引用格式：刁蓉, 李宇航, 宋小宇, 等. 基于动态分区的多量子线路自适应映射 [J]. 电子科技大学学报, 2026, 55(4): 625-640.
Diao Rong, Li Yuhang, Song Xiaoyu, et al. Adaptive multi-programming mapping based on dynamic partitioning[J]. Journal of University of Electronic Science and Technology of China, 2026, 55(4): 625-640. (in Chinese)

基于动态分区的多量子线路自适应映射

刁　蓉[1]，李宇航[1]，宋小宇[1]，程学云[1*]，丁　飞[1]，管致锦[1, 2]

（1. 南通大学人工智能与计算机学院，南通 226019；2. 无锡太湖学院物联网工程学院，无锡 214063）

摘要：将逻辑量子线路映射至连通受限且有噪声的量子设备上，是量子计算的关键瓶颈。在量子设备上并行执行多条线路虽提升量子比特利用率，但也加剧了映射的复杂性。为了减少 SWAP 开销并提高保真度，提出基于动态分区的多量子线路自适应映射。通过预处理机制在映射前过滤高错误率的量子比特与物理连接，采用综合质量指标指导的动态分区策略，利用自适应映射策略生成高质量映射。在 IBMQ Toronto 上进行实验，平均保真度可达 52.39%，与现有方法相比提升 2.17%；映射开销上，平均插入 26.25 个额外 CNOT 门，SWAP 开销较其他方法降低 1.5 个 CNOT 门。该方法具有较好的保真度和较低的映射开销，为多量子线路映射提供更优方案。

关键词：量子计算；多量子线路映射；保真度；SWAP 开销
中图分类号：TP302.2　　**文献标志码：**A　　DOI: 10.12178/1001-0548.2025181

Adaptive multi-programming mapping based on dynamic partitioning

DIAO Rong[1], LI Yuhang[1], SONG Xiaoyu[1], CHENG Xueyun[1*], DING Fei[1], and GUAN Zhijin[1, 2]

(1. School of Artificial Intelligence and Computer Science, Nantong University, Nantong 226019, China;
2. School of Internet of Things Engineering, Wuxi Taihu University, Wuxi 214063, China)

Abstract: Mapping logical quantum circuits onto quantum devices with constrained connectivity and noise is a critical bottleneck for quantum computing. Executing multiple circuits in parallel on quantum hardware can increase qubit utilization, but it will also exacerbate mapping complexity. To reduce SWAP gate overhead and improve fidelity, an adaptive multi-programming mapping approach based on dynamic partitioning is proposed. A preprocessing mechanism filters out qubits with high error rates from physical connections prior to mapping. A dynamic partitioning strategy guided by a comprehensive quality metric is employed, and adaptive mapping is applied to generate high-quality mappings. Experiments on IBM Quantum Toronto achieved an average fidelity of 52.39%, representing a 2.17% improvement over existing methods. Regarding mapping overhead, an average of 26.25 additional CNOT (controlled NOT) gates were inserted, while SWAP gate overhead was reduced by 1.5 CNOT gates compared to other methods. The proposed method provides an optimized solution for multi-programming mapping, demonstrating the merits of both high fidelity and low mapping overhead.

Key words: quantum computing; multi-programming mapping; fidelity; SWAP overhead

量子计算作为一种新型计算范式，通过利用量子比特的叠加与纠缠特性，在解决特定复杂问题上展现出超越经典计算的优势[1]。然而，当前含噪声中等规模量子（noisy intermediate-scale quantum, NISQ）的计算设备，由于连接受限且极易受到噪声和退相干的影响，在实现复杂量子算法时面临严峻挑战[2-3]。

量子算法的实现依赖于量子编译过程，其中核心环节是量子线路映射[4]，即将逻辑量子线路转换为适配 NISQ 设备物理约束的目标线路的过程[5]。随着量子计算设备规模的扩大和各种量子云平台的出现，现有量子线路映射技术面临硬件使用率低和用户等待时间长等问题，因此提出了多量子线路映射方案，但这带来两方面问题：一方面，需要通过

收稿日期：2025−09−09
基金项目：国家自然科学基金面上项目（62072259）；江苏省自然科学基金（BK20221411）；南通市自然科学基金（JC2024100）；南通大学博士启动基金（23B03）
作者简介：刁蓉，主要从事多量子线路映射方面的研究。
*通信作者 E-mail: chen.xy@ntu.edu.cn

对量子计算设备架构分区来分配高保真度物理资源并抑制任务间的串扰；另一方面，线路的初始映射会影响后续为满足近邻约束而引入的 SWAP 门数，从而影响量子线路并行执行的保真度。

针对多量子线路映射问题，目前已有很多相关研究。在分区与资源分配方面，文献 [6] 提出公平可靠分区方法，并通过延迟指令调度来缓解退相干，但在初始映射阶段未充分考虑拓扑结构；文献 [7] 提出基于社区划分的多线路映射机制，引入跨程序 SWAP 以降低门数开销，而其扩展版[8] 在此基础上进一步面向 2D/3D 架构，增加了串扰感知的映射与调度策略，从而提升了多程序执行下的保真度与资源利用率；文献 [9] 则致力于通过在并发门之间放置空闲比特缓解串扰误差，但未充分利用可靠量子比特。此外，文献 [10] 通过强化学习的方法学习量子设备噪声特性优化布局，但训练过程复杂且资源消耗大；文献 [11] 通过连接可靠性指数优化资源分配，解决了在分区效率和安全性上的不足。在映射方面，主要分为最优方法和启发式方法。最优方法将问题转化为数学优化问题求解，如 Qmap[12]、OSLQ-GA[13] 和 BIP[14] 利用 MOVE 或 SWAP 操作减少门数，SATMAP[15] 基于线路结构降低问题规模；启发式方法灵活但无法保证最优解，如 SABRE[16] 通过减少搜索空间加速映射，NASSC[17] 通过前瞻和动态评估代价函数，实现更优的 SWAP 门插入，从而降低线路开销。

现有研究方法在映射过程中对线路的逻辑结构特点考虑不足，未能实现与设备架构拓扑的最优适配，导致结果保真度损失较大。为解决上述问题，本文提出了基于动态分区的多量子线路自适应映射。其核心思想是在映射最初阶段，感知各物理量子比特的读出错误率、不同物理连接的 CNOT 门错误率及其在架构拓扑中的连通性，同时充分考虑量子线路的逻辑交互与计算资源需求，为每个并行线路生成高质量的映射方案，从而减少 SWAP 开销并提高保真度。

1 基本概念

1.1 量子线路

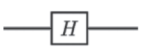
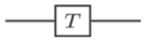
在经典计算机中，基本运算单元为经典比特，其状态仅能取 0 或 1。量子系统的基本单位是量子比特，量子比特同样存在两种基本状态，通常记为 $|0\rangle$和$|1\rangle$。但是与经典比特不同的是它可以处于这两种状态的某种线性叠加态$|\psi\rangle$，即量子比特是两个基态的线性组合[1]，表示为：

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{1}$$

式中，$|\psi\rangle$是量子比特的状态向量，简称为量子态；α和β均是复数，称为量子态$|\psi\rangle$在$|0\rangle$和$|1\rangle$上的概率幅。$|\alpha|^2$和$|\beta|^2$分别是测量该量子态得到 0 和 1 时的概率，且满足$|\alpha|^2 + |\beta|^2 = 1$。

量子门是量子计算的基本操作，用来控制量子比特的状态，从而实现量子计算。本文研究主要基于 IBM 量子计算云平台，该平台所采用的量子线路的量子门主要来源于 Clifford+T 门库[18]。表 1 呈现了 Clifford+T 门库中一些常用量子门的符号表示及对应的酉矩阵形式。

表 1 常用量子门

名称	符号表示	矩阵表示
Not gate		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Hadamard gate		$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
T gate		$\begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{bmatrix}$
S gate		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
CNOT gate		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

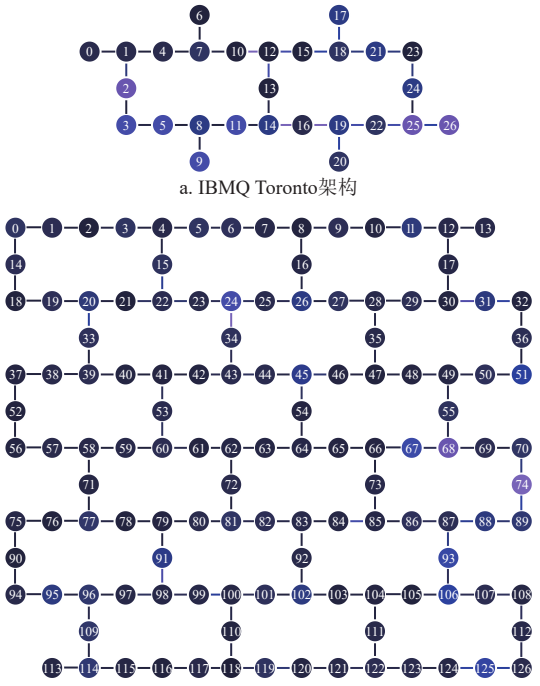
量子线路是量子计算的基本执行方法，由一系列量子门操作组成，用于实现量子算法。在逻辑量子线路中每条线对应一个逻辑量子比特，每个节点代表一个量子门，而线与节点的连接则体现了量子比特间的相互作用。

1.2 量子计算设备

量子计算设备的核心组件是量子处理器 (quantum processing unit, QPU)，其功能是执行量子线路。QPU 的关键特性包含基门、架构拓扑图以及校准数据等。其中，基门指 QPU 所支持的量子门类型，在量子线路执行前的编译阶段，需将线路中所有的门转换为基门的组合。

