面向分布式量子计算的模块化量子编译框架

Davide Ferrari[1],*、Stefano Carretta[2],和 Michele Amoretti[1]

[1]量子软件实验室，帕尔马大学工程与建筑系，帕尔马，43124，意大利（电子邮件：michele.amoretti@unipr.it, davide.ferrari1@unipr.it。网站：http://www.qis.unipr.it/quantumsoftware.html[2]帕尔马大学数学、物理与计算机科学系，意大利帕尔马 I-43124，以及意大利国家核物理研究所米兰-比可卡分部帕尔马关联组，43124，意大利*通讯作者：Davide Ferrari，davide.ferrari1@unipr.it。

arXiv:2305.02969v1 [quant-ph] 4 May 2023

摘要

对于大多数实际应用，量子算法所需的量子比特数量等资源规模很大，远超当前 NISQ 处理器所能提供的资源。借助量子互联网提供的网络和通信功能，分布式量子计算（DQC）被视为一种可扩展的方法，可增加用于计算任务的量子比特数量。要使 DQC 高效有效，量子编译器必须为量子算法找到最佳划分方案，并通过智能的远程操作调度来优化 EPR 对的消耗。同时，量子编译器还应为每个分区找到最佳的局部变换。本文提出了一种用于 DQC 的模块化量子编译框架，同时考虑网络和设备的约束与特性。我们基于所提出的框架实现并测试了一种量子编译器，使用了一些值得关注的电路，例如变分量子本征求解器（VQE）和量子傅里叶变换（QFT）电路；所考虑的网络拓扑各不相同，量子处理器则采用重六边形耦合图。我们还设计了一种远程调度策略，可利用 TeleGate 和 TeleData 操作，并测试了仅使用 TeleGate 或同时使用两者的影响。评估结果表明，TeleData 操作可能有助于减少 EPR 对的消耗，而选择连接性更强的网络拓扑则有助于减少用于远程操作的层数。

索引词—— 分布式量子计算、量子编译、量子互联网。

1 引言

含噪中等规模量子（NISQ）处理器的特点是拥有数百个质量不均、物理连接性受到严格限制的量子比特（qubits）。因此，对大型量子计算机日益增长的需求推动了人们对分布式量子计算（DQC）架构的研究 [1] ，将其视为一种可扩展方案，用于增加可供计算任务使用的量子比特数量；实验工作也已展示了此类设计的一些构建模块 [2]。事实上，借助量子互联网 [3–13], 提供的网络和通信功能，远程量子处理单元（QPUs）能够相互通信与协作，执行单个 NISQ 设备无法独自处理的计算任务。

通常，从局部量子计算转向分布式量子计算时，会面临两大挑战，即量子算法划分和执行管理 [1]。要对一个整体式量子算法进行划分，必须使用量子编译器 来找到最佳拆分方案，也就是尽可能减少在不同设备上存储的量子比特之间执行的量子门数量。此类远程门 可通过三种通信原语实现，我们将它们称为 Teleport [14]（量子态隐形传态）、Cat-Ent（猫态纠缠）和 Cat-DisEnt（猫态退纠缠） [15]。这些原语需要消耗一个纠缠态，并且必须通过量子链路在远程处理器之间分发一个新纠缠态，之后才能执行下一次处理器间操作。通过这些原语，可以执行两类远程操作，即 TeleData 和 TeleGate [2, 16, 17],，如图 1 所示。量子编译器领域的文献 [18–28] 主要关注量子比特分配和远程门调度，而较少关注局部路由的整合。

关于执行管理，一般而言，给定一组待执行的量子电路实例 $\mathscr{P}$ ，应将其划分为互不重叠的子集$\mathscr{P}_i$，使得 $\mathscr{P} = \cup_i \mathscr{P}_i$。随后，必须依次将每个子集分配给可用的量子处理单元。换言之，对于每一轮执行 i，都存在一个调度 $S(i)$，将一些量子电路实例映射到量子网络。如果

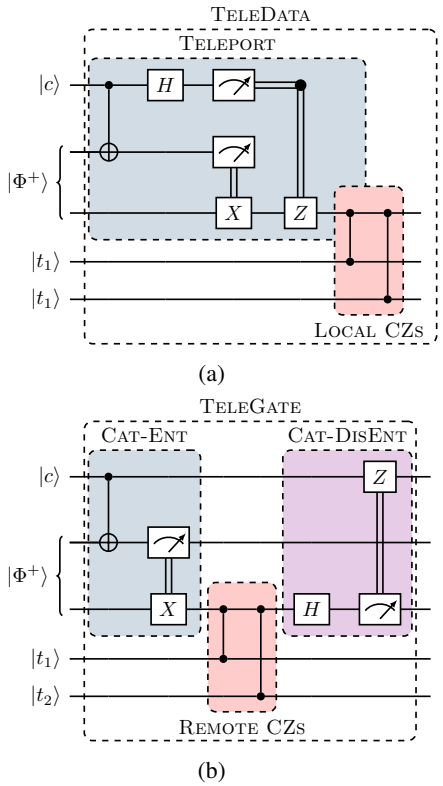


图1：**(1a)** 利用 Teleport 原语表示 TeleData 电路，该原语可将数据量子比特 $|c\rangle$的量子态转移到纠缠对的另一半。虽然纠缠对的状态和原始量子比特 $|c\rangle$此时都已丢失，但随后仍可在传送后的量子比特上执行多个 CZ。**(1b)**利用 Cat-Ent 和 Cat-DisEnt 原语表示 TeleGate 电路。经过 Cat-Ent 操作后，纠缠对的另一半充当原始$c\rangle$控制量子比特的共享副本（由于不可克隆定理，这并非真正的副本）。在 Cat-Ent 和 Cat-DisEnt 之间，可以使用同一控制量子比特、针对不同目标执行多个远程 CZ。值得注意的是，在 Cat-Ent 和 Cat-DisEnt 之间，$|c\rangle$控制量子比特与其共享副本处于纠缠状态，不能作为其他量子门的作用对象。

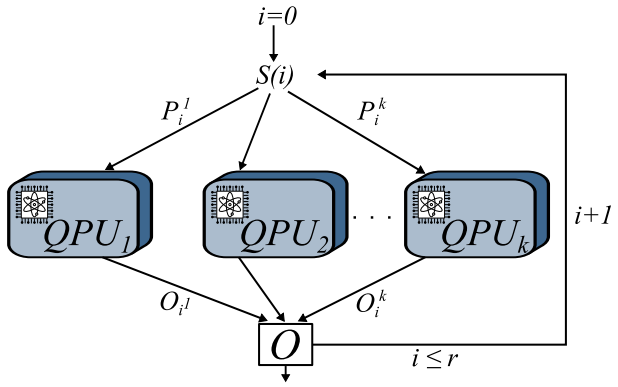


图2：使用k 个量子处理单元执行多个量子电路实例。对于每一轮执行i，调度$S(i)$会将一些量子电路实例映射到量子网络中——每个量子处理单元接收一个量子电路P_i^j，它可以是一个整体电路，也可以是整体电路的一个子电路。经典输出会累积到输出向量O中。

