

MCBGD: Multi-Path Cluster-Based Graph Distribution for Scalable Measurement-Based Quantum Computation

Guanting YU*, Yongzhe LIN**, Yiyang YU**, Zisen WEI**, Xiaofeng JIANG**

*Institute of Advanced Technology, University of Science and Technology of China, Hefei, China

**Department of Automation, University of Science and Technology of China, Hefei, China
(ygt991129, lyz, yuyiyang, wzs001)@mail.ustc.edu.cn, jxf@ustc.edu.cn

Abstract—This paper proposes Multi-path Cluster-Based Graph Distribution (MCBGD), a novel framework enabling scalable Measurement-Based Quantum Computation (MBQC) in resource-constrained quantum networks. To address the challenge of large-scale cluster state requirements in the NISQ era—where quantum computers typically handle only a few dozen qubits while MBQC demands hundreds or thousands—MCBGD introduces a hybrid decomposition strategy that separates cluster states into horizontal and vertical components. The framework features a genetic multi-path routing algorithm for optimized horizontal cluster deployment across qubit availability, channel fidelity, and latency constraints, combined with an entanglement-efficient vertical segmentation scheme. NetSquid simulations on grid networks demonstrate MCBGD’s significant advantages in both fidelity and resource efficiency compared to existing methods. The proposed approach provides a practical pathway for implementing quantum algorithms on distributed quantum processors with limited resources.

Keywords—Quantum Networks, Measurement-Based Quantum Computation, Graph State Distribution

I. INTRODUCTION

Measurement-Based Quantum Computation (MBQC) stands as a core model for quantum information processing, exhibiting significant universality and application potential[1], [2], [3], [4], [5], [6]. Unlike the commonly used quantum circuit model, MBQC operates on resource states known as cluster states or graph states. It relies solely on single-qubit measurements and subsequent byproduct corrections to achieve computational tasks equivalent to those performed by gate circuits. However, a significant challenge arises in the current NISQ era, where quantum computers typically handle only a few dozen qubits, whereas MBQC often demands resource states comprising hundreds or thousands of qubits[7].

To address this resource requirement, Bartolucci et al.[8] proposed Fusion-Based Quantum Computation (FBQC) in a photonic architecture. Their approach decomposes the large-scale cluster state, traditionally prepared offline for MBQC, into smaller graph states components. These smaller components are then dynamically fused during computation to build the computational network, significantly reducing the demands

on physical qubit count and coherence time. In environments where individual nodes offer only limited qubit resources, employing multiple nodes to prepare small graph states and collaboratively execute MBQC presents a promising strategy. Consequently, a key challenge lies in enabling these distributed nodes to prepare small graph states and efficiently fuse them to construct a large-scale graph state with sufficient resources for the intended computation.

Previous work has explored the distribution of graph states across multiple nodes. Cuquet et al.[9] described a method for distributing one-dimensional linear cluster states across a 1D quantum communication network, utilizing entanglement purification to enhance fidelity. Meignant et al.[10] proposed a graph state transfer protocol involving the distribution of GHZ states among remote nodes to create a complete graph state, followed by local graph state editing to achieve the target state. Fischer et al.[11] introduced an alternative connection transfer method: locally generating the target graph state at a specific node and then teleporting copies to destination nodes. Building upon Fischer’s work, Lin et al.[12] presented a method employing entanglement swapping to distribute Bell pairs instead of teleportation, offering increased efficiency for transferring local state copies. Our prior work proposed a graph distribution scheme[13] designed for quantum networks with resource constraints, demonstrating effective distribution under such limitations. Roberto et al.[14] proposed DODAG-X, a multiparty entanglement distribution protocol based on Directed Acyclic Graphs (DAG), which reduces computational overhead for topology verification and path discovery.

However, existing studies primarily focus on distributing small-scale graph states. For MBQC requiring large-scale and high-fidelity resource states, these methods suffer from prohibitive communication/purification overhead due to massive inter-node entanglement pair requirements. The communication and purification overhead required to generate the necessary inter-node entanglement pairs becomes prohibitively costly due to the massive scale and stringent fidelity requirements of the target state.

In this work, we introduce a novel path-routing and qubit-splitting based graph state distribution scheme designed to

enable collaborative MBQC execution across quantum communication networks. We also utilized our method to verify the Quantum Fourier Transform (QFT)[15] and the Quantum Approximate Optimization Algorithm (QAOA)[16] on a quantum network. Our approach explicitly addresses the unique structural characteristics of cluster states required for MBQC computation and simultaneously considers multiple real-world constraints prevalent in quantum networks: channel noise, resource limitations, transmission latency, as well as local computation-related noise, and processing delays. The scheme leverages the formal translation process from the gate circuit model to MBQC. It decomposes the required cluster state into a combination of a small number of multi-qubit horizontal linear cluster states and numerous vertical 2-qubit linear cluster states. Our strategy prioritizes finding suitable paths through the network nodes to deploy these horizontal clusters. Subsequently, we perform qubit splitting on these paths to define segments within the horizontal clusters. Finally, we efficiently distribute the abundant vertical 2-qubit linear cluster states using entanglement swapping.