超导量子计算架构是当前最具潜力的量子计算架构之一，为了确保量子计算设备的性能和质量，架构图采用了相对稀疏的量子比特拓扑结构。当前 IBM 公司已经用超导技术制备出 5、7、16、20、27、65、127 等比特的量子计算设备[19]，其中 IBMQ Toronto 架构和 IBMQ Brisbane 架构如图 1

所示，图中每个节点代表一个物理量子比特，节点之间的连接表示相应量子比特之间的可交互关系或量子门操作。



a. IBMQ Toronto架构

b. IBMQ Brisbane架构

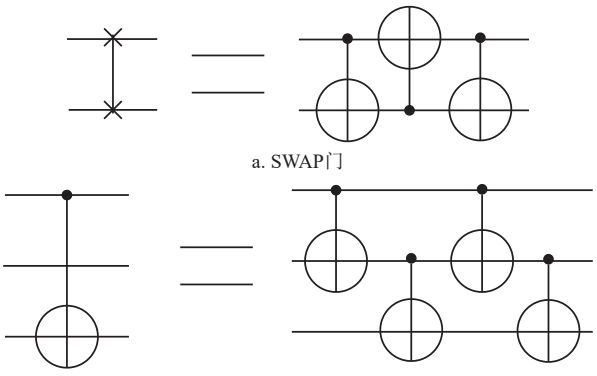
图 1　量子计算设备架构示意图

NISQ 设备中的量子比特对噪声高度敏感，主要误差包括：相干错误[20]、操作错误[21]、测量误差[22] 以及串扰误差[23] 等，其中，串扰误差可通过同步随机基准测试量化[24]。尽管量子云平台定期提供校准数据[25]，如门错误率、读出错误率和相干时间等关键参数，但它们仍随量子比特、耦合链路及时间的变化而波动，如图 1 中节点和边的颜色深浅分别表示物理量子比特读出错误率以及量子比特间门错误率的大小。因此，这些噪声特性不仅揭示了硬件的物理局限，也要求线路映射过程需充分利用校准数据，优先选择错误率较低的量子比特和连接，以提升计算保真度。

1.3　量子线路映射

量子线路映射是为了使逻辑线路满足量子架构的连通性约束而进行的量子线路等价变换，其中包括初始映射和路由两个阶段。初始映射阶段将线路中逻辑量子比特映射至架构中的物理量子比特上；路由阶段则确定量子比特之间的交互路径，以确保所有门操作在量子架构连通性受限约束下得以正确执行[26]。

为了满足量子架构执行量子线路的连通性约束，可通过插入 SWAP 门或采用 Bridge 门等方式来解决。如图 2a 所示，SWAP 门用于交换两个量子比特的量子态，一个 SWAP 门可以分解为 3 个级联的 CNOT 门。图 2b 所示的 Bridge 门由 4 个 CNOT 门组成，效果相当于单个 CNOT 门并且不改变原有的映射布局[27]。然而，这会引入额外的双量子比特门，进而产生更多的噪声，因此需要最小化它们的数量。



a. SWAP门

b. Bridge门

图 2　量子比特的通信方案

1.4　多量子线路映射

随着 NISQ 设备规模的不断扩大，单条量子线路在执行时往往仅能利用设备的部分可用资源，导致大量量子比特处于空闲状态，整体资源利用率低下。为此，将多条量子线路同时映射到同一量子设备上执行，可有效提升设备的资源利用率与计算吞吐量，这种方式称为多量子线路映射。

在该过程中，通常需要对量子设备的架构拓扑图进行分区，即将整个架构拓扑图划分为若干个子区域，每个子区域专门分配给一条量子线路执行，从而在保证多线路并行性的同时尽量降低线路间的串扰与资源竞争。其中本文所述的“架构拓扑图”即量子设备架构的抽象图，如图 3 所示，节点代表物理量子比特，边表示可执行双比特门的物理耦合。

在多量子线路映射过程中需要遵循以下 4 个约束条件。

1）不同逻辑线路映射到物理架构中不同的分区上；

2）分区之间必须相互独立，不能共享物理量子比特；

3）每个逻辑量子比特必须唯一地映射到一个物理量子比特上；

4）量子线路执行时，双量子比特门只能在相邻的量子位上交互。

在图 3 的多量子线路映射模型中，量子设备架

构拓扑图被分成多个分区，需要并行执行的各条线路只能分配到 QPU 上的唯一区域。图中 27 量子位的拓扑图被划分为 4 个分区，分别为 p_1、p_2、p_3和p_4，每个分区一次只分配给一条线路，图中每条线路选择映射到各自适配的分区中。

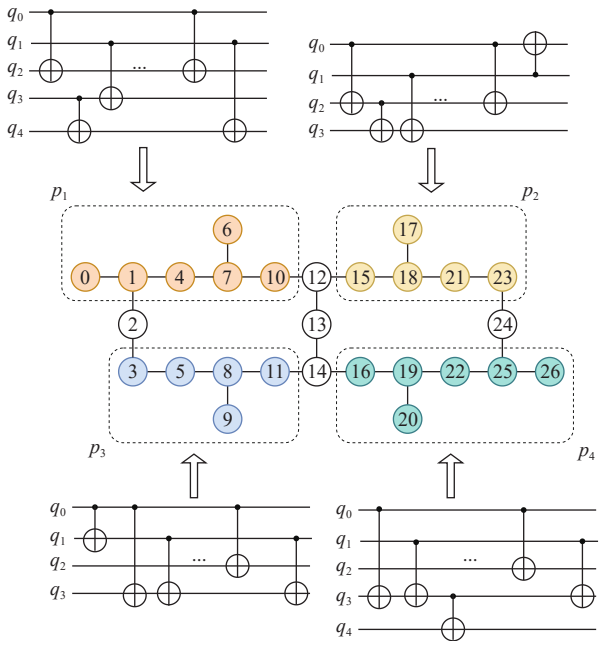


图 3　多量子线路映射模型

为了满足上述约束条件，并且需要保证多线路执行的可行性与尽可能高的保真度，其主要工作包括：

1）对设备架构拓扑图分区，避免线路间的资源竞争；

2）若物理量子比特不相邻，需要插入一个或多个 SWAP 门以实现近邻化，其涉及的物理量子比特严格限制在分区内部，不得跨分区；

3）多线路映射时，应避开设备中串扰强的区域，以保持较高的执行保真度。

2　动态分区

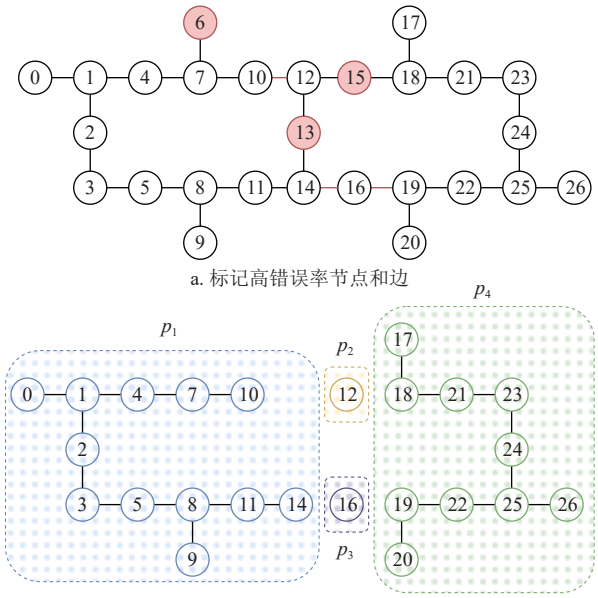
在将多个逻辑量子线路映射到物理架构拓扑图前，需先对架构拓扑图进行划分，为不同量子线路构建清晰的资源边界，减少资源竞争和噪声串扰。基于此，本文设计的划分方法包含架构拓扑图预处理和动态分区两个阶段。首先，通过高保真度子图筛选策略对物理架构拓扑图进行预处理，以构建可靠的执行基础。然后，利用动态分区策略将高保真度子图划分为内部紧密连接且分区间相互隔离的区域。

2.1　高保真度子图筛选

真实的量子计算设备中，不同物理量子比特的读出错误率与物理耦合的门错误率存在差异。若不加区分地使用这些物理量子比特，将导致映射方案的整体性能参差不齐。为了缓解物理量子比特之间保真度差异对整体映射性能的影响，本文提出了针对原始架构拓扑图的预处理方法。

该策略首先统计量子架构中所有物理量子比特的读出错误率与物理耦合边的 CNOT 门错误率，通过设定的阈值标识出错误率较高的物理比特与耦合边，其中该阈值可根据当前设备的整体性能水平动态调整。随后，在原始架构拓扑图中移除这些被标记的物理比特与耦合边，最终构建出由高保真度物理比特与耦合边组成的拓扑子图。

示例 1　如图 4a 所示，在 27 量子位的 IBMQ Toronto 架构拓扑图上，Q_6、Q_{13}和Q_{15}以及(Q_{10}，Q_{12})、(Q_{14}，Q_{16})和(Q_{16}，Q_{19})被标记为高错误率的节点和边。接着移除它们，得到如图 4b 所示的分区。



a. 标记高错误率节点和边

b. 移除标记的节点和边

图 4　预处理架构拓扑图

该过程确保后续的分区与映射在可靠的硬件基础上进行，从而为提升并行执行的保真度奠定了基础。为平衡该策略对路由可能构成的约束，后续的 2.3 节通过按需动态分区合并机制对分区进行合并。由此，本文方法既能充分利用高保真度物理资源，又避免了因过度筛选而引发的路由瓶颈。