支持分布式量子计算时，某些量子电路实例可以拆分为子电路实例，并分别分配给不同的量子处理单元，如图 2 所示。

在本研究中，我们聚焦于第一个挑战，即量子算法划分。我们提出了一种用于分布式量子计算的模块化量子编译框架，它首次同时考虑了网络和设备的约束与特性。我们使用一些感兴趣的电路（变分量子本征求解器、量子傅里叶变换、图态制备）和不同的网络拓扑，展示了基于所提框架的量子编译器的实验评估；所用量子处理器采用重六边形耦合映射。IBM [29] 选择重六边形拓扑，是因为它具有可扩展性和高性能，既能降低错误率，也为探索纠错码提供了机会 [25,30–33]。我们还设计了一种远程调度策略，可利用 TeleGate 和 TeleData 操作，并测试了仅使用 TeleGate 或同时使用两者的影响。评估结果表明，TeleData 操作可能有助于减少 EPR 对的消耗量，而选择连接性更强的网络拓扑则有助于减少专用于远程操作的层数。

本文结构安排如下。第2节讨论分布式量子计算中的量子编译相关工作。第3节详细介绍所提出的模块化量子编译框架。第4节介绍编译器的基于 Python 的实现及其实验评估。最后，第5节总结全文，并讨论尚待解决的问题和未来工作。

2 相关工作

分布式量子计算的量子编译主要包含两个基本步骤：量子比特分配和远程门调度

。在分布式量子计算中，量子比特分配通常被视为一个划分问题。具体而言，对于给定的一组虚拟量子比特，需要选择一种划分方式，将逻辑量子比特的子集映射到各处理器，同时尽可能减少不同子集之间所需的交互次数。主要目标是尽量减少消耗的纠缠比特——即量子处理单元之间共享的 EPR 对——因为这是分布式量子计算的主要瓶颈。

Andr´es-Mart´ınez 和 Heunen [18] 使用猫态纠缠 [15] 来实现远程量子门。所选门集包含所有单量子比特门，以及两个单独的双量子比特门，即 CNOT门和 CZ门（即 Z 门的受控版本）。作者未对量子处理单元之间的纠缠比特连接性施加任何限制。随后，他们将跨多个量子处理单元分配电路的问题归约为超图划分。该方法在五个量子电路上进行了评估，其中包括量子傅里叶变换。所提出的解决方案存在一些缺点，尤其是无法自定义每个量子处理单元的通信量子比特数量。

Sundaram 等人 [19]提出了一种两步解决方案，第一步是量子比特分配。电路被表示为以量子比特为顶点的边加权图。边权重用于估算猫态纠缠操作的数量。随后，问题被建模为最小 k 割，且各分区大小大致相同。第二步是找出能执行所有 TeleGate 的最小猫态纠缠操作集合。作者指出，在一种特殊情形下，该问题可以归约为顶点覆盖问题，从而能够基于整数线性规划在多项式时间内求得最优解。他们还针对一般情形，借助贪心搜索算法提供了$O(\log n)$近似解，其中n 为全局量子门的总数。 [20], 同一组作者随后借助禁忌搜索算法，将该方法扩展到任意拓扑的异构量子计算机网络。

在 [21], 中，Daei 等人将电路转换为以量子比特为顶点的无向图，而边的权重对应于它们之间双量子比特门的数量。随后，使用用于超大规模集成电路设计的 Kernighan-Lin（K-L）算法对图进行划分 [34], ，以最小化不同分区之间的边数。最后，将每个图分区转换为量子电路。

在 [22], 中，作者将电路表示为二分图，其中包含两组顶点——一组对应量子比特，另一组对应量子门——并通过边来编码量子比特和量子门之间的依赖关系。随后，针对量子比特分配问题，他们提出了一种基于动态规划的划分算法，以最小化 TeleData 操作的数量。

Dadkhah 等人 [23]提出了一种启发式方法，用于替换初始量子电路中的等效电路。随后，他们使用遗传算法对量子比特的布局进行分区，从而优化分布式量子计算通信所需的传态次数。

Nikahd 等人 [24] 利用一种最小k割分区算法，该算法被表述为一个 ILP 优化问题，旨在尽量减少远程交互次数。他们使用滑动窗口，将分区算法应用于电路的较小区段，因此分区可借助 TeleData 操作随滑动窗口而变化。

Cuomo 等人在 [25] 中使用整数线性规划公式对编译问题进行建模。该公式受到动态网络问题的庞大理论体系启发。作者成功将该问题定义为 最快多商品流 的特殊情况。这一结果使得能够借助相关文献中的技术进行优化，例如对分布式架构进行 时间扩展 表示。

Ovide 等人 [26], 研究了 Baker 等人 [27] 提出的量子比特分配策略在一些相关电路上的性能，假设本地和网络均为全连接。在 [27], 中，量子比特分配被视为一个图划分问题，假设存在一种 SWAP 操作原语，可在不同量子处理单元之间交换数据量子比特。也就是说，无需检查量子处理单元上是否有可用的空闲数据量子比特。Ovide 等人表明，一般而言，电路越宽，远程操作的数量越多，尽管这高度取决于要编译的具体电路。

3 模块化量子编译框架

如第 1 节所述，目前缺少一种用于将量子电路编译到 DQC 架构的模块化框架。这种框架应与具体电路无关，即能够将任意电路编译到任意合适的 DQC 架构。此外，该框架还应弥合局部编译与面向 DQC 的编译之间的差距。现有文献中的方案解决了量子比特分配和远程门调度问题，但没有考虑每个量子处理单元的局部连通性。图 3 展示了我们提出的通用量子编译框架。

所提出的量子编译框架以量子电路和网络配置作为输入。如图 4 所示，网络配置描述了各量子处理单元如何连接成目标 DQC 架构，包括量子信道容量，即每个信道的通信量子比特数量。网络配置应包含量子处理单元内部配置的说明，即耦合映射，以及可用数据量子比特和通信量子比特的集合。耦合映射是一个有向图，其中每个顶点对应一个量子比特，有向边决定了是否可以在相连的量子比特之间执行双量子比特门[1]。图 5 展示了一个耦合映射示例，其中包含 20 个数据量子比特和 8 个以蓝色突出显示的通信量子比特。

[1]具体而言，有向边的源顶点和目标顶点分别可以是双量子比特门的控制量子比特和目标量子比特。边也可以是无向的，这意味着两个量子比特都可以充当控制量子比特或目标量子比特。