Our key contribution lies in significantly reducing the consumption of critical resources, particularly the number of entangled pairs required for communication. This efficiency enables the construction and distribution of large-scale graph states, suitable for practical MBQC workloads, even within resource-constrained quantum communication networks.

II. PRELIMINARIES

In this section, we introduce fundamental concepts of MBQC essential to our work.

A. Graph States

Graph states represent a class of specialized multiqubit-entangled states. Formed by entangling qubits via controlled-phase gates (CZ), they exhibit specific graph structures. Cluster states are a subclass of graph states typically arranged in chain or lattice configurations. Their defining characteristic is the presence of direct or indirect entanglement between arbitrary qubit pairs, enabling global quantum computation driven by local measurements. This property establishes them as the core resource for MBQC.

Mathematically, consider n qubits initialized in state $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$. Applying controlled-phase gates $CZ_{i,j}$ between adjacent qubits yields the cluster state:

$$|\phi\rangle = \prod_{\langle i,j \rangle} CZ_{i,j} |+\rangle^{\otimes n} \quad (1)$$

where $\langle i,j \rangle$ denotes adjacent qubit pairs in the graph structure, and

$$CZ_{i,j} = |0\rangle\langle 0| \otimes I_j + |1\rangle\langle 1| \otimes Z_j \quad (2)$$

with I_j being the Identity operator and Z_j being the Pauli-Z operator.

B. Pauli-Basis Measurements

MBQC realizes quantum computational tasks by performing measurements on qubits of a cluster state. Specifically, Pauli-basis measurements on graph states can equivalently implement topological editing operations on the graph structure. The following subsections detail the effects of measurements in different Pauli bases.

Z-Basis Measurement: Measurement of vertex a which represents any qubit in the graph states in the Pauli-Z basis removes vertex a and its connecting edges from the graph. If the result $|1\rangle$ is obtained, Pauli-Z gates are applied to all neighbors N_a of a .

Y-Basis Measurement: Measurement of vertex a in the Pauli-Y basis is equivalent to performing local complementation at a before removing the vertex. The results of this measurement correspond to the eigenstates of the Pauli-Y operator, where $|+y\rangle$ and $|-y\rangle$ are defined as:

$$|+y\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle), \quad |-y\rangle = \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \quad (3)$$

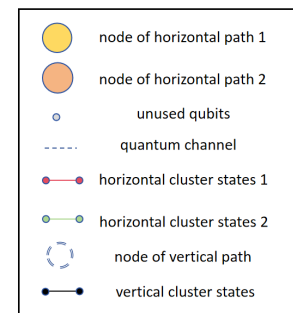
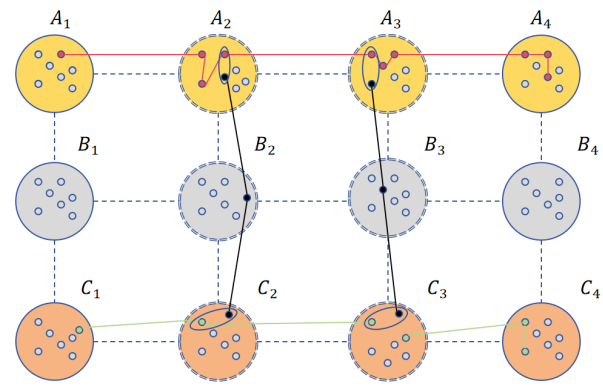
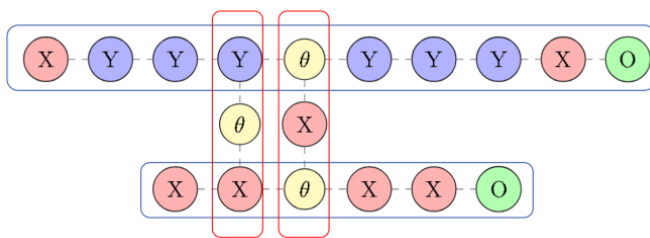
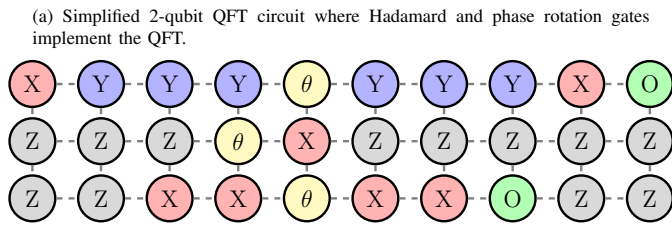
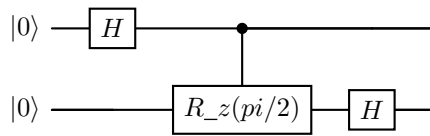
Here, $R_z(\theta)$ and $R_x(\theta)$ denote rotation gates around the Z-axis and X-axis by an angle θ , respectively. For outcome $|+y\rangle$, apply $R_z(\pi/2)$ to N_a ; for $|-y\rangle$, apply $R_x(\pi/2)$ to N_a .