2.2　拓扑感知的动态分区

在获得高保真度架构拓扑子图后，下一步是将其分割为多个适用于并行执行的分区。理想的分区不仅应包含低读出错误率的物理量子比特和低 CONT 门错误率的物理耦合，还应具备较好的拓扑结构。为此，本文给出了评估分区质量的综合指标，并基于此设计了拓扑感知的动态分区方法。

2.2.1　综合质量评估指标

为了评估分区优劣，本文提出了两个综合质量指标，它们是后续分区方法决策的核心依据。

理想的物理量子比特不仅应具备较高保真度，还应与低 CNOT 错误率的邻接耦合边相连，并在拓扑图中处于高度连通且路径距离较短的位置。为此，本文设计了“节点综合保真度”，它是量化物理量子比特综合性能的指标，结合了该比特的读出错误率、相邻连接的平均 CNOT 错误率和拓扑连通性。

设架构拓扑由无向图$G=(V,E)$表示，其中V是物理量子比特的集合，E是边集，其每条边代表量子比特间的物理耦合。对于图中任意一个代表物理量子比特的节点$v\in V$，其节点综合保真度F_v为：

$$F_v=\alpha(1-R_v)+\beta\left(1-\frac{\sum_{u\in N(v)}E(v,u)}{|N(v)|}\right)+\gamma\frac{\deg(v)}{|V|-1} \quad (2)$$

式中，α、β、γ为可调权重系数，满足$\alpha+\beta+\gamma=1$，允许根据具体硬件特性或优化目标调整这 3 个维度（内在保真度、局部连接质量、全局拓扑重要性）的相对优先级。F_v的值越高，代表该比特的综合质量越优。

式 (2) 等式右边第一项旨在量化物理量子比特v的内在保真度，R_v是v的读出错误率。该项表征了量子比特作为独立信息单元的基础可靠性。将量子比特读出错误率作为考量是因为任何量子计算的最终结果均需测量得到，高读出错误率的比特将从根本上损害线路的执行结果。因此，低读出错误率是评判物理量子比特质量的先决条件。

式 (2) 等式右边第二项旨在量化物理量子比特v的局部物理连接质量，$N(v)$是v的邻居比特集合，$E(v,u)$是在v与其邻居u之间的物理连接上执行 CNOT 门的错误率。它主要评估v与其所有邻居比特进行 CNOT 门操作的平均保真度。量子算法的实现高度依赖于量子比特间的交互。一个物理量子比特即便自身保真度极高，若它与邻居的物理耦合边质量低下，其执行相应的 CNOT 门操作的正确率也将受到影响。因此，该项考虑了连接该节点的边也应具备高保真度，反映了该量子比特处于整体门操作错误率较低的局部交互环境中。

式 (2) 等式右边第三项旨在量化物理量子比特v在全局拓扑中的连通性水平，$\deg(v)$是v的度数，即与该量子比特v直接相连的边的数量，而$|V|-1$为节点的最大度数。在线路映射过程中，将频繁交互的逻辑比特映射至高连通性的物理比特上，可减少后续 SWAP 门的插入，进而降低线路总深度与附加噪声。此项确保了所选节点在物理布局上占据优势位置。

评估物理耦合边的质量时，仅考虑其自身的 CNOT 错误率是片面的。理想的物理耦合边，不仅自身应具备低 CNOT 错误率，还应连接两个高保真度的物理量子比特。为此，本文设计了“边综合权重”，它是量化物理量子比特u和v之间物理连接的综合质量指标。

设架构拓扑为无向图$G(V,E)$。对于图中任意一条代表物理耦合的边$e=(u,v)\in E$，其边综合权重W_{uv}为：

$$W_{uv}=\frac{E_{uv}-E_{\min}}{E_{\max}-E_{\min}}+\left(1-\frac{F_u+F_v}{2}\right) \quad (3)$$

边综合权重W_{uv}通过加和连接自身的错误代价与端点节点的质量代价，提供了更全面的物理连接评估标准。W_{uv}值越低，表明该耦合边不仅自身错误率低，且连接的量子比特综合质量也好。

式 (3) 等式右边第一项 E_{uv} 是耦合边（u,v）上执行 CNOT 门错误率；$E_{\min}$和$E_{\max}$分别是拓扑图中所有耦合边的最小与最大 CNOT 门错误率。该指标采用归一化方法，将绝对错误率线性映射至区间$[0,1]$。这种处理方法不仅消除了不同硬件平台间的错误率差异带来的不可比性，从而提升了指标的跨平台普适性，同时也保证了该代价项与后续指标在量纲上的一致性。

式 (3) 等式右边第二项为耦合边(u,v)两端的物理量子比特u和v的综合质量代价，F_u和F_v分别表示物理量子比特u和v的节点综合保真度。将所连接的节点的质量作为代价进行计算，可以确保最终的优质耦合边，其两端是高质量的物理量子比特。

为指导分区向全局最优收敛，需要一个量化评估分区方案优劣的全局目标函数。该函数必须平衡两个核心目标：确保每个分区内部量子比特的高保真度和高连通性，以及分区间的相互隔离。为此，本文设计了分区方案的总体质量函数。

设$P=\{p_1,p_2,\cdots,p_i,\cdots,p_k\}$是拓扑图$G(V,E)$的分区方案。为了对分区方案$P$进行评估，本文设计的分区总体质量函数$Q_P$可表示为：

$$Q_P=a\sum_{i=1}^{k}\mathrm{Coh}(p_i)+b\mathrm{Sep}(p_i,p_j) \quad (4)$$

Q_P是分区方案P的函数，取值依赖于给定的分区结果。不同的分区方案会导致$E_{\text{int}}(p_i)$、E_{cut}、L^i_{avg}等变量发生变化，从而使得Q_P的值不同。另外，a和b是权重系数，用于平衡分区质量评估中的两个核心目标。

Q_P由如下两个核心部分构成。

1） 分区内部凝聚力：评估分区内部的质量。可表示为：

$$\text{Coh}(p_i)=\frac{\sum_{e\in E_{\text{int}}(p_i)}\left(1-\frac{W_e}{W_{\max}}\right)+\sum_{v\in p_i}F_v}{D_i|P_i|/L^i_{\text{avg}}} \tag{5}$$

式中，$E_{\text{int}}(p_i)$代表分区p_i内部边的集合；W_e代表边的权重；D_i代表分区p_i的直径；L^i_{avg}代表分区p_i的平均最短路径。$\text{Coh}(p_i)$是对各个分区p_i的内部凝聚力分数进行求和，内部节点综合保真度总和代表分区内含有的基础硬件资源的质量，内部边综合权重总和代表分区内部边的可靠性，拓扑效率是惩罚项，避免生成结构松散的分区，从而引导选择拓扑上更为紧凑的子图。

2） 分区间隔离度：评估不同分区p_i和p_j间的隔离程度。可表示为：

$$\text{Sep}(p_i,p_j)=\sum_{i\neq j}\frac{\sum_{e\in E_{\text{cut}}}W_e}{W_{\max_\text{cut}}}+\frac{1}{d} \tag{6}$$

式中，E_{cut}代表切割边的集合；$W_{\max_\text{cut}}$代表切割边集合中边的最大权重；d代表分区间平均最短路径长度。通过跨分区的切割边权重和分区节点间的最短路径长度，旨在实现分区间尽可能独立，从而缓解并行时的串扰。

综上，分区总体质量函数Q_P综合考虑了分区内部凝聚力与分区间隔离性。在分区过程中，目标是寻找最大化Q_P的分区方案，从而在提升分区内部凝聚力的同时，尽量降低分区间相互干扰的代价。

2.2.2 迭代切割的动态分区方法

基于 2.2.1 节的综合质量评估指标，本文采用自顶向下的迭代切割方法对拓扑图进行动态分区。其核心在于利用最小生成树的启发式切割，通过最大化分区总体质量函数Q_P来指导迭代过程。

该方法从包含整个连通分量的初始分区开始，通过迭代切断权重最大的边来寻找更优的分区方案。此处权重最大的边可以通过定理 1 判定。

定理 1 在架构拓扑子图中，如果将其物理连接的边综合权重视为切割代价，那么该子图的最小生成树中代价最大的边，必然是所有可能将该子图一分为二的切割方案中，跨越两个新分区边界的连接里权重最大的一条。

证明：设$G=(V,E,w)$为连通的加权无向图，其中V是物理量子比特的集合，E为边集，其中每条边代表量子比特间的物理耦合，$w(e)$代表每条物理连接的切割代价。设T为G的最小生成树，且$e_{\max}=(u,v)$为T中权重最大的边，即对于所有$e\in E(T)$，满足$w(e_{\max})\geqslant w(e)$。在$T$中移除边$e_{\max}$会将节点集$V$划分为两个子集$S_u$和$S_v$。

假设存在边$e'=(x,y)\in E$，其权重严格小于$e_{\max}$，即$w(e')<w(e_{\max})$。构造新的生成树T'，其中e'连接了因移除$e_{\max}$而分离的两个连通分量，因此T'为G的生成树。比较T'与T的总权重：

$$W(T')=W(T)-w(e_{\max})+w(e')$$

根据假设$w(e')<w(e_{\max})$，可以得出：

$$W(T')<W(T)$$

这表明，T'的总权重严格小于T的总权重，这与T是最小生成树的定义相矛盾。因此，假设不成立。这意味着不存在任何连接S_u和S_v的边e'满足$w(e')<w(e_{\max})$。故对于所有连接这两个子集的边e'，必有$w(e')\geqslant w(e_{\max})$。