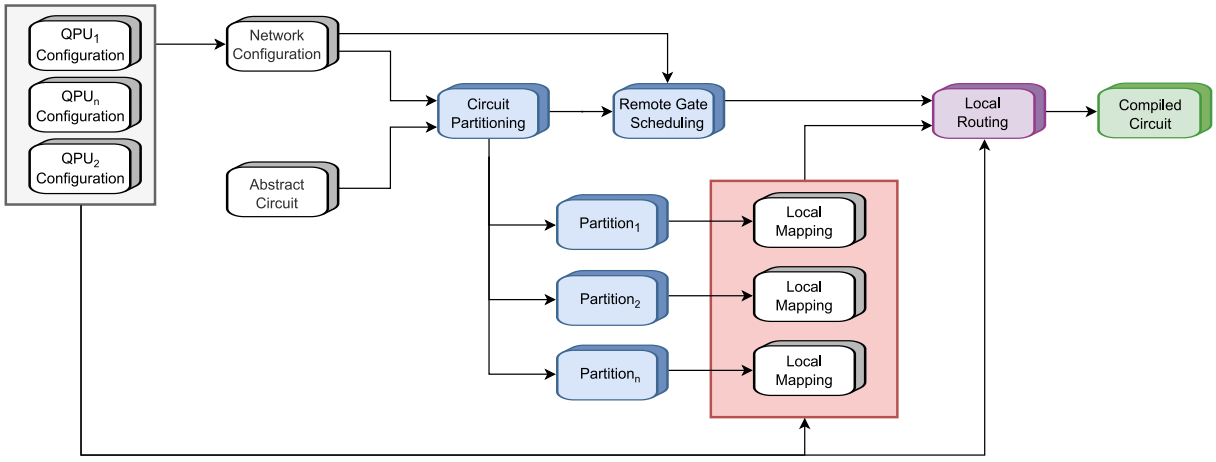


图 3: 所提出的适用于 DQC 架构的模块化量子编译框架的工作流程。

有了这些输入，框架的第一步是进行量子比特分配，如第 3.1 节所述。一旦找到合适的量子比特分配，编译器便会据此对远程门进行调度，并计算分配给每个量子处理单元（QPU）的量子比特的局部映射，具体内容详见第 3.2 节。最后，在考虑先前已调度的远程门的情况下，必须执行局部路由，如第 3.3 节所述。

该框架的输出自然是编译后的电路。该电路是一个独特的对象，包含所有经过恰当调度的本地门和远程门。

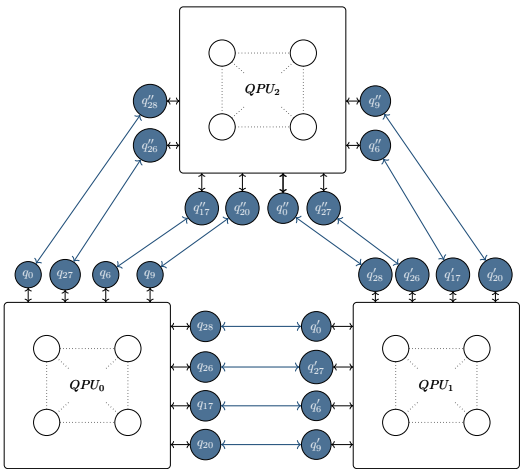


图 4：由 3 个量子处理单元组成的 DQC 架构，如图 5 所示。每个量子处理单元都与其他单元相连，并且每条连接最多支持 4 个通信量子比特 。

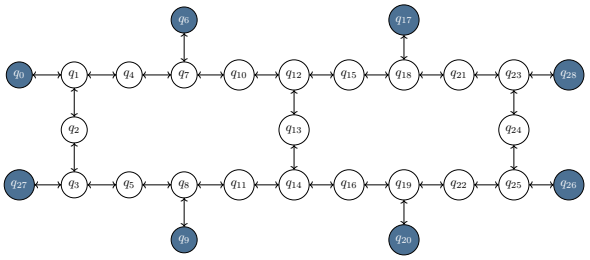


图 5：包含 20 个数据量子比特 和 8 个 通信量子比特的量子处理单元配置，设计灵感来自 IBM 的重六边形器件 [35]。

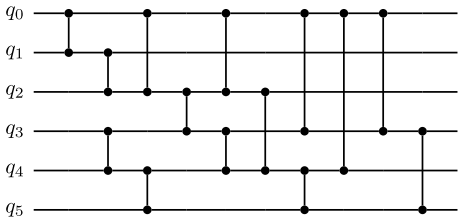
3.1 量子比特分配

如第 1 节所述，目标是对电路进行划分，以尽量降低通信成本，即远程操作的数量

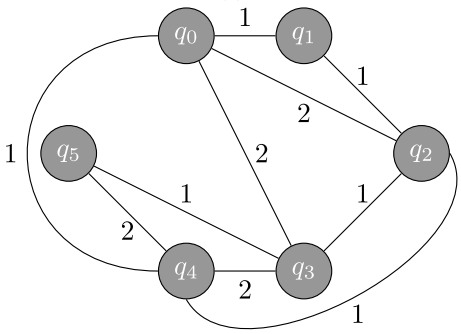
，从而减少消耗的 EPR 对数量。为此，量子电路 qc 可表示为一个无向加权图 $G_{qc}(V, E)$，如图 6 所示，其中每条边 $e \in E$ 的权重为 $W(e) \in \mathbb{N}$。顶点集合 V 对应于 qc 中的量子比特，每条边的权重等于相应量子比特之间双量子比特门的数量。

量子比特分配问题随后可以视为一个图划分问题，其目标是计算一种 k 路划分，使跨越不同分区的边的权重总和最小化。给定 k 个可用的量子处理单元，k 路划分的结果是 k 个大小大致相等的电路分区。目前有多种算法可以高效地找到解决方案。在本研究中，我们采用了 METIS 的多级 k 路划分[2]。这种方法属于初步方案，并非最优方案，因为量子处理单元可能无法得到充分利用，而且电路中的量子比特可能会不必要地分散到所有量子处理单元中。事实上，对于每个分区，我们都会检查将一个量子比特移动到另一个分区是否有助于降低整体通信成本。在所有找到的有效移动方案中，我们选择最佳方案，并迭代地继续搜索可能的移动，直到所有量子比特都已被移动一次，或.....

