

Distribution and Segmentation Methods of Graph States Based on Quantum Networks

Haomin CHEN*, Tianze ZHU*, Guanting YU*, Yongzhe LIN*, Xiaofeng JIANG*

* Department of Automation, University of Science and Technology of China, Hefei China
 chen0215@mail.ustc.edu.cn, tz_19@mail.ustc.edu.cn, ygt991129@mail.ustc.edu.cn,
 lyz@mail.ustc.edu.cn, jxf@ustc.edu

Abstract—In recent years, quantum communication and quantum computing have become strategic technologies of great importance to countries around the world, with profound implications for individuals, businesses, and even nations. Among these, graph states are a very important class of multi-particle entangled states, playing a crucial role in fields such as quantum secret sharing, measurement-based quantum computing, and quantum metrology. However, the distribution of graph states faces the following challenges: existing quantum communication networks designed for single-qubit transmission cannot effectively accomplish graph state distribution tasks, and current methods for graph state distribution lack consideration of node resources, making it difficult to complete graph state distribution in some cases.

We conduct research on cross-node graph states distribution methods in quantum communication networks under limited resource constraints. To address the issue that current quantum communication networks cannot handle graph states and other multi-particle entangled states, a graph state distribution method is proposed. This method combines local operations and classical communication to meet the distribution requirements for graph states, which are multi-particle entangled states. Additionally, to solve the problem of limited quantum resources at individual nodes, we proposed a graph states segmentation method. This method divides a complete graph state into subgraphs for distribution and subsequently integrates them, enabling graph state distribution services in a broader range of resource scenarios.

Key Words—Quantum Communication, Graph States, Entanglement Distribution, Quantum Network, Graph Theory.

I. INTRODUCTION

Graph states [1] are a special class of multi-particle entangled quantum states, represented in the form of a graph to illustrate the entanglement relationships among qubits. Both GHZ states [2] and cluster states [3] are types of graph states. Networks constructed using these more complex entangled states at various nodes can achieve unique functionalities compared to networks that use Bell states for quantum communication, such as quantum secret sharing [4], [5], measurement-based quantum computing [6], [7], quantum error correction coding [8], [9], and quantum multi-party secure computation [10]. Thus, they are a very important quantum resource. However, due to the complex entanglement relationships, operations on graph states require more sophisticated quantum gates. Therefore, effectively preparing and distributing graph states is a primary challenge in leveraging this important quantum resource.

There has been many works about the construction of graph states on a single node, Lu Chaoyang et al. [11] successfully prepared a 6-qubit GHZ state using a polarization beam splitter. Antonio Russo et al. [12] proposed a protocol for generating graph states using diamond NV centers and self-assembled quantum dots. Philip Thomas et al. [13] utilized a cavity quantum electrodynamics platform as a photon source to produce large-scale, high-fidelity graph states, successfully preparing GHZ states with up to 14 photonic qubits and linear cluster states with 12 photonic qubits, achieving better results than conventional spontaneous parametric down-conversion (SPDC) as a photon source. The above works primarily focus on preparing graph states in physical systems.

In the application of graph states, Adrian Keet et al. [14] proposed using graph states for quantum secret sharing. B.A. Bell et al. [15] achieved quantum secret sharing between one sender and four receivers using photons. Massimiliano Proietti et al. [16] introduced an MBQC algorithm to solve the maximum K-cut problem. Reimer et al. [17] employed high-dimensional cluster states (qudits) to achieve one-way quantum computation, achieving higher efficiency and reducing sensitivity to system noise with the same number of particles.

Consider distributing graph states over multi-nodes, M. Cuquet et al. [18] proposed a method for preparing GHZ states among multiple nodes, followed by entanglement purification to merge them into complete graph states. Clément Meignant et al. [19] introduced a method for distributing arbitrary graph states among remote nodes: constructing a complete graph state using GHZ states and then removing excess edges through graph state operations. Alex Fischer et al. [20] proposed an alternative method: using one of the nodes to locally construct a copy of the arbitrary graph state to be distributed, and then using quantum teleportation to transfer the solid-state copy to the corresponding node. These above works are based on the assumption of unlimited node qubit resources, which has significant limitations in practical quantum communication network environments.

