j. The operator U_v^j in equation (8) is defined for walker j by specifying

$$K_v(t) = X_{qj0} \otimes X_{qj1} \bigotimes_{q \neq q_{vj0}, q_{vj1}} I_q \quad (27)$$

for all t, performing an X gate on qubits q_{v0}^j and q_{v1}^j controlled by the position of the j-th walker. Since paths are all edge disjoint, every qubit in the graph spanned by $\mathcal{P}$ interacts with exactly one walker under the application of the generalized coin operator. For walker j, the interaction operator with qubit a_j is given by

$$K_A(t) = \begin{cases} X_{a_j} \otimes I, & \text{when } t = 1, \\ I_{a_j} \otimes I, & \text{when } t \neq 0, \end{cases} \quad (28)$$

where the identity operator after the tensor product symbol acts on all qubits in A except a_j. Considering that u_j is the first neighbor of A in the path p_j, the application of the evolution operator $S(0)C(0)$ gives

$$|\Psi(1)\rangle = \bigotimes_j \frac{1}{\sqrt{2}}(|A, c_A\rangle + |u_j, c_{u_j A}\rangle), \quad (29)$$

where the data qubits are omitted for simplicity. $S(1)C(1)$ generates an entangled state between a_j and the qubits q_{uj0} and q_{uj1} and moves the walker to

320

the neighbors that are 2 hops away from A. Every application of $S(t)C(t)$ for $t > 1$ increases by two the number of qubits entangled with walker j such that, when $t = \Delta(A, B)$, the walker j is in an entangled state with the qubits in p_j, a_j and b_j. Figure 3(a) depicts protocol execution for a 5-by-5 grid network. In this case, P is formed by two shortest-paths in the grid, i.e $K = 2$, and the entire evolution of the walkers takes 4 steps.

B. Distributing GHZ states across network nodes

GHZ projectors are used in [26] to improve the multi-path BSM protocol described in [25]. The logical behavior of the GHZ distribution protocol can be recovered in the quantum walk protocol by slightly modifying the description presented for the multi-path BSM protocol. In this protocol, every qubit in a node used to distribute entanglement is part of the final GHZ state achieved. This is in clear contrast with the BSM multi-path protocol, where each path defines an independent GHZ state. The first modification is that, instead of preparing separable walkers with coins that maps states following the form given in (25), the k walkers need to be entangled in the state

$$\frac{1}{\sqrt{2}}(|A, c_A, \ldots, A, c_A\rangle + |A, c_A^{p_0}, \ldots, A, c_A^{p_j}\rangle). \quad (30)$$

Note that this behavior can be obtained with an interaction operator of the form (10) and a coin operator in the form specified by (18). Secondly, instead of routing walkers through edge disjoint paths, the walkers traverse a tree of the network. More concretely, let $\mathcal{T}$ be a spanning tree of the lattice, rooted at A. k is now the number of leaves in $\mathcal{T}$ and $\mathcal{P}$ is the set of paths in the tree connecting A with leaves. For all vertices v, the interaction operator between the walker and the data qubits is

$$K_v = \bigotimes_{q_v} X_{qv}^{\frac{1}{W_v}}, \quad (31)$$

where W_v is the number of walkers that pass through v, i.e the number of leaves connected to A by v in the tree. Note that K_v is the same for every walker. The exponent $\frac{1}{W_v}$ is required because more than one walker may pass through the same vertex, and operate on the same set of qubits. Since paths are taken from a spanning tree, each walker can take a particular time interval to traverse its path. Thus, the time necessary to entangle all of the necessary qubit in the network and remove the entanglement between the walkers and the data qubits is on the order of $\mathcal{O}(\max_j |p_j|)$, where $|p_j|$ denotes the hop distance of path p_j. Again,

(a) Multi-path propagation.
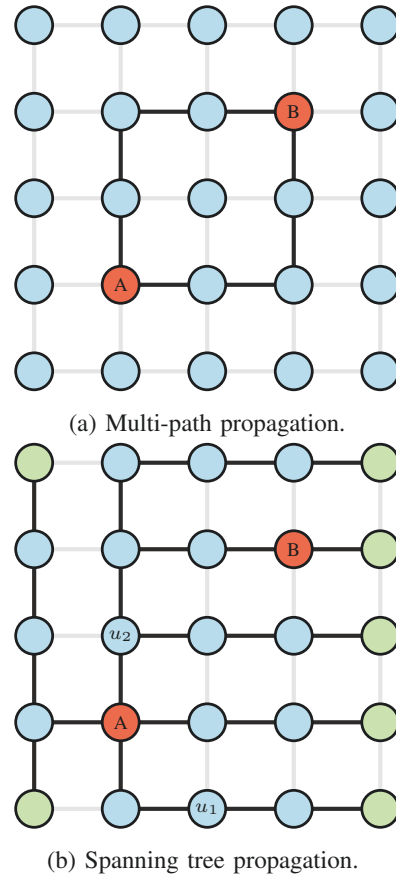

(b) Spanning tree propagation.