X-Basis Measurement: Pauli-X basis measurement of vertex a proceeds as follows: First, apply local complementation to a neighbor $b \in N_a$. second, measure a in the Y-basis; third, reapply local complementation to b . For outcome $|+\rangle$, apply $R_y(-\pi/2)$ to b and Pauli-Z gates to $N_a \setminus (N_b \cup \{b\})$; for $|-\rangle$, apply $R_y(\pi/2)$ to b followed by Pauli-Z gates to $N_b \setminus (N_a \cup \{a\})$.

C. MBQC Implementation via Cluster States

As illustrated in Fig. 1a, a 2-qubit circuit for QFT can be implemented through MBQC using Pauli-basis measurements on the cluster state shown in Fig. 1b.

As the number of qubits and logical depth in quantum circuits increase, the size of the cluster state expands respectively in the vertical and horizontal directions. In most cases, the horizontal dimension of the cluster state is larger than its vertical counterpart. Furthermore, Z-basis measurement qubits can be omitted during state preparation, as such measurements effectively project and remove them from the state. Consecutive sequences of X-basis measurement qubits can also be simplified: even-length sequences reduce to just two qubits while preserving equivalent functionality. Through these optimizations, the resulting compact cluster state structure comprises: horizontal clusters, defined as M linear chains where M denotes the number of logical qubits, and vertical clusters, which are short linear clusters bridging these horizontal chains. We can find from Fig. 1c that the MBQC cluster state corresponding to the 2-qubit circuit for QFT can be decomposed into 2 horizontal cluster states and 2 vertical cluster states. This structural separation of horizontal and vertical clusters is fundamental to our resource-efficient distribution scheme.



D. The Cluster State in Quantum Network

A quantum network is composed of a series of nodes and the channels connecting them. As shown in Fig. 2a, a node contains a quantum processor and channel ports. The quantum processor can both prepare and store qubits and perform measurement and correction operations on them. These ports include ports for classical channels and ports for quantum channels. The quantum processor can send and receive classical information via classical channels and exchange entangled pairs via quantum channels.

In Fig. 2, horizontal cluster state 1 is generated from the red qubits on yellow nodes, horizontal cluster state 2 is generated from the green qubits on orange nodes, and two vertical cluster states are generated from the black qubits on nodes with dashed borders. By fusing the two qubits in their respective small ellipses on each of the four nodes into one qubit, the 2-qubit cluster state required for the MBQC corresponding to the QFT circuit can be constructed.

III. QUBIT-RESOURCE-AVAILABILITY-BASED HORIZONTAL CLUSTER-STATE ROUTING METHOD

In MBQC tasks, the required cluster states can be divided into horizontal and vertical types. Since MBQC task execution requires transmitting a large number of qubit measurement results for feedforward measurement or correction, horizontal

cluster states that occupy the majority of bits will transmit more classical information. Simultaneously, as quantum information is transmitted sequentially with destructive measurements, low-quality entanglement pairs require more resources and time for purification. Therefore, our scheme needs to prioritize multi-path routing of horizontal cluster states based on qubit resource availability, deploying horizontal cluster states on higher-quality paths in the network.

For the problem of selecting multiple paths to deploy cluster states, the following limiting factors need to be considered: First is the availability of qubit resources on the path. It should be ensured that the available qubit resources on each selected path can meet the needs of the cluster state. Second, on the premise that path resource availability is guaranteed, the selected multiple paths should have higher end-to-end fidelity and smaller classical communication delays. Meanwhile, since there is no priority difference among multiple paths, classical multi-path algorithms such as the Yen algorithm [17] and other greedy algorithms will lead to significant quality differences among the multiple paths. Therefore, this paper adopts a multi-path discovery algorithm based on the Genetic Algorithm [18] to find multiple paths that meet the requirements for horizontal cluster state deployment.