In this work, we investigate a segmentation-integration method for graph state distribution under limited node resources, aiming to achieve efficient utilization of qubit resources. Considering that distributing a complete graph state in one go would impose significant resource pressure on individual nodes, we adopt the idea of segmentation and integration. The approach involves first distributing partial

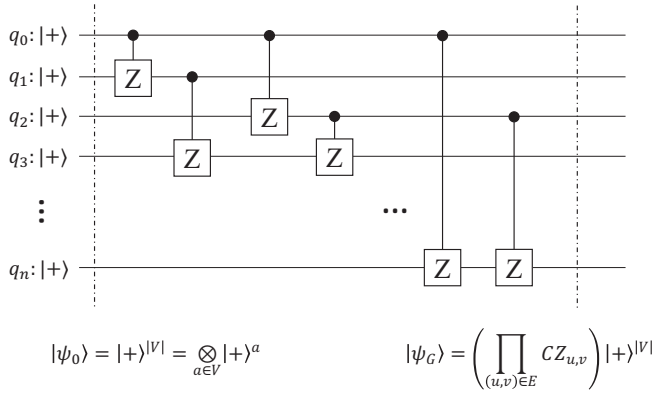


Figure 1: Quantum circuit of creating a graph state.

subgraph states and then recovering the desired complete graph state through operations related to graph states. The method for segmenting graph states will take into account factors such as the resource availability at each node, the expected transmission quality, and the additional quantum resource consumption for integrating the subgraph states, providing a way to divide the complete graph state into several subgraph states.

With our method, cross-node distribution of arbitrary graph states can be achieved. Moreover, by segmenting larger graph states, we can achieve a better distribution success rate even when the quantum bit resources at the nodes are limited. After segmentation, when using graph state copies for quantum teleportation to distribute the subgraph states, the smaller scale of the graph states allows the entangled pairs used for teleportation to be prepared with fewer entanglement swapping operations, thereby improving the fidelity of graph state distribution.

II. PRELIMINARIES

A. Graph States

Graph states are a general term for a class of multi-particle entangled states. They are called graph states because their entanglement relationships can be represented using a graph $G = (V, E)$. We denote graph states as $|\psi_G\rangle$, both GHZ states and cluster states are examples of graph states.

To prepare graph states, a special type of single-qubit state $|+\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ is required. For example, to create a 5-qubit graph state, we need to create 5 states of $|+\rangle$. At this time, these 5 qubits do not yet have entanglement relationships.

To get these 5 qubits entangled, we need to perform controlled Z gate operation on all pairs of vertices belonging to E in graph G .

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, CZ = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad (1)$$

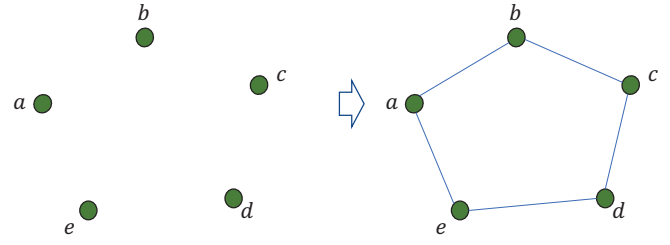
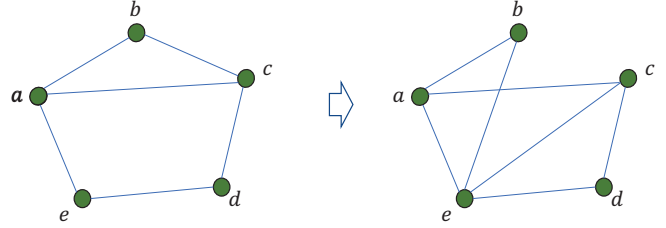


Figure 2: Create a graph state.

Figure 3: Local complementation on node a

$$|\psi_G\rangle = \left(\prod_{(u,v) \in E} CZ_{(u,v)} \right) |+\rangle^{|V|} \quad (2)$$

$$|+\rangle^{|V|} = \bigotimes_{a \in V} |+\rangle^a \quad (3)$$

In the example of Fig. 2, we need to perform CZ gate operation between (a, b) , (b, c) , (c, d) , (d, e) and (e, a) . $\otimes$ means tensor product.