此结论表明，在所有连接子集S_u和S_v的边中，$e_{\max}$权重最高。证毕。

本文设计的迭代分区算法，其核心思想是全局择优的贪心策略。首先，基于定理 1，在当前所有分区内构建最小生成树，并将其权重最大的边作为候选切割边；然后，利用分区总体质量函数Q_P对每个候选切割进行评估，计算其对整个分区方案的质量增益。最后，仅采纳并执行能带来最大正增益的唯一最佳切割。此迭代过程持续进行，直至无法找到任何可提升Q_P的切割方案为止。这种方法确保了每一步决策都服务于全局最优目标，有效规避了局部最优，保证了最终分区方案的高质量。

动态分区算法的伪代码算法如下所示。

算法 1: 动态分区算法

Input: $G'=(V, E)$, C //输入处理后的拓扑图和线路集合

Output: P_all //输出分区结果集合

P_all ← Ø;//初始化

for each component in Connected(G') do//遍历所有连通分量

```
    if |component| < MIN_SIZE then
      P_all ←P_all ∪ {component};
    else
      subG ← Subgraph(G', component);//提取连通子图
      P_cur ← {component};//初始分区方案为整个连通分量
      improvement ← True;
      while improvement do
        best_gain ← 0;//记录最佳增益值
        P_best ← P_cur;
        for each P in P_cur where |P| ≥ 2*MIN_SIZE do
          T ← MST(P); //生成最小生成树
          for each (u,v) in TopEdges(T, LIMIT) do//检查 MST 中权重最高的若干条边
            P_trial ← Split(P, u, v) ∪ (P_cur − {P});//分区拆分
            if ∀P ∈ P_trial with |P| ≥ MIN_SIZE then
              gain ← Evaluate (subG, P_trial);//计算新分区的质量增益
              if gain > best_gain then//更新最佳方案
                best_gain ← gain;
                P_best← P_trial;
              end if
            end if
          end for
        end for
        if best_gain >ΔQ0 then//终止条件
          P_cur← P_best;
        else
          improvement ← False;
        end if
      end while
      P_all ← P_all ∪ P_cur;
    end if
  end for
  return P_all//返回分区结果
```

在时间复杂度上，算法 1 的主要计算开销来自迭代分区过程。对于给定的连通分量，其内部的迭代次数在最坏情况下与顶点数$|V|$成正比，每次迭代需构建最小生成树，时间复杂度为$O(|E|\log|V|)$，并评估候选切割方案，时间复杂度约为$O(|V|+|E|)$。综合外层循环与内层迭代，总时间复杂度在最坏情况下为$O(|V|^3\log|V|)$。算法 1 需存储输入图结构、当前分区方案与最终分区集合，总空间复杂度为$O(|V|+|E|)$。

示例 2　在移除高错误率的节点和边后，初始分区集合为$\{p_1,p_2,p_3,p_4\}$，如图 4b 所示。由于p_2和p_3尺寸过小，不对其进行切割。因此，接下来的分区过程主要针对p_1和p_4进行，其流程如下。

第 1 轮

1）全局搜索：在当前分区集合中找到最大化分区总体质量函数Q_P的唯一最佳切割点。

2）尝试分割p_1：首先为分区p_1构建最小生成树，并识别出权重最高的若干条候选切割边。切割边A将p_1分割为p_{1_A1}和p_{1_A2}。计算新分区集合$\{p_{1_A1},p_{1_A2},p_2,p_3,p_4\}$相对于初始分区集合的质量增益，假设为$\Delta Q_{P_A}=5.2$；接着切割边$B$得到的增益为$\Delta Q_{P_B}=4.8$；经过若干次尝试，记录下在$p_1$中能找到的最大增益是 5.2。如图 5 所示。

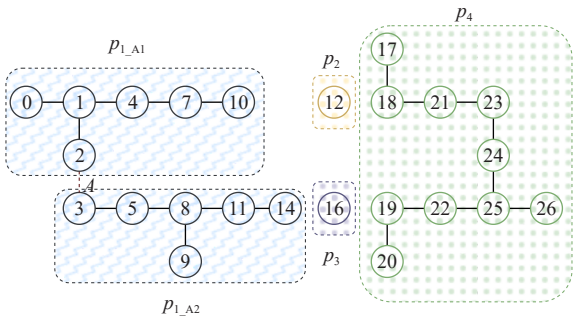


图 5　迭代分区示意图

3）尝试分割p_4：用同样的方法为p_4构建最小生成树并评估其候选切割边。在p_4中，切割边C带来的增益为$\Delta Q_{P_C}=3.1$；经过多次尝试，记录下在p_4中能找到的最大增益是 3.1。

4）本轮决策：检查完所有可分割的分区后，比较所有记录下的最大增益值。本轮全局最佳增益$\Delta Q_{\max}=5.2$，来自对p_1切割边A的操作。

5）执行更新：只执行此最佳操作。当前分区集合更新为$\{p_{1_A1},p_{1_A2},p_2,p_3,p_4\}$。

第 2 轮

新一轮全局搜索：基于更新后的分区集合，重复第 1 轮的过程。

算法收敛

1）迭代与收敛：只要每一轮能找到带来正增益的全局最优切割方案，该过程将持续进行。

2）终止条件：直到某一轮完整的全局搜索结束后，发现所有候选切割方案带来的增益不大于

零，即$\Delta Q_P \leqslant 0$，迭代循环终止。此时的分区集合被视为最终的分区结果。

2.3 多线路资源分配

在将拓扑图划分为多个分区后，下一步需要将线路分配到物理分区上。简单的分配策略可能因无法满足线路的资源需求或资源碎片化而失败。

为形式化定义此问题，给定待执行的线路集合$C_{\text{set}} = \{c_1, c_2, \cdots, c_i, \cdots, c_j, \cdots, c_k\}$和分区集合$P_{\text{set}} = \{p_1, p_2, \cdots, p_m\}$，多量子线路映射可行性问题旨在寻找分配函数$A$: $C_{\text{set}} \to P_{\text{set}}$，需满足以下条件：

1）对于任意线路$c_i \in C_{\text{set}}$，分配到的目标分区$A(c_i)$的规模必须不小于c_i的规模，即$|A(c_i)| \geqslant |c_i|$；

2）对于任意两个不同的线路$c_i, c_j \in C_{\text{set}}$，被分配到的分区不相交。

定理 2 对于给定的静态分区方案P_{set}，如果存在线路c_k，其规模大于所有单个分区的最大规模，即$|c_k| > \max_{p_i \in P_{\text{set}}} |p_i|$，则多线路映射可行性问题无解。

证明：根据可行性问题的定义，任何有效的分配方案都必须满足资源充足性条件，即任何待分配的线路c_k都能找到分区$p_j \in P_{\text{set}}$使得$|p_j| \geqslant |c_k|$。在定理 2 的条件下，$|c_k| > \max_{p_i \in P_{\text{set}}} |p_i|$，这意味着对于所有$p_i \in P_{\text{set}}$，都有$|p_i| < |c_k|$。因此，线路$c_k$无法找到满足资源充足性条件的目标分区。故不存在有效的分配函数 A 满足可行性问题的条件。因此，该问题无解。证毕。

定理 2 揭示了静态分区的局限性。当线路规模超过任何单个分区时，静态分区将导致分配失败。在此基础上，本文提出按需动态分区合并机制，在因分区尺寸不足而导致分配失败的条件下被触发，通过动态组合多个邻近的未使用分区，构建出满足线路规模的新分区。该机制将在静态分区约束下无解的问题转化为具备可行解的问题，从而提升了资源分配方案的灵活性和成功率。

实现多量子线路分配流程如下，首先，根据线路的 CNOT 密度构建优先级调度队列，优先处理资源需求最高的线路。随后，在所有可用分区中为每条线路执行最优匹配搜索，对每个分区应用映射算法进行最优映射的搜索与质量评估。为了应对静态分区的局限性，算法集成了按需动态合并机制，若线路因尺寸过大而首次分配失败，该机制可以自动合并相邻未使用分区以创建足够大的新分区，并启动第二次分配尝试。

资源分配算法的伪代码算法如下。

算法 2: 资源分配算法

Input: C, P, G_{f}, G_{full}, $t_{\max}$//输入待执行线路集合、初始分区集合、处理后拓扑图、完整拓扑图和最大尝试次数

Output: A_{map} //输出每条线路分配的所在分区

Qprio ← SortByConstraint(C);//按 CNOT 密度排序

$A_{\text{map}} \leftarrow \varnothing$; $Q_{\text{used}} \leftarrow \varnothing$;//初始化分配方案和已用比特集合

$P_{\text{cur}} \leftarrow P$;//当前可用分区集合

for each circuit c in Q_{prio} do//遍历优先级队列

  success ← False;

  $t \leftarrow 1$;

  while ($t \leqslant t_{\max}$) ∧ (success = False) do

    (p*, L*, s*) ← (null, null, −∞);//当前最佳分配

    for each partition p in Pcur do//遍历所有分区

      $Q_{\text{avail}} \leftarrow p - Q_{\text{used}}$;//当前分区可用比特

      if $|Q_{\text{avail}}| \geqslant |c|$ ∧ IsConn(G_{f}, Q_{avail}) then//检查是否满足约束条件

        $G_{\text{map}} \leftarrow$ GetGraph(p, G_{f}, G_{full});