The constraints that the path needs to satisfy are divided into the following two aspects:

Availability of qubit resources on the path: Each node in the network has limited capacity to prepare cluster

states. Simultaneously, the initial fidelity of entanglement between nodes cannot meet the error tolerance requirements of the computing task, requiring consumption of large numbers of entanglement pairs for purification. The available qubit resources on each path should satisfy the following constraint:

$$\sum_{i=1}^{h_m} q_{n_m^i} \geq N_m \quad (4)$$

where h_m represents the hop of the m -th horizontal path, N_m denotes the number of qubit resources required for the m -th horizontal cluster state, and $q_{n_m^i}$ represents the available qubit number at node n_m^i which is the i -th node on the m -th path. The number of qubit resources consumed for purification per entanglement pair depends on the initial fidelity of the entanglement pair, which is estimated using the average fidelity of the quantum channel.

Fidelity per hop on the path: As the quality of quantum channels between node pairs may vary in quantum communication networks, and entanglement consumption during purification directly depends on the initial fidelity of the quantum channel, high-fidelity links should be selected during path selection. The number of rounds r is defined as the smallest positive integer satisfying the termination condition for purifying links between nodes n_m^i and n_m^{i+1} , formally expressed as:

$$r = \min \left\{ k \in \mathbb{N}^+ \mid \begin{array}{l} f_k \geq f_{(n_m^i, n_m^{i+1})}^{\text{target}}, \\ f_k = \frac{f_{k-1}^2}{f_{k-1}^2 + (1 - f_{k-1})^2}, \\ f_0 = f_{(n_m^i, n_m^{i+1})}^{\text{init}} \end{array} \right\} \quad (5)$$

where $f_{(n_m^i, n_m^{i+1})}^{\text{init}}$ denotes the initial fidelity between nodes n_m^i and n_m^{i+1} , $f_{(n_m^i, n_m^{i+1})}^{\text{target}}$ represents the target fidelity. In symmetric purification, each round consumes two pairs of low-fidelity entangled pairs to synthesize one high-fidelity entangled pair. Therefore, the constraint on the number of qubits consumed during the purification process can be derived as

$$2^r \leq \frac{q_{n_m^i}}{d_{n_m^i}} \quad (6)$$

where $d_{n_m^i}$ is the number of links requiring purification at node n_m^i

After satisfying the qubit-resource constraints, we define the path metric (individual fitness) as:

$$Fit_h(P) = \underbrace{\sum_{m=1}^M \sum_{i=1}^{h_m} C_{mi}}_{\text{Path Contribution}} - \lambda \Phi(P) \quad (7)$$

where the i -th link of the m -th horizontal path contribution term C_{mi} is:

$$C_{mi} = k_q \frac{q_{n_m^{i+1}}}{N_m} + k_f \ln \left(f_{(n_m^i, n_m^{i+1})}^{\text{init}} \right) - k_t t_{(n_m^i, n_m^{i+1})} \quad (8)$$

Here, $t_{(n_m^i, n_m^{i+1})}$ is the classical delay of the link between nodes n_m^i and n_m^{i+1} . k_q , k_f , k_t , and λ represent coefficients for node resource availability, link fidelity, classical time delay, and path set overlap penalty, respectively.

To avoid local optima in the genetic algorithm, we introduce the path set overlap penalty factor:

$$\Phi(P) = \sum_{e \in E} \mathbf{1}_{(\sum_{p \in P} \mathbf{1}_{e \in p} \geq 2)} \quad (9)$$

where E is the set of all edges in the quantum network, and $\mathbf{1}_{e \in p}$ is the indicator function:

$$\mathbf{1}_{e \in p} = \begin{cases} 1 & \text{if link } e \in p \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

This penalty prevents excessive link overlap that would concentrate entanglement consumption and prolong purification time.

Algorithm 1 Genetic Multi-path Routing (GMR)

Input: Network topology $\mathcal{G}$, source and destination pairs $\{(s_m, d_m)\}_{m=1}^M$, qubit requirements $\{N_m\}_{m=1}^M$, population size P , max generations $G_{\max}$