[2]METIS GitHub 仓库。



(a)



(b)

图 6：**(6a)** 中的量子电路可以表示为**(6b)**中的加权图。

找不到时间或更好的动作时停止。图 7 展示了初始解得到改进的示例。

3.2 远程门调度

我们设计了一种编译流程，用于为 DQC 架构调度远程门，以研究同时使用 TeleData 和 TeleGate 操作的影响。算法 1 描述了其主要策略，该策略需要三个输入项：待分布式执行的量子电路、执行该电路所用网络的配置，以及由编译框架中先前流程计算得出的合适量子比特分配。该流程逐个扫描量子电路中的量子门；当遇到一个根据当前分区涉及不同量子处理单元上的量子比特的量子门时，扫描便会停止。随后，该流程会搜索可行的 TeleData 操作，通过将一个或两个量子比特传送到同一个量子处理单元上来覆盖该量子门。选择 TeleData 操作时会考虑每个量子处理单元的内存容量，同时确保不会因传送而覆盖存储有重要信息的数据量子比特。最后，为每个可能的 TeleData 操作分配一个成本，该成本由公式 1 给出：

$$\frac{n_{EPR}}{n_{cov}}\frac{delay}{\bar{d}_t} \tag{1}$$

其中，n_{EPR} 是消耗的 EPR 对数量，n_{cov} 是覆盖的量子门数量，可能包含比原本需要覆盖的更多量子门——如图 8b 所示——而延迟 是实际执行量子门前必须等待的时间，以离散间隔为单位计量。

[3]这里，“覆盖”指的是使量子门能够执行。

延迟 根据量子链路上次用于纠缠生成的时间以及量子门应当执行的时间进行估算。在执行 TeleData 操作之前，可能需要等待先前已调度的操作完成。延迟 按物理量子比特的平均退相干时间d_t 进行缩放。

算法 1远程门调度器

输入：量子电路 *QC*、网络配置 *N* 和量子比特分配 *P*

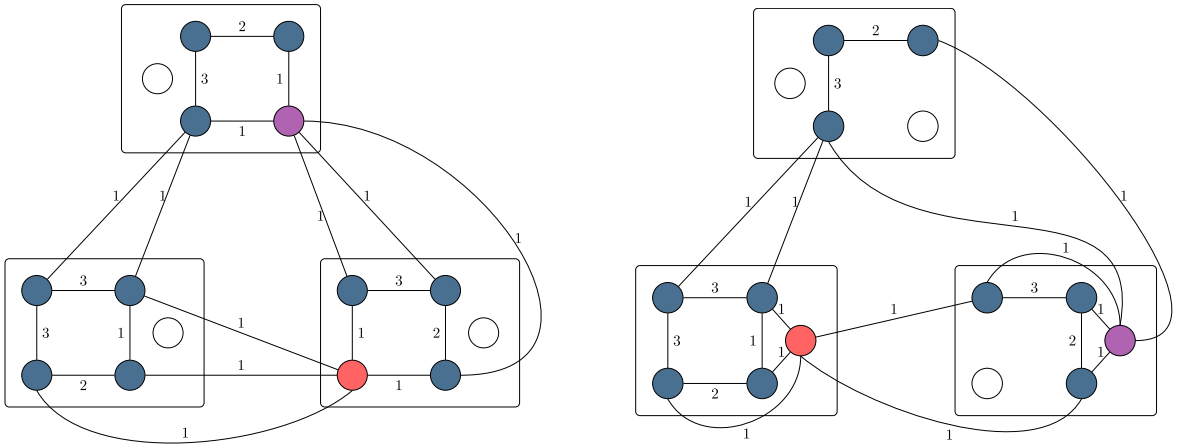
输出：包含远程门的量子电路*D*

```
1: 函数调度
2:     D ← /0
3:     已覆盖 ← /0
4:     对于所有g ∈QC 执行
5:         如果g ∉D，则
6:             如果g 是本地的则
7:                 将g 放入covered 和D
8:             else
9:                 TeleData ← 查找 TELEDATA(g, N, P)
10:                TeleGate ← 查找 TELEGATE(g, N, P)
11:                if成本(TeleData) ¡ 成本(TeleGate) then
12:                    将 TeleData 放入 D
13:                else将TeleGate放入D
14:
15:                结束如果
16:                将 g 放入覆盖 并将D
17:                将额外覆盖的量子门放入 D 并已涵盖
18:            结束 if
19:        结束 if
20:    结束循环
21: 结束函数
```

该过程会选择成本最低的 TeleData 操作，然后使用 TeleGate 操作覆盖剩余未覆盖门的一部分[4] 。TeleGate 操作的选择和调度方式与 TeleData 操作类似。TeleGate 利用 Cat-Ent 原语，如图 1b 所示 [15]。实际上，TeleGate 可分为三个阶段。首先，借助 Cat-Ent 原语（图 1b 蓝色框中的原语），将远程门的控制量子比特“共享”给持有目标量子比特的量子处理单元。然后，由同一个共享控制量子比特控制的门即可在本地执行。最后，借助 Cat-DisEnt 原语（图 1b 紫色框中的原语），将控制量子比特的共享态恢复到第一个量子处理单元。

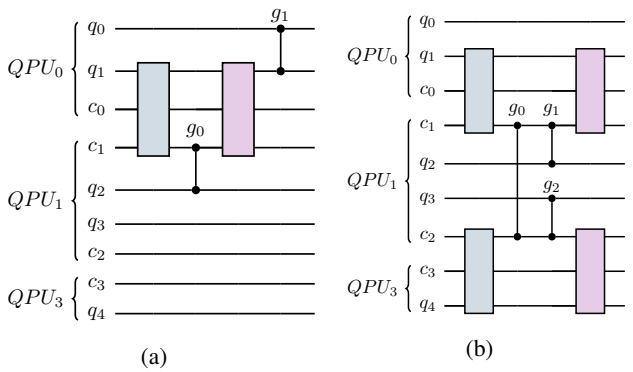
如图 8 所示，TeleData 和 TeleGate 都可以根据此类操作的成本（按公式 1 计算），将一个量子比特迁移到另一个量子处理单元，或将两个量子比特都迁移到其他量子处理单元。图 8a 展示了第一种情况：通过一次 TeleGate 操作，将量子比特 q_1 共享给 QPU_1 ，从而覆盖量子门g_0 。图 8b 展示了第二种情况，其中

[4]要覆盖部分的维度由一个可自定义的参数设定。



(a) (b)

图 7：**(a)** 初始图划分示例。共有三个分区，每个分区包含 4 个量子比特，不同分区之间有 8 条边。通信成本总计为 8。白色节点表示尚未使用的可用量子比特。**(b)** 移动量子比特后的图划分，从而获得更优解。此时通信成本总计为 6。



(a) (b)

图 8：可通过将一个量子比特迁移到另一个量子比特所在的量子处理单元，或将两个量子比特都迁移到其他量子处理单元，来实现量子门。此概念对 TeleGate 和 TeleData 操作均适用。**(a)** 通过使用一个 TeleGate，将量子比特 q_1 共享给 量子处理单元$_1$，即可实现量子门 g_0。**(b)** 通过使用两个 TeleGate，将量子比特 q_1 和 q_4 共享给 量子处理单元$_2$，从而实现量子门 g_0, g_1 和 g_2。