        (L, s) ← MappingAlgorithm(c, p, G_{map});//调用映射算法

        if $s > s$* then

          (p*, L*, s*) ← (p, L, s);//更新最佳分配

        end if

      end if

    end for

    if L* ≠ null then//成功找到分配方案

      $A_{\text{map}}[c] \leftarrow (L^*, p^*)$;//记录分配结果

      $Q_{\text{used}} \leftarrow Q_{\text{used}} \cup$ values(L*);//更新集合

      success ← True;//标记成功并退出尝试循环

    else if $t < t_{\max}$ then//非最后一次尝试时合并分区

      $P_{\text{new}} \leftarrow$ MergePartitions(c, P_{cur}, Q_{used}, G_{full});

      if P_{new} ≠ null then//合并成功则更新分区集合

        $P_{\text{cur}} \leftarrow P_{\text{new}}$;

      else

        $t \leftarrow t_{\max}$;//无法合并则提前终止

      end if

    end if

    $t \leftarrow t + 1$;

  end while

end for

return A_{map}//返回最终分配方案

算法 2 的时间复杂度主要分为排序和映射两个阶段。排序阶段复杂度为$O(|C|\log|C|)$，其中

$|C|$表示量子线路的数量。映射阶段需对每条线路尝试$t_{\max}$次，每次枚举所有候选分区$|P|$并执行映射评估，该过程复杂度为$O(t_{\max}|C||P|f(|c|,|V|))$，其中$|c|$为线路规模，$|V|$为物理比特数，$f(|c|,|V|)$为单次映射的复杂度，故总时间复杂度为$O(|C|\log|C|+t_{\max}|C||P|f(|c|,|V|))$。空间复杂度为$O(|V|+|C|)$，用于存储分区、已分配比特及映射结果。

示例 3　经过动态分区算法的迭代优化后，最终的分区方案如图 5 所示，得到了包含 5 个独立分区的集合$P=\{p_{1_A1},p_{1_A2},p_2,p_3,p_4\}$。各分区的规模分别为：$|p_{1_A1}|=6$，$|p_{1_A2}|=6$，$|p_2|=1$，$|p_3|=1$，$|p_4|=10$。

当前需将包含两条逻辑线路的任务集$C_{\text{set}}=\{C_A,C_B\}$分配到分区上。其中线路$C_A$规模为 12 量子比特，CNOT 密度高。$C_B$规模为 8 量子比特，CNOT 密度低，如图 6 所示。算法执行流程如下。

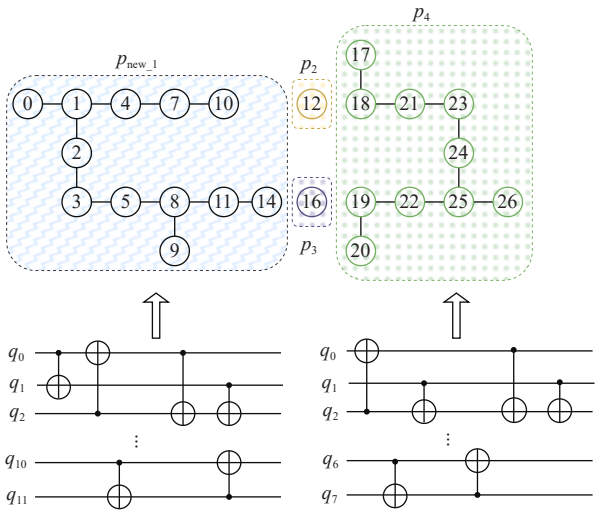


图 6　资源分配示意图

1）优先级调度：根据 CNOT 密度，确定调度队列为$[C_A,C_B]$。因此，首先为线路C_A分配资源。

2）为线路C_A首次尝试分配：检查所有可用分区。发现规模最大的分区p_4，其规模为 10，依然无法满足 C_A 的需求。首次分配尝试失败。

3）按需动态合并：检查是否存在多个未被占用且相邻的分区。检测到分区p_{1_A1}和p_{1_A2}在原始拓扑中是通过边A连接，并且处于未被占用状态。算法将它们合并为新分区$p_{\text{new_1}}$，其总规模为 12 个量子比特。分区集合更新为$\{p_{\text{new_1}},p_2,p_3,p_4\}$。

4）为线路C_A二次分配：在新分区集合上为C_A再次执行匹配。$p_{\text{new_1}}$规模为 12，满足C_A所需规模大小。接着调用映射算法，为C_A在$p_{\text{new_1}}$中找出最优的映射方案。p_2、p_3、p_4均被视为可用分区，但评估后发现只有$p_{\text{new_1}}$的规模满足要求，于是采纳此方案。线路C_A分配到新分区$p_{\text{new_1}}$上，并记录占用的物理比特。

5）为线路C_B分配：检查剩余未占用分区。此时，未被占用分区为$\{p_2,p_3,p_4\}$。其中，p_2、p_3大小为 1，不满足要求，而p_4大小为 10，满足 C_B 所需规模大小。接着调用映射算法，为C_B在p_4中找到最优的映射方案。

6）所有线路均成功分配。最终分配方案为：C_A映射到分区$p_{\text{new_1}}$；C_B映射到分区p_4。

3　自适应映射策略

将逻辑线路分配到物理分区后，下一步需确定映射方案。高质量的映射可以在路由时最小化 SWAP 门的开销，并充分利用量子设备的高保真度量子比特，从而影响执行保真度。因此本文设计了自适应映射策略。此策略根据线路的逻辑比特数，在确保最优解的精确方法与高效求解的启发式方法之间进行选择。

3.1　基于代价函数最优的映射

对于逻辑比特数不超过 4 的线路，其所有可能映射可在有限时间内完成遍历。因此，本文提出基于代价函数的精确求解策略，以确保全局最优解。线路的逻辑比特数n基于对映射方案枚举复杂度分析设定：逻辑比特数$n\leqslant 4$的线路可通过全排列枚举快速完成求解，兼具求解质量与计算可行性。

为了有效指导该过程，本文构建了代价函数$C(M)$来评估每个潜在映射方案M的综合质量：

$$C(M)=w_1C_{\text{SWAP}}(M)+w_2C_f(M) \tag{7}$$

式中，$C_{\text{SWAP}}(M)$代表计算所需的 SWAP 门总数；$C_f(M)$代表最大化执行保真度，由节点质量（映射中所有物理比特的综合保真度之和）与边质量（映射中 CNOT 门利用的边的权重之和）构成；w_1和w_2是权重系数，用于在评估映射方案时进行权衡取舍。

设线路的逻辑比特数为n，目标物理分区的比特数为N。该方法需要从N个物理比特中选出n个进行全排列。因此，总时间复杂度为$O(N!/(N-n)!)$。复杂度随问题规模呈阶乘级增长，计算成本急剧上升，使得该策略仅在处理$n\leqslant 4$的线路时具备可行性，体现了其应用规模的内在局限性。

3.2　基于贪心和模拟退火算法的映射

对于更复杂的线路，本文采用基于模拟退火

（simulated annealing, SA）的启发式方法进行映射。为加速 SA 算法的收敛并提升最终解的质量，其起点并非随机生成，而是通过交互感知的贪心启发式方法构建。

3.2.1 交互感知的贪心启发式初始布局

该方法旨在将对线路影响最大的核心逻辑比特优先分配到物理分区中的核心位置。其构建过程如下。

1）识别核心逻辑量子比特：通过分析线路交互图，计算每个逻辑比特的交互总次数与节点度。根据这两个指标对逻辑比特进行降序排序，生成优先级列表。

2）确定核心物理量子比特：在目标物理分区内选择种子物理比特。该比特需满足分区内度数最高，若存在多个，则选择综合保真度最高的比特。

3）分层贪心扩展：将优先级最高的逻辑比特分配到种子物理量子比特上，然后根据优先级列表，采用两层扩展策略迭代分配剩余逻辑比特。第 1 层，在种子比特的未占用的邻居比特中选择综合保真度最高的物理比特；第 2 层，若邻居比特均被占用，则扩展搜索至已分配区域中未占用的邻居比特，并从中选择综合保真度最高的物理比特。

利用分层贪心策略生成的初始布局不仅在拓扑上是紧凑的，而且占据了分区内最高质量的物理资源区域。

3.2.2 动态加权的模拟退火全局优化

贪心方法生成的初始布局质量虽高，但可能为局部最优解。为了在更广阔的解空间中探索，下面详细介绍本文采用模拟退火进行全局优化的方法。

1）多目标代价函数：SA 算法的代价函数同时优化距离成本和保真度成本。距离成本代表逻辑比特在架构拓扑图上的距离，旨在减少 SWAP 门；保真度成本基于所选物理比特的综合质量，旨在提高执行保真度。

2）动态自适应权重：在退火的不同阶段，上述两个成本的相对权重随进程动态调整。初始距离权重与保真度权重分别根据式 (8) 和式 (9) 设定：

$$w_d^{(0)} = 0.6 - 0.2\min(\text{cx_density}, 1.0) \tag{8}$$

$$w_f^{(0)} = 1 - w_d^{(0)} \tag{9}$$

式中，cx_density表示线路中的 CNOT 门密度。在退火过程中，权重随温度区间动态调整：当温度 $T > 0.7T_0$ 时，保持初始权重 $w_d = w_d^{(0)}$；当 $0.3T_0 < T \leqslant 0.7T_0$ 时，提高距离权重至 $w_d = w_d^{(0)} + 0.1$ 以找到拓扑结构合理、SWAP 开销低的映射方案；当 $T \leqslant 0.3T_0$ 时，降低距离权重至 $w_d = w_d^{(0)} - 0.1$，保真度成本权重增加，算法在已基本成型的映射方案上进行局部优化，将比特微调至更高质量的物理位置。此外，该权重调整机制还具备实时自适应能力，若在迭代中发现某一成本相较于历史最优解显著恶化，其权重会临时放大，以指导算法纠偏。

本文通过自适应策略，对给定的线路进行映射。首先，根据线路的逻辑比特数 n 进行分流：对于 $n \leqslant 4$ 的线路，采用精确求解策略；对于 $n > 4$ 的线路，使用启发式优化流程。第 1 阶段，通过匹配线路的核心逻辑比特与分区的核心物理比特，快速生成结构紧凑的高质量初始解。第 2 阶段，以该初始解为起点进行模拟退火迭代优化，初期优先减少 SWAP 开销，后期转向提高保真度，并通过 Metropolis 准则概率性地接受次优解以跳出局部最优，直至收敛。此外，本文采用与 SABRE[16] 相同的路由方法。