Output: Optimized paths $\mathcal{P}^*$

```

1: Initialize population  $\mathcal{P}$ 
2: while  $t < G_{\max}$  do
3:   for each individual  $\mathcal{P}_i \in \mathcal{P}$  do
4:      $Fit \leftarrow 0$ 
5:     for each path  $p_m \in \mathcal{P}_i$  do
6:       if  $\sum_k q_k < N_m$  then
7:          $Fit \leftarrow -\infty$ 
8:         break
9:       end if
10:      Compute  $C_{mk}$  for each hop in  $p_m$ 
11:       $Fit \leftarrow Fit + \sum_k C_{mk}$ 
12:     end for
13:      $Fit \leftarrow Fit - \lambda \Phi(\mathcal{P}_i)$ 
14:   end for
15:    $\mathcal{P}_{\text{new}} \leftarrow \emptyset$ 
16:   for  $j \leftarrow 1$  to  $P/2$  do
17:     Select  $\mathcal{P}_a, \mathcal{P}_b \leftarrow \text{tournamentSelect}(\mathcal{P})$ 
18:      $\mathcal{C}_1, \mathcal{C}_2 \leftarrow \text{crossover}(\mathcal{P}_a, \mathcal{P}_b)$ 
19:      $\mathcal{C}'_1 \leftarrow \text{mutate}(\mathcal{C}_1)$ ,  $\mathcal{C}'_2 \leftarrow \text{mutate}(\mathcal{C}_2)$ 
20:      $\mathcal{P}_{\text{new}} \leftarrow \mathcal{P}_{\text{new}} \cup \{\mathcal{C}'_1, \mathcal{C}'_2\}$ 
21:   end for
22:   for each candidate individual  $\mathcal{C} \in \mathcal{P}_{\text{new}}$  do
23:     if  $\mathcal{C}$  doesn't satisfy eq.(4) or eq.(6) then
24:        $\mathcal{P}_{\text{new}} \leftarrow \mathcal{P}_{\text{new}} \setminus \{\mathcal{C}\}$ 
25:     end if
26:   end for
27:    $\mathcal{P} \leftarrow \text{selectElites}(\mathcal{P} \cup \mathcal{P}_{\text{new}})$ 
28:    $t \leftarrow t + 1$ 
29: end while
30:  $\mathcal{P}^* \leftarrow \arg \max_{\mathcal{P}} Fit(\mathcal{P})$ 
31: return  $\mathcal{P}^*$ 

```

We encode the set of M paths as chromosomes. Using tournament selection and multi-point crossover, we exchange path segments between parent individuals. Mutation generates new segments or replaces entire paths. After repairing invalid offspring, an elite strategy narrows the search space, and iteration limits are applied as needed. Through this process, a set of globally near-optimal paths for deploying horizontal cluster states can be identified.

IV. VERTICAL CLUSTER-STATE SEGMENTATION METHOD BASED ON ENTANGLEMENT-PAIR DEPLOYMENT COST

After determining M horizontal paths required for cluster state deployment, it is necessary to generate small-scale cluster states at the nodes along these paths and purify the entangled pairs serving as quantum channels. Based on the size of cluster states prepared at each node and the positions of decomposed vertical linear cluster states within the original cluster state, the deployment scheme for vertical cluster states (hereafter referred to as the segmentation scheme) can be derived.

We define the vertical segmentation scheme S as:

$$S = \{s_m^i \mid m = 1, 2, \dots, M; i = 1, 2, \dots, h_m\} \quad (11)$$

where s_m^i denotes the number of bits segmented to horizontal path node n_m^i , subject to the constraint:

$$\sum_{i=1}^{h_m} s_m^i = N_m, \quad s_m^i \geq 0, \quad m = 1, 2, \dots, M \quad (12)$$

Since deployment nodes for vertical cluster states are not necessarily adjacent in the network, we employ the graph states distribution (GSD) scheme proposed by Lin et al.[12] to establish vertical linear cluster states (referred to as vertical entangled pairs) between non-adjacent nodes. Due to the high fidelity requirements for graph states in MBQC, these vertical entangled pairs require purification. The number of entangled pairs consumed by the GSD scheme is proportional to the hop count of the vertical entanglement path, while each hop along the path requires purification. Therefore, the most critical metric for evaluating vertical cluster state deployment schemes is the total hop count of all vertical entanglement paths, for which the Dijkstra algorithm can find the shortest vertical path. Additionally, similar to horizontal paths in Section 3, path link fidelity and classical time delay influence the cost of purification and the decoherence of cluster states. Therefore, we similarly employ these two indicators to calculate the fitness of segmentation schemes and utilize genetic algorithms to obtain globally approximately optimal segmentation schemes.

We define the i_v -th vertical path on the m -th horizontal path (where the source node is located) as $P_m^{i_v}$. The classical time delay on $P_m^{i_v}$ is defined as:

$$t(P_m^{i_v}) = \sum_{j=1}^{\text{hop}(P_m^{i_v})} t_{(n_m^{i_v,j}, n_m^{i_v,j+1})} \quad (13)$$

where $t_{(n_m^{i_v,j}, n_m^{i_v,j+1})}$ denotes the classical time delay between the j -th node $n_m^{i_v,j}$ and $(j+1)$ -th node $n_m^{i_v,j+1}$ on vertical path $P_m^{i_v}$.