通过两次 TeleGate 操作，将量子比特 q_1 和q_4 共享给 QPU_1,，从而覆盖量子门g_0。这样一来，量子门g_1 和g_2 也得以覆盖。相同的概念也可应用于 TeleData 操作。

该遍还会仅通过调度 TeleGate 操作来编译电路的同一部分。最后，该遍可以计算电路各部分的成本——一部分使用 TeleData 和 TeleGate，另一部分仅使用 TeleGate——并选择成本最低的策略。此处的时间成本仅指消耗的 EPR 对数量。最后，该遍继续扫描量子门，以寻找下一个待覆盖的量子门。

3.3 局部路由

设计的局部路由遍以已调度远程操作的已分区量子线路为输入，并据此处理局部路由。该遍需要已调度远程操作的已分区量子线路、网络配置以及各量子处理单元配置，具体而言，还需要耦合映射，其中包括数据量子比特与通信量子比特之间的连接。

核心策略很直接。该流程会扫描电路，并针对每个涉及同一量子处理单元上未直接连接的量子比特的量子门，计算所需 SWAP 门的最短序列。遇到 TeleData 或 TeleGate 操作时，它会先检查相关数据量子比特是否靠近可用通信量子比特中的某一个，这些通信量子比特对应于该远程操作使用的量子链路。如果不是，则计算到最近最少使用的通信量子比特的最短路径。选择最近最少使用的通信量子比特，是为了尽可能减少纠缠生成的延迟。在编译的这一阶段，由于本地 SWAP 操作，数据量子比特的状态可能会转移到通信量子比特上，反之亦然。这不一定会造成问题 [36],，但在远程操作完成后、再次使用该通信量子比特之前，必须将其移回原始位置。这一点至关重要，否则新的远程操作可能会导致物理存储在通信量子比特位置上的数据量子比特状态丢失。图 9 展示了一个此类示例。

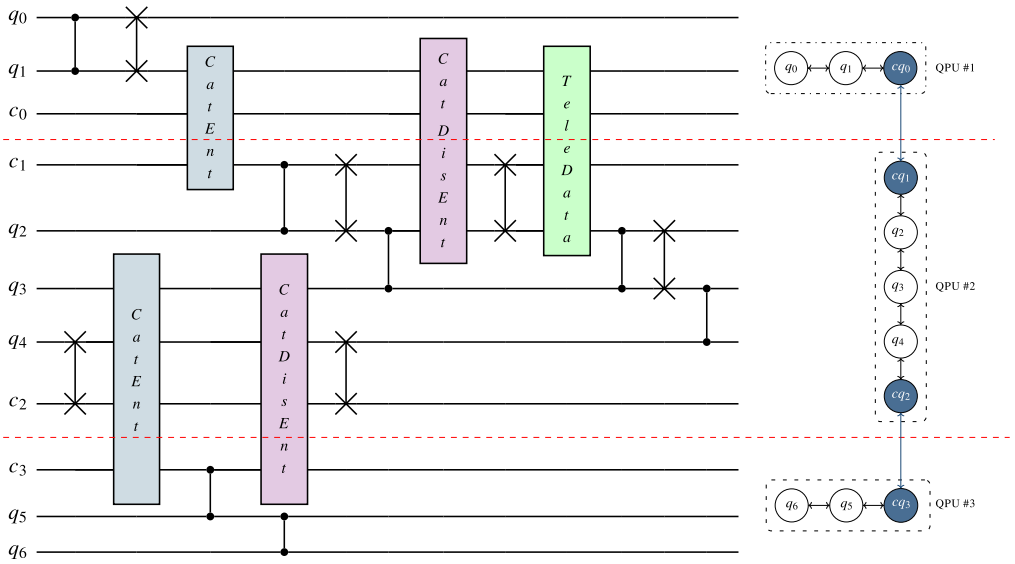


图 9：远程门调度和局部路由示例。局部量子门与 Cat-Ent、Cat-DisEnt 和 TeleData 操作以及 SWAP 门交错执行。

4 评估

我们基于第 3 节介绍的模块化框架实现了一个量子编译器。我们使用三类量子电路对该编译器进行了测试，分别是变分量子本征求解器、量子傅里叶变换和图态电路（图 10 展示了所用图态的一个示例）。上述电路针对图 4 和图 11 所示的 DQC 架构进行编译，这两种架构分别由 3 个和 5 个量子处理单元组成，分别记为*Net-3* 和 *Net-5*。为了增加可用数据量子比特的数量，图 5 中的量子处理单元可以采用模块化方式扩展 [29]。下文中，*QPU-n* 表示具有n个数据量子比特的量子处理单元。对于每种量子电路，测试内容包括仅使用 TeleGate 操作的远程量子门调度，以及同时使用 TeleGate 和 TeleData 操作的远程量子门调度。通过分析每个编译后电路的深度、消耗的 EPR 对数量以及专用于远程操作的层数，评估测试结果。

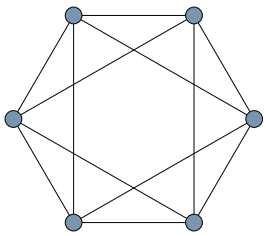


图 10：用于创建图态电路的图态示例。该图态包含 6 个量子比特，但可以扩展到任意大的量子比特数量。

图 12、13 和 14 展示了使用 QPU-21 在 Net-3 上对变分量子本征求解器、量子傅里叶变换和图电路进行编译的结果。可以看出

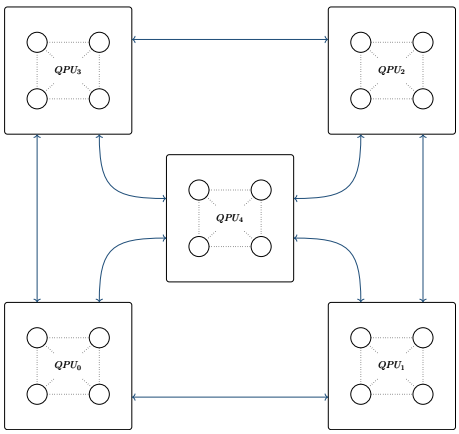


图 11：由 5 个量子处理单元组成的分布式量子计算架构。每个量子处理单元至少与其他 3 个量子处理单元相连，并且根据量子处理单元的拓扑结构，每条连接可能包含多于 1 个 通信量子比特。

与 TeleGate 一同利用 TeleData 操作往往更有益于降低总深度，尤其是对于量子傅里叶变换电路。此外，对于量子傅里叶变换电路，使用 TeleData 操作可以大幅减少所消耗的 EPR 对数量以及专用于远程操作的层数，但会使变分量子本征求解器电路的性能略有下降。对于图态电路，使用 TeleData 操作会导致远程操作层数增加，尽管增幅很小；与此同时，所消耗的 EPR 对数量则会减少。