Fig. 3: Recovering protocol behavior with quantum walks. Protocols that rely on GHZ projections apply entanglement swapping to distribute entanglement between A and B. Entanglement swapping is non-unitary, although the overall effect of applying a GHZ projection is a maximally entangled state between qubits. The quantum walk approach recovers this behavior by generating GHZ states unitarily. (a) A multi-path BSM protocol is recovered by sending separable walkers into each edge-disjoint path used in a round of the protocol, denoted in the figure by shaded edges. Each path forms an independent GHZ state between all qubits in the path. (b) The GHZ projection protocol is recovered by sending entangled quantum walks to each leaf l_i of a spanning tree of the network rooted in A, represented in the figure by green vertices. All qubits in the network are, in this case, in a GHZ state.

protocol execution for the generic GHZ projector case is exemplified for a 5-by-5 grid in Figure 3(b). In this case, $k = 7$ and the whole process takes 7 time steps to complete, since 7 is the size of the largest path in the tree. For nodes u_1 and u_2, the value of W is 1 and 3 respectively.

VI. CONCLUSION

The quantum walk protocol proposed in this article provides a logical description for a network control plane capable of performing universal distributed quantum computing. The description abstracts the implementation of the quantum walker system, as well as the implementation of quantum operations in the network nodes. It considers that quantum error correction yields the application of perfect unitary operators. The key idea that the protocol builds upon is the use of a quantum walker system as a quantum control signal that propagates through the network one hop a time. In spite of abstracting physical implementations, the propagation of the walker stipulates latency constraints for the protocol. A generic controlled operation between a qubit in node A with a qubit in node B demands $\mathcal{O}(\Delta(A, B))$ steps of walker evolution. In the context of a possible physical realization of such control system, this latency constraints translates directly to the physical distance between nodes A and B. The description of the protocol in the logical setting also masks the effects of walker propagation in the fidelity of distributed operations. When considering imperfect operators, the fidelity of the final outcome is bounded by the fidelity of the coin and shift operators in the quantum walk system. Throughout this manuscript, the propagation of a quantum walk across the network depended on a particular path of the network to be traversed. This has the clear requirement that every node in the network knows the network topology and that the operators used can be defined by the exchange of classical messages between the nodes. The protocol described was used to represent entanglement distribution protocols defined in the literature in terms of quantum control information exchanged between the nodes in the network. This result highlights connections between the proposed protocol and entanglement distribution protocols.

There are two clear directions for future work considering our results. The first relates to the investigation of a network implementation for a quantum walk system. Such implementation would allow for a realistic characterization of quantities like fidelity and latency. The second point is the description of control exclusively with quantum information. In this setting, nodes would transmit a quantum state containing both the state of the walker and the target node to which control information must be transmitted to.

REFERENCES

[1] H Jeff Kimble. The quantum internet. *Nature*, 453(7198):1023–1030, 2008.

[2] Stephanie Wehner, David Elkouss, and Ronald Hanson. Quantum internet: A vision for the road ahead. *Science*, 362(6412), 2018.

[3] Robert Beals, Stephen Brierley, Oliver Gray, Aram W Harrow, Samuel Kutin, Noah Linden, Dan Shepherd, and Mark Stather. Efficient distributed quantum computing. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 469(2153):20120686, 2013.

[4] Angela Sara Cacciapuoti, Marcello Caleffi, Francesco Tafuri, Francesco Saverio Cataliotti, Stefano Gherardini, and Giuseppe Bianchi. Quantum internet: networking challenges in distributed quantum computing. *IEEE Network*, 34(1):137–143, 2019.

[5] Laszlo Gyongyosi and Sandor Imre. Scalable distributed gate-model quantum computers. *Scientific reports*, 11(1):1–28, 2021.

[6] Rodney Van Meter and Simon J Devitt. The path to scalable distributed quantum computing. *Computer*, 49(9):31–42, 2016.

[7] Stefano Pirandola, Riccardo Laurenza, Carlo Ottaviani, and Leonardo Banchi. Fundamental limits of repeaterless quantum communications. *Nature communications*, 8(1):1–15, 2017.

[8] Kevin S Chou, Jacob Z Blumoff, Christopher S Wang, Philip C Reinhold, Christopher J Axline, Yvonne Y Gao, Luigi Frunzio, MH Devoret, Liang Jiang, and RJ Schoelkopf. Deterministic teleportation of a quantum gate between two logical qubits. *Nature*, 561(7723):368–373, 2018.