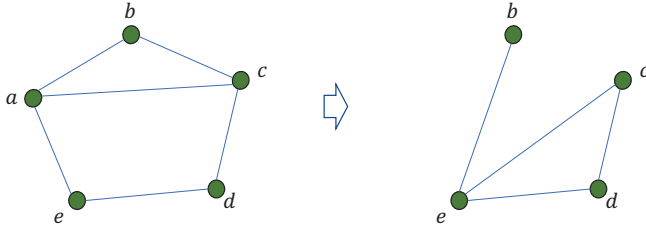
Here are some other graph states operations which can affect graph states graphically:

- **Delete/Add Edges:** perform CZ gate operation on the edge (u, v) . if it exists, this operation will delete it, or this operation will add an edge at this position.
- **Local Complementation:** perform an operator $U_a^{LC} = \sqrt{-iX_a} \otimes \sqrt{iZ_{Na}}$ to a vertex $a \in V$, we can get the structure of graph states $G' = G \cup G_{Na} \bmod 2$. G_{Na} , the complete graph of the neighbouring vertices of vertex a . Above means original graph G is superimposed with G_{Na} , if there are overlapping edges, they are deleted; if not overlapping, they are retained.
- **Z-measurement:** perform Z-measurement to vertex a , can delete vertex a and its incident edges.
- **Y-measurement:** perform Y-measurement to vertex a , is equivalent to perform Local Complementation to vertex a , and then perform Z-measurement to vertex a .

We will use some of these graph states operation to achieve the cross-nodes distribution of graph states.

B. Quantum Network

A quantum network is a set of nodes and edges $G_{net} = (V_{net}, E_{net})$. As Fig.5 shows, a node consists of a finite number of quantum storage qubits and a quantum processor.

Figure 4: Y-measurement on node a .

Adjacent nodes are connected by classical and quantum channels, allowing for the free exchange of classical information, while quantum information is transmitted with loss over the channels.

We set up the quantum network in a grid topology, where qubits within the nodes can be allocated to channels in any direction. At any given time, each qubit can only be assigned to one channel. Each node can serve as both the sender and receiver for quantum teleportation, as well as a repeater node for entanglement swapping.

III. GRAPH STATES DISTRIBUTION

For a graph state distribution task, a request will be sent with the required graph structure G and the vertices of graph G corresponding to the locations in the quantum communication network. After receiving the request, we need to generate and distribute the corresponding graph state based on the request, ultimately returning a graph state $|\psi_G\rangle$ as the output result of the system.

To distribute graph states between different nodes in a quantum communication network, we create a copy of the graph state to be distributed at a certain node. By using entanglement swapping technology, we share entangled pairs over long distances, and utilize quantum teleportation to send the qubits of the graph state to the corresponding nodes.

As Fig.5 and fig.6 shows, we need to distribute a GHZ state over nodes A, B and C , so we create a copy on node A . Then we need to share entangled pairs between (A, C) and (A, B) , so we distribute entangled pairs along the path, and execute entanglement swapping on repeater nodes, then we can share entangled pairs between the node preparing the copy and the node where the graph state is to be distributed. Finally, by performing quantum teleportation at the graph state copy, the graph state can be distributed to various nodes in the quantum network.

IV. GRAPH SEGMENTATION

Since quantum resources are often very limited in quantum communication networks, a single node must provide both the quantum bits for preparing copies and those for quantum teleportation. This can lead to resource constraints that make it difficult for nodes to meet graph state distribution requests. Therefore, we adopt the approach of dividing the complete graph G into several subgraph G_i , distributing the subgraph states $|\psi_{G_i}\rangle$ separately, and then using graph state operations

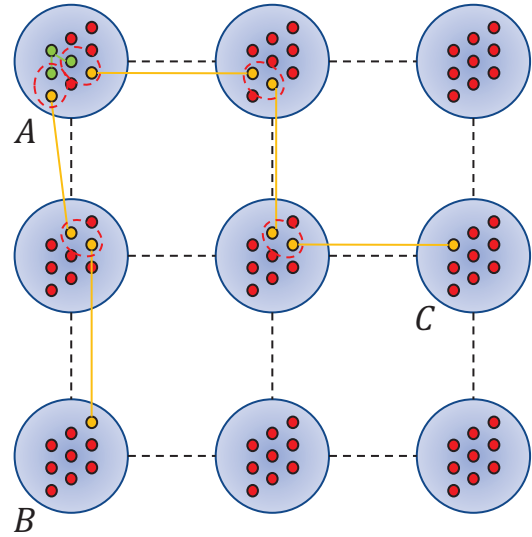


Figure 5: Distribute graph state use copy. Green qubits represent graph states copy to be distributed, yellow qubits represent entangle pairs, red circles represent entanglement swapping or quantum teleportation.