The end-to-end fidelity of vertical path $P_m^{i_v}$ is defined as:

$$f(P_m^{i_v}) \approx \prod_{j=1}^{\text{hop}(P_m^{i_v})} f_{(n_m^{i_v,j}, n_m^{i_v,j+1})} \quad (14)$$

The fitness of vertical segmentation scheme S is:

$$\text{Fit}_v(S) = \sum_{m=1}^M \sum_{i=1}^{l_m} [k'_f \cdot f(P_m^{i_v}) - k'_t \cdot t(P_m^{i_v})] \quad (15)$$

where k'_f , and k'_t represent coefficients for vertical path hop count, fidelity, and classical time delay, respectively; l_m denotes the number of vertical paths on the m -th horizontal path. Our objective is to find the segmentation scheme that maximizes the fitness function $\text{Fit}_v(S)$. As previously mentioned, we still employ a genetic algorithm to seek an approximately optimal segmentation scheme. Each segmentation scheme S is encoded as a chromosome, with the fitness of each chromosome defined as shown in Eq.(15). The remaining procedures—including selection, crossover, and mutation—remain consistent with those in the genetic algorithm described in the preceding section. The detailed algorithm implementation is described in Algorithm 2.

Algorithm 2 Optimized Vertical Segmentation (OVS)

Input: Network $\mathcal{G}$, horizontal paths $\{P_m\}$,

qubit requirements $\{N_m\}$, qubit pairs $\mathcal{B}$

Output: Optimal segmentation scheme S^*

```

1: procedure MAIN
2:    $S \leftarrow \text{INITIALIZEPOPULATION}()$ 
3:    $S^* \leftarrow \emptyset, f^* \leftarrow -\infty$ 
4:   for  $t = 1$  to  $G_{\max}$  do
5:     for  $S_i \in S$  do
6:        $\mathcal{L}_i \leftarrow \text{BUILDVERTICALPATHS}(S_i)$ 
7:       if  $\text{HASNODECAPACITYVIOLATION}(\mathcal{L}_i)$  then
8:          $\text{SETFITNESS}(S_i, -\infty)$ 
9:       else
10:         $f \leftarrow \text{CALCULATEFITNESS}(\mathcal{L}_i)$ 
11:         $f \leftarrow f + \text{LOADBALANCINGPENALTY}(S_i)$ 
12:      end if
13:      if  $f > f^*$  then
14:         $S^* \leftarrow S_i, f^* \leftarrow f$ 
15:      end if
16:    end for
17:     $S' \leftarrow \text{GENETICOPERATIONS}(S)$ 
18:     $S \leftarrow \text{ELITEMIGRATION}(S, S')$ 
19:  end for
20:  return  $S^*$ 
21: end procedure

```

V. EXPERIMENTAL RESULTS

We evaluate MCBGD on 20×20 grid networks using NetSquid simulations [19], where qubit resources and channel parameters are randomly initialized with qubits per node

following a uniform distribution $\mathcal{U}(10, 50)$ and channel initial fidelity following $\mathcal{U}(0.75, 0.85)$. The target entanglement purification fidelity is set to 99.5%, and we compare MCBGD against EDCG and DODAG-X for QAOA and QFT circuits with 2–7 logical qubits.

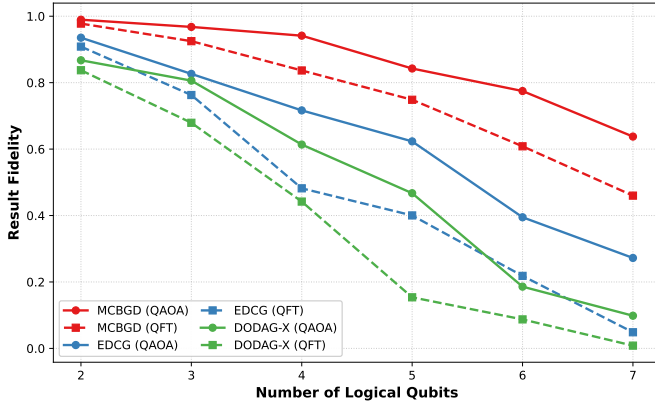


Fig. 3: Fidelity of the results in QAOA and QFT.

As illustrated in Fig. 3, MCBGD demonstrates superior fidelity preservation across varying logical qubit scales. For QAOA circuits at 5 logical qubits, MCBGD achieves a fidelity of 0.84, outperforming EDCG (0.62) and DODAG-X (0.46) by 35.5% and 82.6%, respectively. Similarly, for QFT circuits at 5 logical qubits, MCBGD maintains a fidelity of 0.74, compared to 0.41 (EDCG) and 0.18 (DODAG-X). The polynomial-scale growth of graph states inherently introduces fidelity degradation, but MCBGD’s resource-aware distribution—specifically its explicit modeling of node resource constraints during graph-state construction (unlike EDCG’s idealized infinite-resource assumption)—mitigates this effect by reducing errors from resource conflicts.