实现该映射策略的伪代码算法如下。

算法 3: 自适应映射算法

Input: C, P, G', THRESH//输入待映射量子线路集合、可用分区集合、处理后的拓扑图、线路阈值

Output: MapRes//输出映射结果

MapRes ← Ø;//初始化映射结果集合

for each circuit c in C do

  n ← num_qubits(c);

  if n ⩽ THRES then//处理逻辑比特数不超过 4 的线路

    L_{best} ← ExhaustiveSearch(c, P, G);//精确求解逻辑比特数不超过 4 的线路

    MapRse[c] ← L_{best};

  else

    Linit ← GreedySeed (c, P, G);//生成初始布局

    if Linit = null then

      MapRes[c] ← False;

    else

      L_{cur} ← Linit;//当前解设为初始解

      L_{best} ←Linit;//最优解设为初始解

      T ← T_0;//初始化温度

      while TerminationCondition(T, L_{best}) = False do//未达终止条件时循环

        for i=1 to ITER do//当前温度下迭代

          w ← GetWeights(T);//获取当前温度下的

权重

$L_{new} \leftarrow$ Neighbor(L_{cur}, P, G);//生成新候选解

$\Delta E \leftarrow$ Cost(L_{new}, w) − Cost(L_{cur}, w);//计算能量差

if ΔE <0 or Random() < exp(−$\Delta E/T$) then//Metropolis 准则判断

$L_{cur} \leftarrow L_{new}$;//接受新解

if Cost(L_{cur}) < Cost(L_{best}) then

$L_{best} \leftarrow L_{cur}$;

end if

end if

end for

$T \leftarrow$ cooling(T);//降温操作

end while

if not IsValid(L_{best}, G) then//验证拓扑有效性

MapRes[c] ← False;

else

MapRes[c] ← Lbest;

end if

end if

end if

end for

return MapRes//返回映射结果

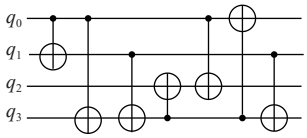
算法 3 的时间复杂度主要由模拟退火过程决定。单条线路的初始布局生成耗时$O(n)$，其中n为线路的量子比特数。退火过程中每轮温度迭代需进行ITER次邻域操作和代价评估，每次复杂度为$O(n)$，因此单线路总时间为$O(\text{ITER}\times Kn)$，其中K表示温度迭代次数。对于$|C|$条线路，整体时间复杂度为$O(|C|\text{ITER}\times Kn)$。空间复杂度为$O(n)$，用于存储各种布局状态。

示例 4　将图 6 中包含 12 个逻辑比特的线路C_A映射到p_{new_1}上。由于线路的逻辑比特数较大，算法将自动选择基于贪心和模拟退火的启发式策略。

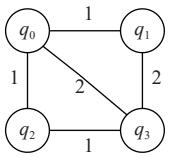
1）分析线路C_A，发现逻辑比特q_1,q_2,q_6,q_7的交互最为频繁，构成核心逻辑量子比特。如图 7b 为图 7a 量子线路中量子比特之间的交互图，可以确定q_3交互最频繁。

2）在分区p_{new_1}中，物理比特Q_4及其邻居因其高连通度和高保真度被识别为核心物理量子比特。

3）将q_1映射到Q_4，然后依据逻辑交互关系和保真度，逐层有序地将其余逻辑比特映射到Q_4的邻居比特，形成初始布局M_0。



a. 任意量子线路



b. 量子比特交互图

图 7　核心逻辑量子比特示意图

4）将M_0作为模拟退火的起点，通过随机交换映射、移动映射等操作探索邻域解。退火初期，优先接受能减少 SWAP 门的解；退火后期，则更倾向于接受能提升保真度的解。

5）经过多轮迭代降温后，得到在距离成本和保真度成本之间达到平衡的映射方案。

4　实验结果及结果分析

为了验证多量子线路自适应映射方法的有效性，本文进行了一系列实验，旨在评估在多线路并行执行时的保真度和 SWAP 门数，包括基准线路集、性能评估指标以及对比方法等。

4.1　实验设置

受 IBM Qiskit 运行时服务区域访问限制[19]，本文借助 Qiskit 的 QASM 模拟器[28]，在 27 量子位的 IBMQ Toronto 上进行模拟。此外，为验证本文方法的扩展性，在 156 量子位的 IBMQ Marrakesh 上开展了实验。实验过程中，模拟器的校准数据保持一致。本文参照文献 [29] 的技术手段，模拟因串扰引发的条件误差。该模拟器不仅使用真实量子位的拓扑与校准数据，还随机生成串扰放大率。

4.1.1　基准测试集

本文评估采用 Revlib[30] 和 QASMBench[31] 公开基准库中的线路，如表 2 所示。所选线路涵盖多种类型，包括 Bernstein-Vazirani 算法、算术函数等，其量子比特数在 3～17，与实验所用设备规模适配。线路中 CNOT 门数为 2～336，便于评估不同交互强度下的映射性能。

4.1.2　性能评估指标

1）试验成功概率（probability of a successful trial, PST）：PST 是评估量子线路执行保真度的重要指标，定义为产生正确结果的试验次数与总试验次数之比。为准确测量 PST，本文参照文献 [6] 的实验设计，在目标量子设备上完成每条线路的映射与运行，并进行 8 192 次重复试验。

表 2 基准线路集

ID	线路	量子比特数	CNOT门数
1	bv_n3	3	2
2	bv_n4	4	3
3	peres_3	3	7
4	toffoli_n3	3	6
5	fredkin_n3	3	8
6	xor5_254	6	5
7	3_17_13	3	17
8	4mod5-v1_22	5	11
9	mod5mils_65	5	16
10	alu-v0_27	5	17
11	decod24-v2_43	4	22
12	4gt13_92	5	30
13	aj-e11_165	5	69
14	4gt4-v0_72	6	113
15	sf_276	6	336
16	ham7_104	7	149
17	C17_204	7	205
18	ex2_227	7	275
19	dnn_n8	8	192
20	ising_model_10	10	90
21	qft_10	10	90
22	sys6-v0_111	10	98
23	sym9_146	12	148
24	rd53_311	13	124
25	hwb4_49	16	107
26	decod24-enable_126	16	149
27	cnt3-5_180	16	215
28	qft_16	16	240
29	cm82a_208	16	283
30	qec9xz_n17	17	32

2）映射后 CNOT 门数量：评估降低 SWAP 开销方面的能力。本文参照文献 [11] 的方法，通过统计映射后 CNOT 门的数量，来量化不同方法在减少 SWAP 开销方面的效果。

4.1.3 对比方法

为了全面评估本文方法，选取了以下具有代表性的多量子线路映射方案进行对比，并采用与 SABRE 相同的映射转换方法。

1）基线[6]：设计了一种多量子线路映射算法，旨在降低映射对保真度的影响。

2）SABRE[16]：通过反向遍历技术生成初始映射，利用量子线路的可逆性，将正向遍历的最终映射作为反向遍历的初始映射，以此优化初始映射质量。在映射中，将多个并发量子线路合并为单线路后进行处理。

3）QuMC[27]：初始映射通过两种量子比特分区算法实现。GSP 算法遍历架构拓扑图寻找最优分区，QHSP 算法则优先选择高保真度量子比特构建分区。

4）CDAP[8]：基于 FN 社区检测算法构建层次树，依据架构拓扑图和校准数据，将物理量子比特划分为紧密相连的社区，优先合并高可靠性的量子比特，并按 CNOT 密度对线路排序和分配社区，以此生成初始映射。

4.1.4 参数设置

本文方法涉及多类参数，其设定基于算法目标与实验对比分析。

1） 节点综合保真度权重：式 (2) 中，各个维度的权重分别设为$\alpha=0.1$，$\beta=0.1$，$\gamma=0.8$。该配置旨在突出拓扑连通性（$\gamma=0.8$）对降低路由开销的关键作用，并弱化局部噪声对整体评估的过度影响，以提高评分的稳健性。

2） 分区总体质量权重：式 (4) 中，$a=0.6$，$b=0.4$，强调分区内部凝聚力以确保映射可行性，同时保留适当的分区间隔离权重以抑制并行串扰。

3） 映射代价权重策略：式 (7) 中的权重w_1与w_2根据线路的逻辑比特数自适应设定。对于逻辑比特数$n\leqslant 4$的线路，采用固定权重（$w_1=0.5$，$w_2=0.5$），在可遍历的解空间中平衡 SWAP 开销与保真度。对于$n>4$的线路，采用动态权重机制，其初始值根据线路的 CNOT 密度计算，见式 (8) 和式 (9)，并在模拟退火过程中随温度分段调整。

4） 其他关键参数：预处理架构拓扑选用量子比特与连接错误率的第 90% 作为阈值，以过滤错误率最高的 10% 的部分；模拟退火超参数（初始温度$T_0=10$，降温系数为 0.95，每温度迭代次数为 100）经小范围扰乱实验发现（初温 5～20、降温系数 0.9～0.99、迭代次数 50～200），在不显著增加运行时间的前提下，保真度和额外插入 CNOT 门数的相对趋势保持稳定，因此最终确定当前参数组合。