所有图中，当信道容量从 2 增至 4 时，电路总深度均略有增加。乍看之下，这似乎有悖常理；这可能是局部路由带来的开销，因为局部路由试图使用

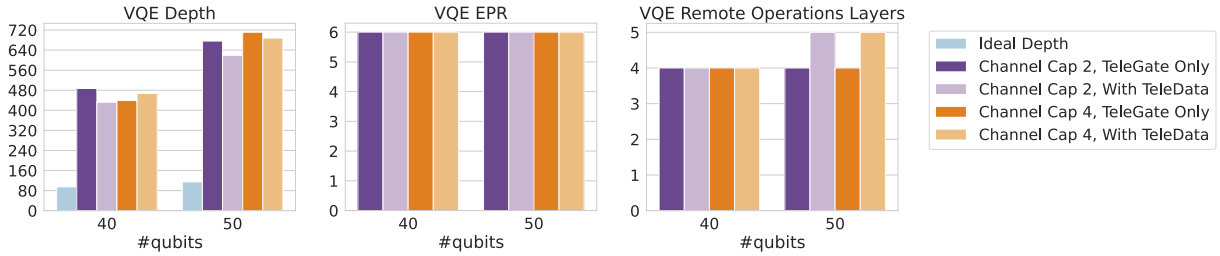


图12：使用 QPU-21 为 Net-3 编译的变分量子本征求解器电路的结果。量子比特数量从 40 到 50 不等，信道容量从 2 到 4 不等。

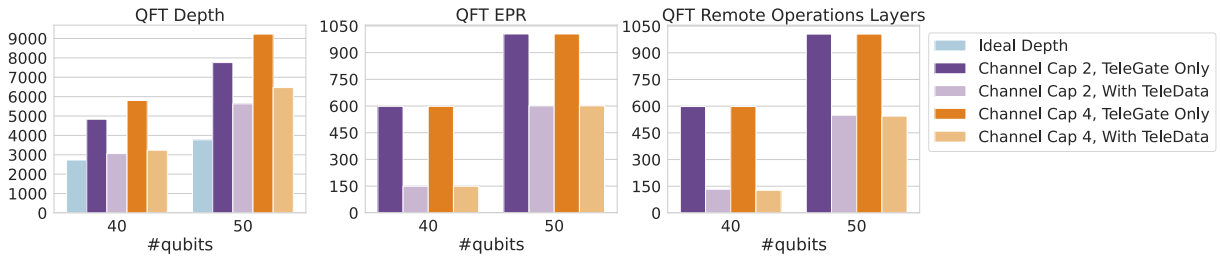


图13：使用 QPU-21 为 Net-3 架构编译的量子傅里叶变换电路的结果。量子比特数量从 40 到 50 不等，信道容量从 2 到 4 不等。

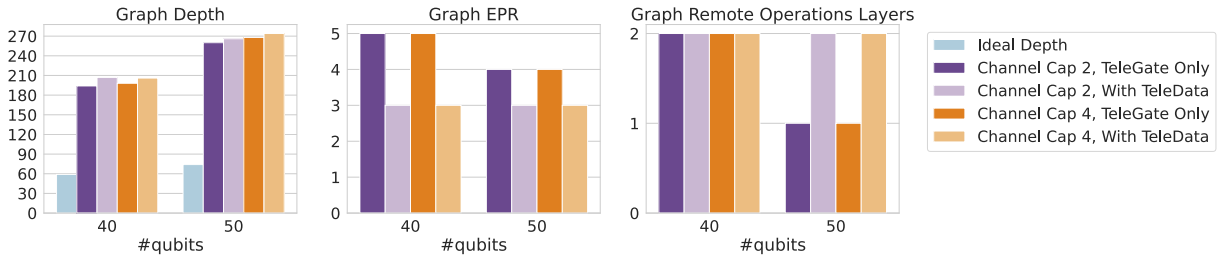


图14：使用 QPU-21 为 Net-3 架构编译的图电路的结果。量子比特数量从 40 到 50 不等，信道容量从 2 到 4 不等。

所有可用的通信量子比特，而不考虑它们在局部耦合图中与数据量子比特之间的距离。有趣的是，远程操作所占用的层数似乎并不随信道容量而变化。我们推测，由于每个量子处理单元上数据量子比特与通信量子比特之间的连通性较低，局部路由操作会造成上游瓶颈，即使信道容量增加，也会产生不利影响。对此还需进一步研究。

还使用 Net-3 和 QPU-63 设备进行了一些测试，电路使用的数据量子比特数量从 80 到 140 不等。结果见图 15、16 和 17。改变信道容量时，结果仍然没有太大差异；不过，在分布式量子傅里叶变换电路时，使用 TeleData 操作大有裨益。从消耗的 EPR 对数量来看，在所测试的电路类别中，量子傅里叶变换电路似乎最依赖远程操作。

保持使用 Net-3、但改用 QPU-125 设备后，编译器在最多包含 250 个量子比特的电路上进行了测试。此时，可以从图 18、19 和 20 中观察到一个有趣的现象。对于图态电路，随着量子比特数量增加，且网络的数据量子比特容量达到上限，尽管消耗的 EPR 对数量保持不变，远程操作的层数几乎没有明显增加。

最后，通过采用配备 QPU-125 设备的 Net-5，进一步增加了数据量子比特的总数。因此，如图 21、22 和 23 所示，可以编译多达 600 个量子比特的电路。这些图中有两个值得注意的结果。首先，对于变分量子本征求解器电路，结果显示，当达到最大量子比特数且采用 TeleData 操作时，远程操作层数略有增加。对于图态电路，则可以观察到相反的趋势：远程操作

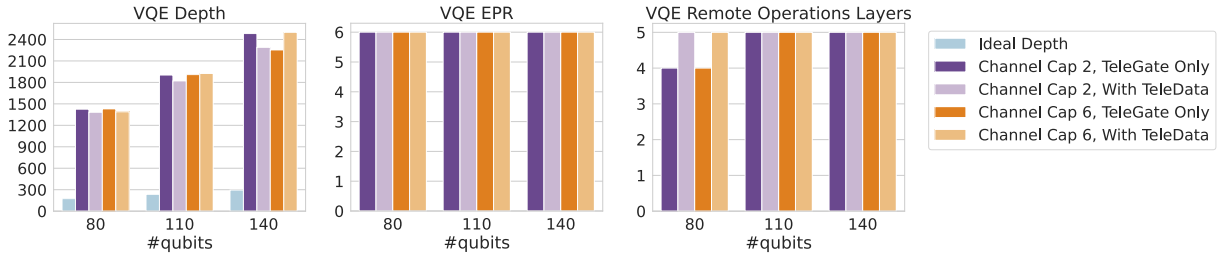


图15：针对使用 QPU-63 的 Net-3 编译的变分量子本征求解器电路结果。量子比特数量在 80 到 140 之间变化，而信道容量在 2 到 6 之间变化。