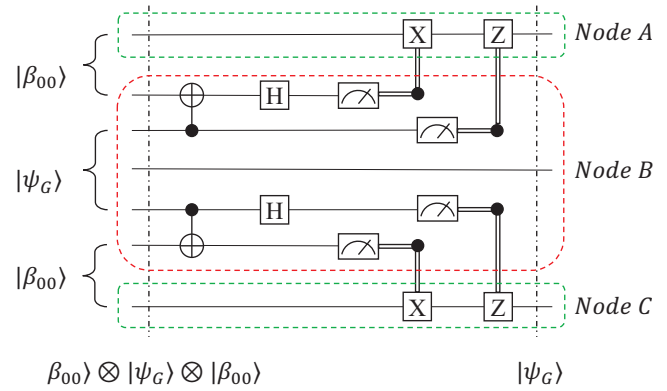


Figure 6: Quantum circuit of graph states distribution.

to integrate these subgraph states into the requested graph state $|\psi_G\rangle$.

A. Add weight to edge

To get a proper segmentation of a graph, we need to consider many factors that affect the success rate of graph states distribution and the distribution quality of graph states. For each edge $e_i \in E$, we will give it a weight $W(e_i)$, this weight will show the cost of distributing graph states. The parameters involved here are:

- C_{e_i} , available bandwidth of link e_i : The minimum number of available qubits among the two nodes belonging to link e_i . The smaller the bandwidth, the greater the cost of distributing the graph state. Therefore, this metric uses an inverse function, specifically, use $\frac{1}{C_{e_i}}$ as a metric.
- f_{e_i} , entanglement fidelity of link e_i : Since higher fidelity corresponds to better quality of graph state distribution,

and the composite fidelity of multi-hop links can be approximated using a product, this metric takes the negative logarithm of the fidelity, specifically, use $-\ln(f_{e_i})$ as a metric.

- d_{e_i} , classical time delay of link e_i : The delay for classical communication over the link. The lower the delay, the higher the efficiency of graph state distribution. Therefore, this metric is based on the communication delay, specifically, use d_{e_i} as a metric.

We can compute the weight of e_i based on parameters above:

$$W(e_i) = -\ln(f_{e_i}) \cdot k_1 + d_{e_i} \cdot k_2 + \frac{1}{C_{e_i}} \cdot k_3 \quad (4)$$

k_i is undetermined coefficient, we can adjust it by requirement.

After traversing all $e_i \in E$, we can get the weight matrix W , turning the original graph $G = (V, E)$ to a weighted undirected graph $G = (V, E, W)$.

Normalized Cut is a technique used for graph segmentation, widely applied in fields such as image processing and computer vision. Its main idea is to minimize the cut cost of the graph while considering the size of each subset, in order to achieve a more reasonable segmentation result. First, calculate the degree matrix D and the weight matrix W , then compute the graph's Laplacian matrix $L = D - W$. Then calculate the eigenvectors of the Laplacian matrix and use these eigenvectors for K-means clustering to segment the graph into different regions. Calculate each region's normalized cut value:

$$NCut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, B)}{assoc(B, V)} \quad (5)$$

$cut(A, B)$ represents the sum of the weights of the edges between region A and region B , V denotes the intersection of A and B , and $assoc(A, V)$ is the sum of the weights from all nodes in A to V .

Select the segmentation with the minimum normalized cut value as the result. After obtaining the segmentation, calculate the maximum bit consumption per node $\max(Cost_{node})$, the estimated time required for the distribution operation $T_{estimate}$ and the expected number of quantum gates N_{gate} used for distributing the graph state. Ensure that the nodes have sufficient resources to complete the distribution and that the operation time does not exceed the quantum bit decoherence time. Otherwise, re-segment the graph state using the aforementioned method.