[9] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[10] David Kielpinski, Chris Monroe, and David J Wineland. Architecture for a large-scale ion-trap quantum computer. *Nature*, 417(6890):709–711, 2002.

[11] JM Pino, JM Dreiling, C Figgatt, JP Gaebler, SA Moses, MS Allman, CH Baldwin, M Foss-Feig, D Hayes, K Mayer, et al. Demonstration of the trapped-ion quantum ccd computer architecture. *Nature*, 592(7853):209–213, 2021.

[12] Pieter-Jan Stas, Bartholomeus Machielse, David Levonian, Ralf Riedinger, Mihir Bhaskar, Can Knaut, Erik Knall, Daniel Assumpcao, Rivka Bekenstein, Yan Qi Huan, et al. High-fidelity quantum memory for the silicon-vacancy defect in diamond. *Bulletin of the American Physical Society*, 2021.

[13] Jun-Feng Wang, Fei-Fei Yan, Qiang Li, Zheng-Hao Liu, Jin-Ming Cui, Zhao-Di Liu, Adam Gali, Jin-Shi Xu, Chuan-Feng Li, and Guang-Can Guo. Robust coherent control of solid-state spin qubits using anti-stokes excitation. *Nature Communications*, 12(1):1–9, 2021.

[14] David Awschalom, Karl K Berggren, Hannes Bernien, Sunil Bhave, Lincoln D Carr, Paul Davids, Sophia E Economou, Dirk Englund, Andrei Faraon, Martin Fejer, et al. Development of quantum interconnects (quics) for next-generation information technologies. *PRX Quantum*, 2(1):017002, 2021.

[15] Dorit Aharonov, Andris Ambainis, Julia Kempe, and Umesh Vazirani. Quantum walks on graphs. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 50–59, 2001.

[16] Andrew M Childs. Universal computation by quantum walk. *Physical review letters*, 102(18):180501, 2009.

[17] Andrew M. Childs, David Gosset, and Zak Webb. Universal computation by multiparticle quantum walk. *Science*, 339(6121):791–794, 2013.

[18] Neil B Lovett, Sally Cooper, Matthew Everitt, Matthew Trevers, and Viv Kendon. Universal quantum computation using the discrete-time quantum walk. *Physical Review A*, 81(4):042330, 2010.

[19] Thomas Nitsche, Fabian Elster, Jaroslav Novotný, Aurél Gábris, Igor Jex, Sonja Barkhofen, and Christine Silberhorn. Quantum walks with dynamical control: graph engineering, initial state preparation and state transfer. *New Journal of Physics*, 18(6):063017, jun 2016.

[20] Xiang Zhan, Hao Qin, Zhi-hao Bian, Jian Li, and Peng Xue. Perfect state transfer and efficient quantum routing: A discrete-time quantum-walk approach. *Physical Review A*, 90(1):012331, 2014.

[21] Heng-Ji Li, Xiu-Bo Chen, Ya-Lan Wang, Yan-Yan Hou, and Jian Li. A new kind of flexible quantum teleportation of an arbitrary multi-qubit state by multi-walker quantum walks. *Quantum Information Processing*, 18(9):266, 2019.

[22] Yu Wang, Yun Shang, and Peng Xue. Generalized teleportation by quantum walks. *Quantum Information Processing*, 16(9):221, 2017.

[23] Chrysoula Vlachou, Walter Krawec, Paulo Mateus, Nikola Paunković, and André Souto. Quantum key distribution with quantum walks. *Quantum Information Processing*, 17(11):1–37, 2018.

[24] Meng Li and Yun Shang. Entangled state generation via quantum walks with multiple coins. *arXiv preprint arXiv:2011.01643*, 2020.

[25] Mihir Pant, Hari Krovi, Don Towsley, Leandros Tassiulas, Liang Jiang, Prithwish Basu, Dirk Englund, and Saikat Guha. Routing entanglement in the quantum internet. *npj Quantum Information*, 5(1):1–9, 2019.

[26] Ashlesha Patil, Mihir Pant, Dirk Englund, Don Towsley, and Saikat Guha. Entanglement generation in a quantum network at distance-independent rate, 2020.

[27] Stefano Pirandola. End-to-end capacities of a quantum communication network. *Communications Physics*, 2(1):1–10, 2019.

[28] Emanuel Knill and Raymond Laflamme. Theory of quantum error-correcting codes. *Phys. Rev. A*, 55:900–911, Feb 1997.

[29] Joschka Roffe. Quantum error correction: an introductory guide. *Contemporary Physics*, 60(3):226–245, 2019.