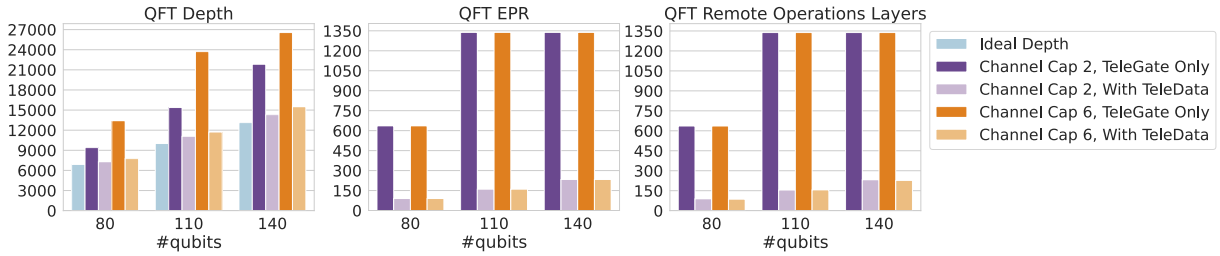


图16：针对使用 QPU-63 的 Net-3 架构编译的量子傅里叶变换电路结果。量子比特数量在 80 到 140 之间变化，而信道容量在 2 到 6 之间变化。

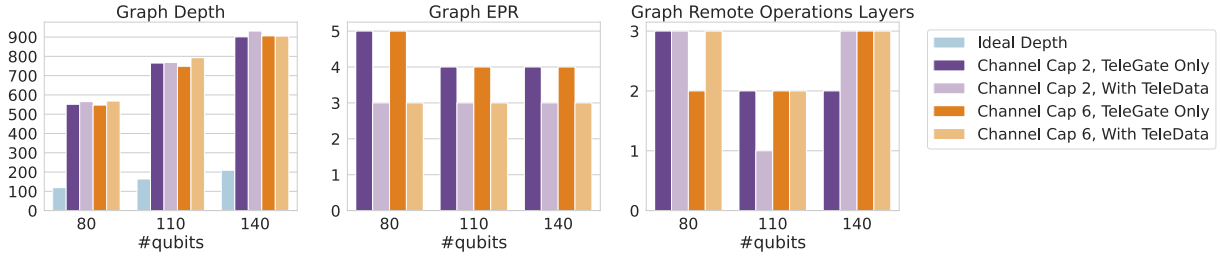


图17：针对使用 QPU-63 的 Net-3 架构编译的图电路结果。量子比特数量在 80 到 140 之间变化，而信道容量在 2 到 6 之间变化。

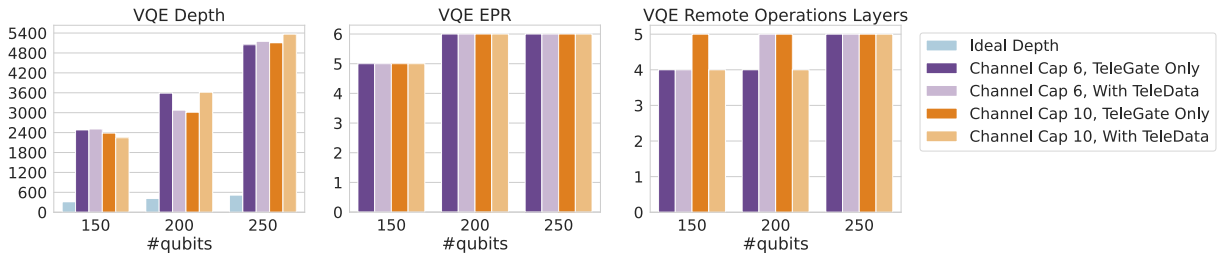


图18：针对使用 QPU-125 的 Net-3 编译的变分量子本征求解器电路结果。量子比特数量在 150 到 250 之间变化，而信道容量在 6 到 10 之间变化。

层数有所减少，尽管幅度不大；此时已达到网络允许的数据量子比特最大数量。这一趋势与此前对同类电路的观察相反，尽管此前采用的是 Net-3 拓扑。这凸显了不同网络拓扑的影响，并表明选择连接性更强的网络事实上是有益的。