B. Integration of subgraph states

After getting several subgraph states, we need to do some graph states operations to integrate these states into a complete graph state.

$$|\psi_0\rangle = |\psi_{G_1}\rangle \otimes |\beta_{00}\rangle \otimes |\psi_{G_2}\rangle \quad (6)$$

First, between the two nodes that need to establish a connection for the graph state, a pair of Bell state entangled pairs must be distributed between the nodes. Then, use CZ gates to connect each subgraph state with the entangled pairs,

Algorithm 1 Normalized Cut

- 1: **Input:** Weighted graph $G = (V, E, W)$
- 2: **Output:** Segmented subsets A and B
- 3: **Steps:**
- 4: Compute the degree matrix D , where D_{ii} is the sum of weights of edges connected to vertex i .
- 5: Calculate the Laplacian matrix $L = D - W$.
- 6: Perform eigenvalue decomposition on the Laplacian matrix L .
- 7: Select the top k eigenvectors from the decomposition.
- 8: Apply K-means clustering on the selected eigenvectors to segment the graph.
- 9: For each segmentation, compute the normalized cut value $NCut(A, B)$.
- 10: Select the segmentation with the minimum normalized cut value as the final result.

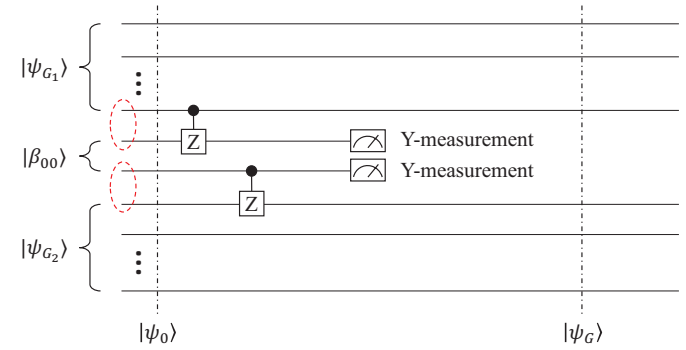


Figure 7: Quantum circuit of integration. Red circle represents these qubits are in the same node.

forming a larger graph state. Finally, perform Y-measurements on the qubits at the positions that originally belonged to the entangled pairs, which will integrate the subgraph states into a bigger one, the effect on the graph structure is to connect two subgraphs with an edge, which is equivalent to implementing a cross-node CZ gate.

Fig.7 illustrates the process of connecting two graph states at a location where subgraph state integration is required. During the actual integration process, there may be more subgraph states that need to be integrated, and there could be additional connection points between the subgraph states. By performing operations on the graph at all the required connection points, we can ultimately obtain the graph state $|\psi_G\rangle$ needed for the task.

V. EXPERIMENT

A. Experiment setup

The experiment was set up on the Python platform using the NetSquid library related to quantum networks. The network is a 10x10 grid quantum communication network, with the number of qubits within each node and the parameters of quantum channels between nodes initialized randomly. Randomly generated graph state requests are represented by

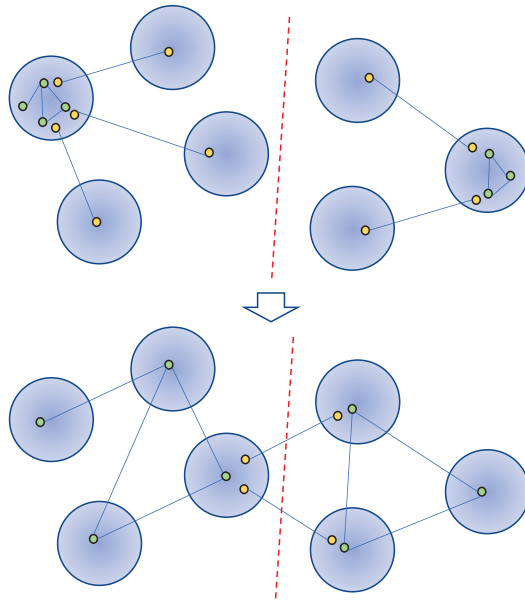


Figure 8: Segmentation and integration of a graph state. Green qubits represent subgraph states, yellow qubits represent entangle pairs.