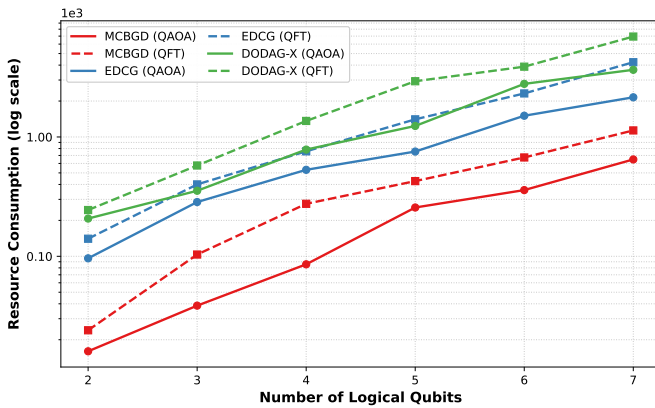


Fig. 4: Resource consumption for purification protocols in QAOA and QFT.

As shown in Fig. 4, resource consumption metrics further highlight MCBGD’s efficiency. For 7-qubit QFT circuits, MCBGD consumes only 1134.5 purification units, in stark contrast to EDCG’s 4229.7 and DODAG-X’s 6941.7 units. This corresponds to a reduction of 73.2% in resource

consumption compared to EDCG and 83.7% compared to DODAG-X. Similar reductions (67.3–82.1%) are observed for QAOA circuits. These efficiency gains stem from MCBGD’s architecture: instead of distributing a pre-built total cluster state (as in EDCG), MCBGD constructs the total cluster state from multiple local sub-cluster states, reducing the number of long-range entanglement swaps required and avoiding resource bottlenecks.

To deeper understand MCBGD’s advantages, we analyze the mechanisms behind its performance. The 35.5% fidelity improvement of MCBGD over EDCG (5-logical-qubit QAOA) arises from two key mechanisms: first, explicit resource constraint modeling during graph-state construction, where MCBGD accounts for node-specific qubit availability and channel limitations during horizontal deployment, avoiding the resource conflict overheads inherent to EDCG’s infinite-resource assumption; second, local entanglement optimization via vertical segmentation, where MCBGD converts long-range entanglement pair requirements into local short-range entanglement swaps, reducing reliance on low-quality long-range pairs and boosting overall fidelity.

MCBGD’s 73.2% reduction in purification resources for 7-qubit QFT circuits versus EDCG stems from its sub-cluster state construction architecture. By building the total cluster state from multiple locally constructed sub-clusters rather than distributing a monolithic cluster state (as in EDCG), MCBGD minimizes the need for error-prone long-range entanglement swaps. This design directly reduces per-link resource usage and eliminates redundant resource allocation, achieving significant efficiency gains.

Experiments further validate MCBGD’s scalability in a 20×20 grid (400 nodes). As logical qubit count increases from 2 to 7, MCBGD’s fidelity only drops by 12% (0.84 → 0.74), while EDCG and DODAG-X degrade by 31% and 58%, respectively. This indicates MCBGD’s multi-path routing and vertical segmentation strategies effectively mitigate increasing path complexity and entanglement pair demands with network scale, outperforming baselines in scalability.

These results confirm that MCBGD not only resolves the resource bottlenecks of traditional MBQC in large-scale cluster state preparation but also provides a feasible technical path for practical deployment of distributed quantum computing in the NISQ era.

VI. CONCLUSIONS

This paper introduces MCBGD, a novel framework designed to address the scalability challenges of MBQC in resource-constrained quantum networks. By hierarchically decomposing large-scale cluster states into horizontal and vertical components, MCBGD enables distributed deployment of quantum resources while balancing critical constraints such as qubit availability, channel fidelity, and communication latency. Its key advancements lie in three aspects: (1) A genetic multi-path routing algorithm optimizes horizontal cluster deployment through path overlap penalties, effectively reducing purification time compared to greedy methods; (2) An optimized

vertical segmentation scheme minimizes entanglement-path costs via genetic optimization, achieving superior resource efficiency over centralized approaches; (3) NetSquid simulations on 20×20 grids validate its practicality—5-logical-qubit QAOA/QFT circuits maintain robust fidelity (outperforming baseline methods) with significantly reduced purification resources. These results confirm MCBGD’s capability to enable distributed MBQC under realistic network conditions with limited resources.