4.2 实验结果及分析

本文选择表 2 中的两条线路进行实验，测得其保真度和 SWAP 门开销。

图 8 展示了两条简单线路（ID 1～6）并行执行的性能对比，柱状图量化两者平均保真度的差异，折线图表征额外增加的 CNOT 门数量。在此场景下，Baseline、SABRE、CDAP 和 QuMC 的平均 PST 值分别为 70.91%、75.79%、78.55% 和 79.77%，而本文方法的平均保真度高达 83.06%，分别比 Baseline、SABRE、CDAP 和 QuMC 高出

了 12.15%、7.27%、4.51% 和 3.29%，相较于次优的 CDAP 和 QuMC 取得了提升。

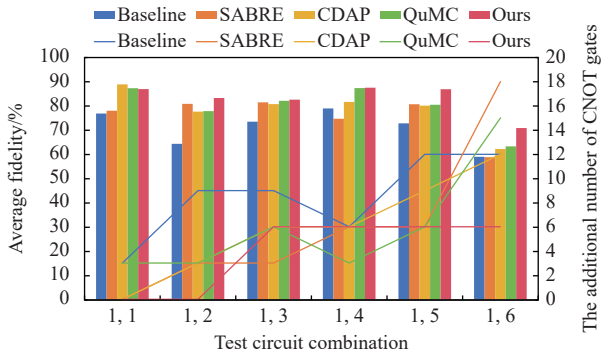


图 8　ID 1～6 两条线路的平均保真度和 CNOT 门数对比

在统计额外增加的 CNOT 门数量时，Baseline、SABRE、CDAP 和 QuMC 的平均额外增加 CNOT 门数分别为 8.5（Baseline）和 6（其余三者），而本文方法仅为 4，较 Baseline 减少 4.5，较 SABRE、CDAP 和 QuMC 减少 2。

这种优势主要归功于对逻辑比特数不超过 4 的线路设计的精确求解策略（详见 3.1 节），该策略能够在可控的计算时间内遍历所有可能的映射方案，确保找到全局最优解。

最优搜索是在经过高保真度子图筛选（详见 2.1 节）后的拓扑图上进行。这意味着，最优解不仅在拓扑上最佳，而且所利用的物理量子比特也是拓扑图中最可靠的部分。

图 9 展示了复杂线路（ID 7～12）并行执行时的性能对比，测量了平均保真度的变化，并记录了额外增加的 CNOT 门数量。在此场景下，所有方法的保真度均显著下降，Baseline、SABRE、CDAP 和 QuMC 的平均 PST 值分别为 18.63%、16.45%、19.81% 和 20.68%。尽管如此，本文方法仍以 21.73% 的平均保真度保持领先，比 Baseline、SABRE、CDAP 和 QuMC 分别高出 3.1%、5.28%、1.92% 和 1.05%。

在统计额外增加的 CNOT 门数时，所有方法的 SWAP 开销均显著上升，Baseline、SABRE、CDAP 和 QuMC 平均额外增加的 CNOT 门数分别为 54、49.5、53.5 和 53.5，而本文方法仅为 48.5，保持较低的 SWAP 开销，相较于 Baseline 减少 5.5，比 SABRE 减少 1，较 CDAP 和 QuMC 减少 5。

在复杂线路并行执行的情况下，本文中交互感知的贪心启发式初始布局（详见 3.2.1 节）发挥了关键作用。通过识别交互频繁的逻辑量子比特，并优先将其映射到分区中度最大的物理量子比特上，为后续提供了优于随机或顺序布局的起点。这直接提升了后续优化过程的起点，更易于收敛到高质量解。

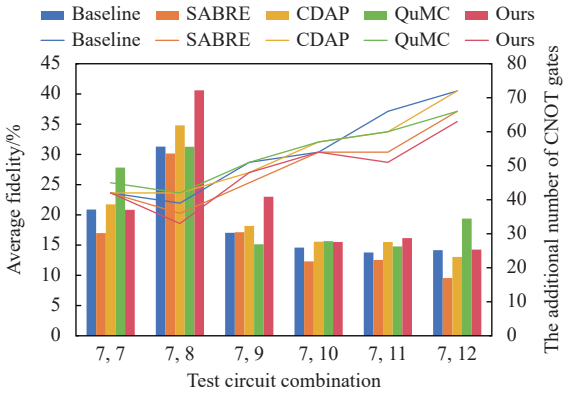


图 9　ID 7～12 两条线路的平均保真度和 CNOT 门数对比

动态加权的模拟退火全局优化（详见 3.2.2 节）中的动态权重机制使得方法在优化初期专注于降低 SWAP 开销，而在后期专注于提升保真度。这种优化策略比单一目标的启发式方法更为有效。

此外，拓扑感知的动态分区（详见 2.2 节）与多线路资源分配（详见 2.3 节）为处理复杂任务提供了坚实基础。前者确保复杂线路被分配到内部连通性好的分区；后者使得复杂任务能够优先获得最优的硬件资源。

下面详细分析本文方法中不同技术策略对整体性能的对比结果，所有数据均呈现在表 3 中，noPre 的目的是验证预处理模块的必要性；noPartition 旨在验证动态分区算法的优越性；noSA 的目的是验证 SA 算法优化过程的有效性。PST1 和 PST2 分别代表并行执行的两条线路的试验成功概率，反映各线路的执行保真度；Addi 表示映射过程中平均额外插入的 CNOT 门数，用于衡量 SWAP 操作引入的路由开销。

预处理模块（详见 2.1 节）通过移除架构拓扑图中高错误率的量子比特和物理耦合边，旨在为后续提供高保真度的子图。与完整方案（本文，平均 PST 为 52.39%）相比，noPre 方案的平均 PST 下降至 48.48%，降低了 3.91%。同时，平均额外增加的 CNOT 门数从 26.25 增加到 29。这种性能下降是符合预期的。在缺少预处理步骤时，分区和映射只能在含噪声的拓扑图上执行，使得线路映射到高错误率区域。这直接导致门操作错误率的增加，最终体现为 PST 的降低。这证明预处理模块对于保证方法性能下限起到了关键作用。

表 3 不同技术策略方法的性能对比结果

ID	本文方法			noPre			noPartition			noSA		
	PST1/%	PST2/%	Addi/个	PST1/%	PST2/%	Addi/个	PST1/%	PST2/%	Addi/个	PST1/%	PST2/%	Addi/个
1、1	87.99	85.96	0	85.63	85.86	0	85.45	87.57	0	84.85	88.11	0
1、2	85.93	80.69	0	85.72	81.14	0	85.16	80.47	0	84.33	80.88	0
1、3	87.79	77.47	6	84.13	69.92	6	85.79	72.74	6	84.53	72.69	6
1、4	87.84	87.34	6	85.83	86.79	6	85.61	84.97	6	84.24	83.46	6
1、5	87.27	86.65	6	85.73	65.77	9	85.35	72.57	6	84.62	72.3	6
1、6	87.87	53.87	6	86.19	37.96	9	85.08	43.3	6	87.55	47.4	6
7、7	22.52	19.18	42	17.07	22.34	45	17.44	22.99	48	18.44	20.19	48
7、8	23.21	58.04	33	17.11	54.96	33	16.55	61.96	36	17.75	44.24	45
7、9	22.57	23.44	48	15.86	15.99	51	17.7	25.11	51	17.69	16.7	48
7、10	22.73	8.28	54	17	7.98	57	18.63	12.13	54	18.05	10.08	54
7、11	22.06	10.23	51	15.69	12.55	63	18.4	11.43	54	17.1	10.71	60
7、12	23.55	4.97	63	22.23	3.98	69	20.86	4.75	72	17.75	6.58	84
Avg	52.39		26.25	48.48		29	50.08		28.25	48.76		30.25

拓扑感知的动态分区（详见 2.2 节）旨在根据架构拓扑图的综合质量指标和待执行线路的规模，将高保真度的区域划分为多个大小合适、内部拓扑优越的独立分区。noPartition 方案使用了仅基于物理连通性的简单分区方法，其平均 PST 为 50.08%，相比完整方案下降了 2.31%，平均额外增加的 CNOT 门数增加到 28.25。noPartition 的基础分区方法无法感知分区的质量。在大多数情况下，它会将所有连通的量子比特划分为单一的大分区。这会导致任务在共享资源池中串行竞争，并将本可物理隔离的并行线路紧密部署，加剧了串扰。因此，本文的动态分区策略在实现高效并行和抑制串扰方面具有重要作用。

本文映射方法的核心是模拟退火优化过程（详见 3.2.2）。它旨在将贪心启发式生成的初始布局进行迭代优化。noSA 方案移除了模拟退火优化，仅依赖贪心布局。结果显示其性能下降显著，平均 PST 下降至 48.76%，减少了 3.63%，平均额外增加的 CNOT 门数从 26.25 增加至 30.25。该结果揭示了初始映射质量与最终执行保真度之间的强相关性。尽管本文的贪心策略能够提供合理的初始布局，但它极易陷入局部最优，导致大量逻辑上邻近的 CNOT 门必须通过插入额外的 SWAP 门来满足近邻约束。这直接导致了 CNOT 数大幅增加，不仅增加了量子线路的深度，也增加了线路整体出错的概率。因此，noSA 方案中 PST 显著下降的根本原因在于 SWAP 开销的急剧膨胀。这证明了模拟退火优化过程是实现低 SWAP 开销和高保真度的关键技术。

此外，为了验证本文方法在更大规模量子设备上的有效性，选取表 2 中 ID 为 13～30 中的任意 5 条较大规模线路，在 156 量子位的 IBMQ Marrakesh 上开展了实验，结果见表 4。本文方法的平均保真度为 20.96%，相较于 Baseline、SABRE、QuMC、CDAP 仍保持 4.37%～6.93% 的优势；同时，平均额外增加 CNOT 门数低于其他方法。