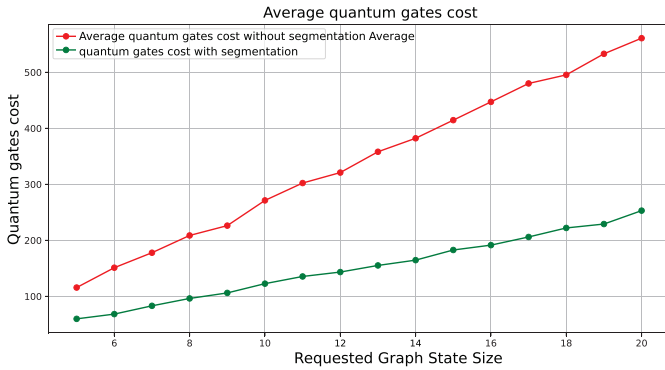


Figure 9: Average quantum gates cost

randomly connected graphs consisting of 5 to 20 nodes. After segmenting and integrating the graph states required for the requests, the number of quantum gates used during the distribution process is calculated (measurement operation is also counted as a quantum gate operation), and the qubit resource consumption at each node during the graph state distribution process is recorded.

B. Results and analysis

Fig.9 illustrates the number of quantum gates executed for directly preparing copies and using teleportation for graph state distribution, as well as for distributing and integrating graph states after segmenting. It can be observed that as the size of the requested graph state increases, the number of quantum gates executed also rises. However, the number of quantum gates used by our method is significantly lower than that of direct distribution. This is primarily because

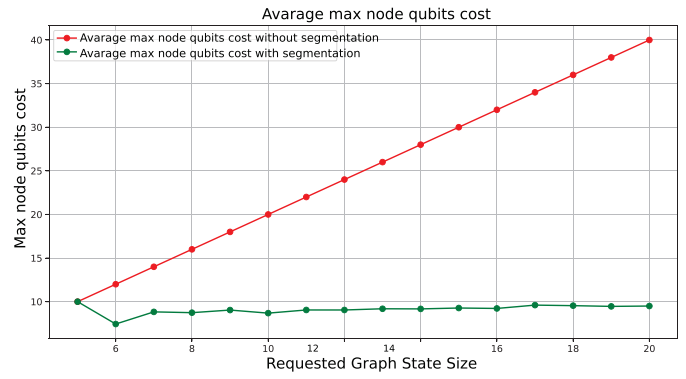


Figure 10: Average max node qubits cost

the number of entanglement swaps required after graph state partitioning is greatly reduced, and each entanglement swap operation typically requires two quantum gate operations. Since executing quantum gates is a lossy process, reducing the number of quantum gates can effectively diminish losses, thereby improving the fidelity of the graph states and the success rate of distribution.

Fig.10 represents the maximum single-node qubit consumption in two scenarios. The values are generally determined by the nodes preparing copies of the graph states. In addition to preparing the copies, these nodes must hold many entangled pairs for quantum teleportation, which places a significant resource burden on individual nodes. In our method, the size of the subgraph states can be controlled based on the partitioning conditions set. For example, Fig.9 controls the number of bits in the subgraph state to not exceed 5. Even as the size of the requested graph state gradually increases, the qubit resource consumption at individual nodes can be kept under control, thereby reducing the probability of congestion in practical quantum communication networks.

VI. CONCLUSIONS

In existing graph state distribution methods, there is a lack of consideration for node resources. However, in practical quantum communication networks, the limited availability of quantum resources within nodes is one of the key factors that restrict communication efficiency. Therefore, proposing a graph state distribution method that takes into account the limited quantum resources of nodes is essential for the practical application of graph states. We introduce a segmentation-integration method for graph state distribution, which segments large-scale graph states into several subgraph states for separate preparation and distribution. Finally, by performing graph state-related operations on some nodes, the subgraph states are restored to the desired complete graph state. This approach effectively reduces the quantum resources required at individual nodes when generating large-scale graph states, helping to maintain the smooth operation of the entire quantum communication network and preventing congestion due to any single node.

ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundations of China (Grant No. 62173315, 62341113, 62101525, 62201543, 62021001 and U23A20275), by the Youth Innovation Promotion Association CAS (Grant No. 2020450), by the China Postdoctoral Science Foundation (Grant No. 2023M733422), by Anhui Provincial Major Science and Technology Project (Grant No. 202103a05020024), by the Fundamental Research Funds for the Central Universities, and by the China Environment for Network Innovations (CENI).