Looking ahead, several directions warrant further exploration. First, we plan to extend MCBGD to support dynamic workloads by developing adaptive segmentation algorithms that adjust cluster-state decomposition in response to real-time resource fluctuations. Second, we aim to generalize the framework to heterogeneous quantum network topologies, including hybrid quantum-classical architectures and multi-domain networks, to broaden its applicability. Third, integrating MCBGD with emerging quantum hardware—such as trapped-ion or superconducting qubit processors—will be critical to validate its performance in real-world quantum communication environments. Finally, we will investigate robustness optimizations under imperfect noise models, including channel decoherence and measurement errors, to enhance MCBGD’s reliability for large-scale quantum algorithms like quantum chemistry simulation and combinatorial optimization. By addressing these challenges, MCBGD has the potential to bridge the gap between theoretical MBQC models and practical quantum computing deployments.

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] R. Raussendorf and H. J. Briegel, “A one-way quantum computer,” *Phys. Rev. Lett.*, vol. 86, pp. 5188–5191, May 2001.
- [2] R. R. Ferguson, L. Dellantonio, A. A. Balushi, K. Jansen, W. Dür, and C. A. Muschik, “Measurement-based variational quantum eigensolver,” *Phys. Rev. Lett.*, vol. 126, p. 220501, Jun 2021.
- [3] F. K. Marquversen and N. T. Zinner, “Applications and resource reductions in measurement-based variational quantum eigensolvers,” *Quantum Science and Technology*, vol. 8, no. 4, p. 045001, Jul 2023.
- [4] M. Proietti, F. Cerocchi, and M. Dispenza, “Native measurement-based quantum approximate optimization algorithm applied to the max k -cut problem,” *Phys. Rev. A*, vol. 106, p. 022437, Aug 2022.
- [5] T. Matsuo, T. Satoh, S. Nagayama, and R. Van Meter, “Analysis of measurement-based quantum network coding over repeater networks under noisy conditions,” *Physical Review A*, vol. 97, no. 6, p. 062328, 2018.
- [6] S. Barz, E. Kashefi, A. Broadbent, J. F. Fitzsimons, A. Zeilinger, and P. Walther, “Demonstration of blind quantum computing,” *science*, vol. 335, no. 6066, pp. 303–308, 2012.
- [7] K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, “Noisy intermediate-scale quantum algorithms,” *Rev. Mod. Phys.*, vol. 94, p. 015004, Feb 2022.
- [8] S. Bartolucci, P. Birchall, H. Bombín *et al.*, “Fusion-based quantum computation,” *Nature Communications*, vol. 14, p. 912, 2023.
- [9] M. Cuquet and J. Calsamiglia, “Growth of graph states in quantum networks,” *Phys. Rev. A*, vol. 86, p. 042304, Oct 2012.
- [10] C. Meignant, D. Markham, and F. Grosshans, “Distributing graph states over arbitrary quantum networks,” *Phys. Rev. A*, vol. 100, p. 052333, Nov 2019.
- [11] A. Fischer and D. Towsley, “Distributing graph states across quantum networks,” in *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 2021, pp. 324–333.
- [12] J. Lin, L.-Y. Hsieh, and C.-W. Tsai, “An enhanced graph state distribution method for quantum networks,” *Physica Scripta*, vol. 100, no. 4, p. 045111, Mar 2025.
- [13] H. Chen, T. Zhu, G. Yu, Y. Lin, and X. Jiang, “Distribution and segmentation methods of graph states based on quantum networks,” in *2025 27th International Conference on Advanced Communications Technology (ICACT)*, 2025, pp. 77–82.
- [14] R. Negrin, N. Dirnegger, W. Munizzi, J. Talukdar, and P. Narang, “Efficient multiparty entanglement distribution with dodag-x protocol,” 2025.
- [15] Y. S. Weinstein, M. Pravia, E. Fortunato, S. Lloyd, and D. G. Cory, “Implementation of the quantum fourier transform,” *Physical review letters*, vol. 86, no. 9, p. 1889, 2001.
- [16] L. Zhu, H. L. Tang, G. S. Barron, F. A. Calderon-Vargas, N. J. Mayhall, E. Barnes, and S. E. Economou, “Adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer,” *Phys. Rev. Res.*, vol. 4, p. 033029, Jul 2022.
- [17] J. Y. Yen, “Finding the k shortest loopless paths in a network,” *Management Science*, vol. 17, no. 11, pp. 712–716, 1971.
- [18] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor: University of Michigan Press, 1975.
- [19] T. Coopmans, R. Knegjens, A. Dahlberg, D. Maier, L. Nijsten, J. de Oliveira Filho, M. Papendrecht, J. Rabbie, F. Rozpedek, M. Skrzypczyk *et al.*, “Netsquid, a network simulator for quantum information using discrete events,” *Communications Physics*, vol. 4, no. 1, p. 164, 2021.