表 4 ID 13-30 5 条线路的平均保真度和 CNOT 门数对比

ID	Baseline		SABRE		QuMC		CDAP		本文方法	
	PST/%	Addi/个	PST/%	Addi/个	PST/%	Addi/个	PST/%	Addi/个	PST/%	Addi/个
13、17、18、21、22	15.33	1746	20.72	1674	16.52	1884	24.08	1991	27.81	1674
13、14、16、29、30	21.94	1578	19.81	2106	21.17	1794	24.73	2091	31.78	1728
14、15、21、25、30	19.2	1527	12.8	1971	22.65	1722	29.51	2027	30.62	1518
14、16、20、25、30	17.39	1098	17.95	2115	21.79	1086	20.47	1441	26.52	1017
15、16、22、25、30	14.44	1716	16.69	3099	20.36	1953	19.47	2392	27.48	1710
15、19、21、22、28	12.97	1806	12.54	1927	15.32	1866	13.15	2267	11.40	1911
15、19、21、26、27	11.21	1857	11.59	2184	16.16	1965	12.03	2419	12.45	1803
15、21、23、26、29	13.42	2331	16.07	2247	15.23	2550	10.77	2920	17.17	2181
16、18、20、23、29	6.98	2205	7.6	2679	6.87	2382	5.26	2613	11.85	2085

表 4（续）

ID	Baseline		SABRE		QuMC		CDAP		本文方法	
	PST/%	Addi/个	PST/%	Addi/个	PST/%	Addi/个	PST/%	Addi/个	PST/%	Addi/个
18、20、24、28、30	5.5	1881	7.45	2082	7.39	2121	8.71	2545	12.26	1920
19、22、26、27、30	11.69	1425	10.25	2832	10.99	1761	15.01	2008	23.55	1290
20、21、23、24、25	18.27	1224	16.44	1533	18.31	1128	15.86	1269	18.64	1083
Avg	14.03	1699.5	14.16	2204.08	16.06	1851	16.59	2165.25	20.96	1660

实验结果表明本文方法在更大规模的不同拓扑结构下仍能保持稳定的保真度与较低的 SWAP 开销，证明本文方法对不同拓扑具备适应性以及扩展性。

通过一系列的对比分析，验证了所提出的方法中各个技术策略的有效性。实验结果表明，预处理模块为方法提供了可靠的基础；动态分区策略是实现高效并行与串扰抑制的关键；而自适应映射策略在最小化路由开销、保证最终执行保真度方面起到了决定性的作用。得益于这三者的协同作用，本文方法在各项关键性能指标上，能够达到较为理想的水平。

5　结束语

本文研究了 NISQ 时代下多量子线路映射问题，提出了基于动态分区的多量子线路自适应映射，其主要贡献如下。

1）提出了高保真度子图筛选策略。该策略在映射前过滤掉高错误率的量子比特和物理连接，为后续构建高保真度的子拓扑图。

2）设计了动态分区方法。该方法由综合评估分区拓扑与保真度的质量函数指导，生成物理特性优越且能灵活合并以适应不同线路的分区。

3）开发了自适应映射策略。该策略能够识别线路中量子比特间的交互，并将其与分区中拓扑和保真度最佳的区域进行匹配，从而生成高质量初始解，后续采用模拟退火进行全局优化，使其在保真度与 SWAP 开销间寻求最佳平衡。

未来研究可将初始映射策略与具备噪声感知和并行感知能力的路由算法相结合，实现映射流程的全程优化，并在更大规模和多样化拓扑结构的量子设备上进行测试和扩展。

参 考 文 献

[1] Preskill J. Quantum computing 40 years later[M]//Hey T. Feynman lectures on computation. Boca Raton: CRC Press, 2023: 193-244.

[2] Preskill J. Quantum computing in the NISQ era and beyond[J]. Quantum, 2018, 2: 79.

[3] IBM. IBM unveils 400 qubit-plus quantum processor and next-generation IBM quantum system two[EB/OL]. (2022-11-09)[2025-07-03]. https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two.

[4] 张辰逸, 尚涛, 刘建伟. 基于交换门的前瞻启发式量子线路映射算法[J]. 电子科技大学学报, 2023, 52(4): 489-497.
Zhang Chenyi, Shang Tao, Liu Jianwei. SWAP-based prospective heuristic quantum circuit mapping algorithm[J]. Journal of University of Electronic Science and Technology of China, 2023, 52(4): 489-497. (in Chinese)

[5] Tan Bochen, Cong J. Optimal layout synthesis for quantum computing[C]//Proceedings of the 2020 IEEE/ACM International Conference on Computer-Aided Design. San Diego: IEEE, 2020: 1-9.

[6] Das P, Tannu S S, Nair P J, et al. A case for multi-programming quantum computers[C]//Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture. Columbus: IEEE, 2019: 291-303.

[7] Liu Lei, Dou Xinglei. QuCloud: A new qubit mapping mechanism for multi-programming quantum computing in cloud environment[C]//2021 IEEE International Symposium on High-Performance Computer Architecture. Seoul: IEEE, 2021: 167-178.

[8] Liu Lei, Dou Xinglei. QuCloud+: A holistic qubit mapping scheme for single/multi-programming on 2D/3D NISQ quantum computers[J]. ACM Transactions on Architecture and Code Optimization, 2024, 21(1): 9.

[9] Ohkura Y, Satoh T, Van Meter R. Simultaneous execution of quantum circuits on current and near-future NISQ systems[J]. IEEE Transactions on Quantum Engineering, 2022, 3: 2500210.

[10] Harper B, Tonekaboni B, Goldozian B, et al. Crosstalk attacks and defence in a shared quantum computing environment[J]. Advanced Quantum Technologies, 2025, 8(10): e2500009.

[11] Upadhyay S, Ghosh S. SHARE: Secure hardware allocation and resource efficiency in quantum systems [C]//2024 IEEE International Conference on Quantum Computing and Engineering. Montreal: IEEE, 2024: 1109-1119.

[12] Lao Lingling, van Someren H, Ashraf I, et al. Timing and resource-aware mapping of quantum circuits to superconducting processors[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2022, 41(2): 359-371.

[13] Tan Bochen, Cong J. Optimal qubit mapping with simultaneous gate absorption[C]//2021 IEEE/ACM

International Conference on Computer Aided Design. Munich: IEEE, 2021: 1-8.
[14] Nannicini G, Bishop L S, Günlük O, et al. Optimal qubit assignment and routing via integer programming[J]. ACM Transactions on Quantum Computing, 2023, 4(1): 7.
[15] Molavi A, Xu A, Diges M, et al. Qubit mapping and routing via MaxSAT[C]//2022 55th IEEE/ACM International Symposium on Microarchitecture. Chicago: IEEE, 2022: 1078-1091.
[16] Li Gushu, Ding Yufei, Xie Yuan. Tackling the qubit mapping problem for NISQ-era quantum devices[C]// Proceedings of the 24th International Conference on Architectural Support for Programming Languages and Operating Systems. Providence: ACM, 2019.
[17] Liu Ji, Li Peiyi, Zhou Huiyang. Not all SWAPs have the same cost: A case for optimization-aware qubit routing[C]//2022 IEEE International Symposium on High-Performance Computer Architecture. Seoul: IEEE, 2022: 709-725.
[18] Gheorghiu V, Huang Jiaxin, Li S M, et al. Reducing the CNOT count for Clifford+T circuits on NISQ architectures[J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2023, 42(6): 1873-1884.
[19] IBM. IBM quantum platform[EB/OL]. [2025-07-03]. https://quantum.ibm.com/.
[20] Iverson J K, Preskill J. Coherence in logical quantum channels[J]. New Journal of Physics, 2020, 22(7): 073066.
[21] Magesan E, Gambetta J M, Johnson B R, et al. Efficient measurement of quantum gate error by interleaved randomized benchmarking[J]. Physical Review Letters, 2012, 109(8): 080505.
[22] Nation P D, Kang H, Sundaresan N, et al. Scalable mitigation of measurement errors on quantum computers[J]. PRX Quantum, 2021, 2(4): 040326.
[23] Niu Siyuan, Todri-Sanial A. Analyzing crosstalk error in the NISQ era[C]//2021 IEEE Computer Society Annual Symposium on VLSI. Tampa: IEEE, 2021: 428-430.
[24] Gambetta J M, Córcoles A D, Merkel S T, et al. Characterization of addressability by simultaneous randomized benchmarking[J]. Physical Review Letters, 2012, 109(24): 240504.
[25] Kurniawan H, Rodríguez-Soriano L, Cuomo D, et al. On the use of calibration data in error-aware compilation techniques for NISQ devices[C]//2024 IEEE International Conference on Quantum Computing and Engineering. Montreal: IEEE, 2024: 338-348.
[26] Itoko T, Raymond R, Imamichi T, et al. Optimization of quantum circuit mapping using gate transformation and commutation[J]. Integration, 2020, 70: 43-50.
[27] Niu Siyuan, Todri-Sanial A. Enabling multi-programming mechanism for quantum computing in the NISQ era[J]. Quantum, 2023, 7: 925.
[28] IBM. Qiskit: An open-source framework for quantum computing[EB/OL]. [2025-07-03]. https://www.ibm.com/quantum/qiskit.
[29] Ash-Saki A, Alam M, Ghosh S. Experimental characterization, modeling, and analysis of crosstalk in a quantum computer[J]. IEEE Transactions on Quantum Engineering, 2020, 1: 3101406.
[30] RevLib. An online resource for reversible benchmarks[EB/OL]. [2025-07-03]. https://www.revlib.org/.
[31] QASMBench Benchmark Suite. A low-level OpenQASM benchmark suite for NISQ evaluation and simulation[EB/OL]. [2025-07-03]. https://github.com/pnnl/QASMBench/tree/master.

责任编辑 税 红