REFERENCES

- [1] M. Hein, J. Eisert and H. J. Briegel, "Multiparty entanglement in graph states", *Physical Review A*, vol. 69, n. 20, pp. 1-20, 2004.
- [2] D. Bouwmeester, J.-W. Pan, M. Daniell, H. Weinfurter and A. Zeilinger, "Observation of three-photon Greenberger-Horne-Zeilinger entanglement", *Physical Review Letters*, vol. 82, no. 7, pp. 1345-1349, 1999.
- [3] H. J. Briegel and R. Raussendorf, "Persistent entanglement in arrays of interacting particles", *Physical Review Letters*, vol. 86, no. 5, pp. 910-913, 2001.
- [4] D. Markham and B. C. Sanders, "Graph states for quantum secret sharing", *Physical Review A*, vol. 78, no. 4, pp. 1-17, 2008.
- [5] Y. Ouyang, K. Goswami, J. Romero, et al., "Approximate reconstructability of quantum states and noisy quantum secret sharing schemes", *Physical Review A*, vol. 108, no. 1, pp. 1-5, Jul 2023.
- [6] R. Raussendorf and H. J. Briegel, "A One-Way Quantum Computer", *Physical Review Letters*, vol. 86, no. 22, pp. 5188-5191, May 2001.
- [7] P. Walther, K. J. Resch, T. Rudolph, et al., "Experimental one-way quantum computing", *Nature*, vol. 434, no. 7030, pp. 169-176, 2005.
- [8] B. A. Bell, D. A. Herrera-Martí, M. S. Tame, et al., "Experimental demonstration of a graph state quantum error-correction code", *Nature Communications*, vol. 5, no. 1, pp. 1-10, 2014.
- [9] M. Hein, W. Dür, J. Eisert, et al., "Entanglement in graph states and its applications", *Quantum computers, algorithms and chaos*, pp. 115-218, IOS Press, 2006.
- [10] Y. Lu and G. Ding, "Quantum secure multi-party summation with graph state", *Entropy*, vol. 26, no. 1, pp. 80, 2024.
- [11] C.-Y. Lu, X.-Q. Zhou, O. Gühne, et al., "Experimental entanglement of six photons in graph states", *Nature physics*, vol. 3, no. 2, pp. 91-95, 2007.
- [12] A. Russo, E. Barnes, and S. E. Economou, "Photonic graph state generation from quantum dots and color centers for quantum communications", *Physical Review B*, vol. 98, no. 8, pp. 1-12, 2018.
- [13] P. Thomas, L. Ruscio, O. Morin and G. Rempe, "Efficient generation of entangled multiphoton graph states from a single atom", *Nature*, vol. 608, no. 7924, pp. 677-681, 2022.
- [14] A. Keet, B. Fortescue, D. Markham and B. C. Sanders, "Quantum secret sharing with qudit graph states", *Physical Review A*, vol. 82, no. 6, pp. 1-11, 2010.
- [15] B. A. Bell, D. Markham, D. A. Herrera-Martí, et al., "Experimental demonstration of graph-state quantum secret sharing", *Nature communications*, vol. 5, no. 1, pp. 1-12, 2014.
- [16] M. Proietti, F. Cerecchi and M. Dispenza, "Native measurement-based quantum approximate optimization algorithm applied to the Max K-Cut problem", *Physical Review A*, vol. 106, no. 2, pp. 1-11, 2022.
- [17] C. Reimer, et al., "High-dimensional one-way quantum processing implemented on d-level cluster states", *Nature Physics*, vol. 15, no. 2, pp. 148-153, 2019.
- [18] M. Cuquet and J. Calsamiglia, "Growth of graph states in quantum networks", *Physical Review A*, vol. 86, no. 4, pp. 1-15, Oct. 2012.
- [19] C. Meignant, D. Markham and F. Grosshans, "Distributing graph states over arbitrary quantum networks", *Physical Review A*, vol. 100, no. 5, pp. 1-6, 2019.
- [20] A. Fischer and D. Towsley, "Distributing graph states across quantum networks", 2021 IEEE International Conference on Quantum Computing and Engineering (QCE), pp. 324-333, 2021.