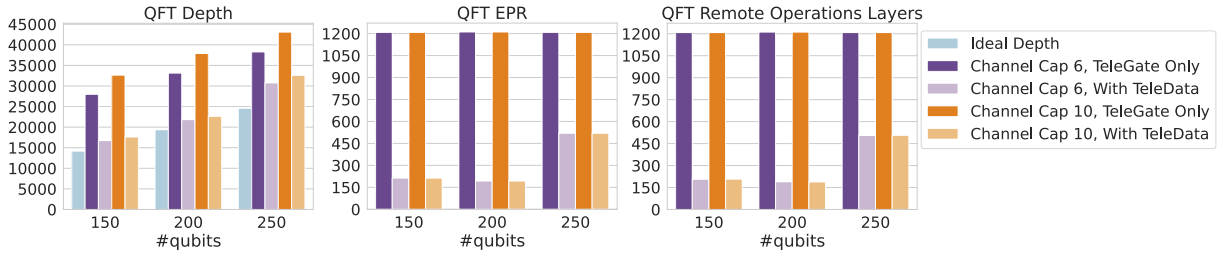


图19：使用 QPU-125 为 Net-3 架构编译的量子傅里叶变换电路的结果。量子比特数量从 150 到 250 不等，信道容量从 6 到 10 不等。

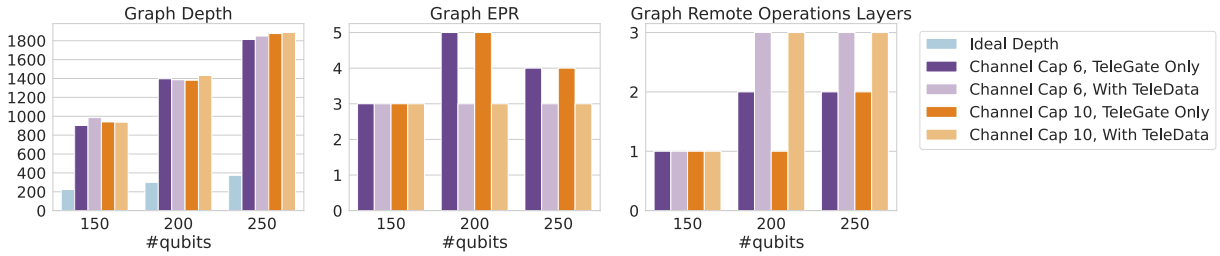


图20：使用 QPU-125 为 Net-3 架构编译的图电路的结果。量子比特数量从 150 到 250 不等，信道容量从 6 到 10 不等。

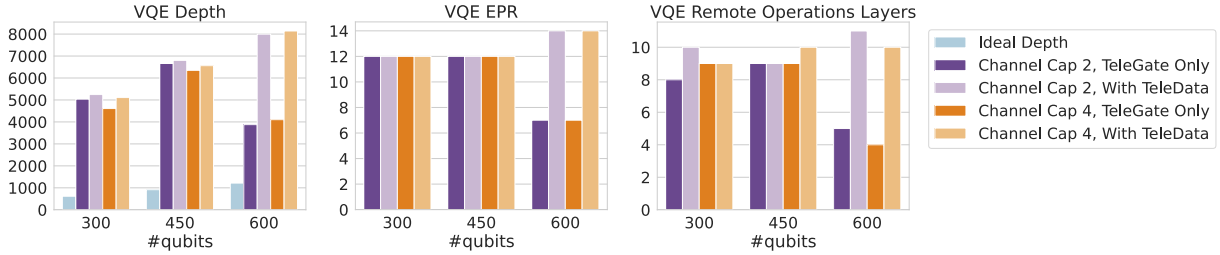


图21：使用 QPU-125 为 Net-5 编译的变分量子本征求解器电路的结果。量子比特数量从 300 到 600 不等，信道容量从 2 到 4 不等。

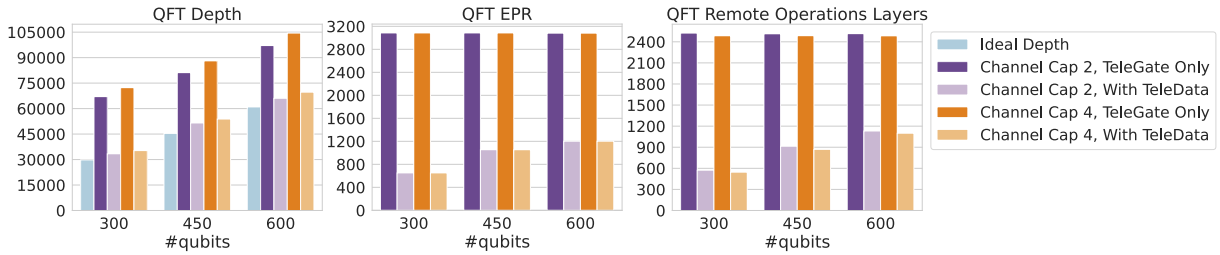


图22：使用 QPU-125 为 Net-5 架构编译的量子傅里叶变换电路的结果。量子比特数量从 300 到 600 不等，信道容量从 2 到 4 不等。

5 结论

在本研究中，我们提出了一种面向分布式量子计算的通用模块化量子编译框架，同时考虑了网络和设备约束及其特性。我们展示了量子编

基于所提出框架的编译器，使用了一些具有研究价值的电路（变分量子本征求解器、量子傅里叶变换、图态制备），这些电路具有不同的宽度（最多 600 个量子比特）。我们考虑了不同的网络拓扑，其中量子处理器采用重六边形耦合图。我们还提出了一种远程调度策略，可以同时利用这两种

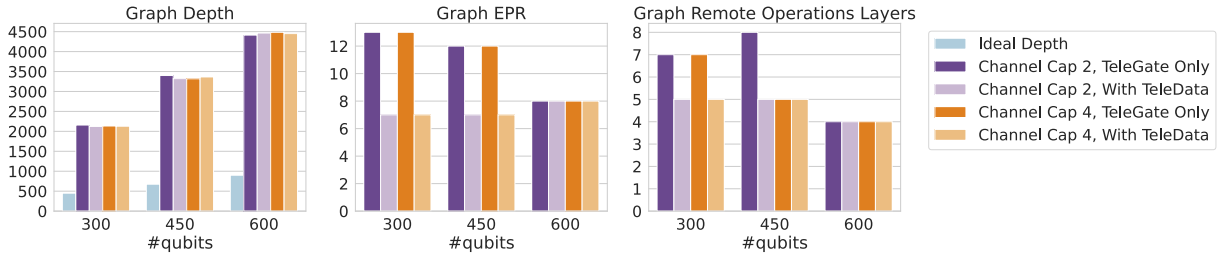


图23：使用 QPU-125 编译到 Net-5 架构的图电路的结果。量子比特数量从 300 变化到 600，而信道容量从 2 变化到 4。

TeleGate 和 TeleData 操作，并测试了仅使用 TeleGate 或同时使用两种操作所产生的影响。我们观察到，TeleData 操作可能会对消耗的 EPR 对数量产生积极影响。此外，我们还表明，选择连接性更强的网络拓扑有助于减少专用于远程操作的层数。

关于未来的工作，我们将专注于把噪声自适应编译策略集成到该框架中，既用于局部路由 [37]，也用于远程门调度。随后，我们将评估不同策略对计算结果质量的影响，而计算结果质量还取决于合适指标的选择。为了生成此类指标，我们需要实际执行编译后的电路，可以通过量子网络模拟器或在真实硬件上执行。在第一种情况下，目前已经有不同抽象层级的模拟器可供使用，具体取决于模拟需要达到的逼真程度。这些模拟对于理解远程操作以及由此产生的任何局部路由开销如何因噪声影响计算质量至关重要。

致谢

作者感谢欧盟通过“量子互联网联盟”项目（欧盟“地平线欧洲”计划，资助协议编号：101102140）提供的量子技术旗舰计划资金支持。本研究得益于意大利帕尔马大学的高性能计算（HPC）设施。

数据可用性

本文所示所有图表的复现所需的全部数据和代码均可在 https://doi.org/10.5281/ zenodo.7896589 获取。

参考文献

[1] M. Caleffi、M. Amoretti、D. Ferrari、D. Cuomo、J. Il- liano、A. Manzalini 和 A. S.Cacciapuoti，“分布式量子计算：一项综述，”*arXiv:2212.10609*，2022年。

[2] R. Van Meter 和 S. J. Devitt，“通往可扩展分布式量子计算之路。”计算机，第 49 卷，第 9 期，第 31–42 页，2016 年 9 月。

[3] S. Pirandola 和 S. L. Braunstein，“物理学：携手构建量子互联网”，自然，第 532 卷，第 7598 期，第 169–171 页，2016 年 4 月。

[4] E. Gibney，“中国卫星为量子互联网迈出一大步”，自然，第 535 卷，第 7613 期，第 478–479 页，2016 年 7 月。

[5]W. D¨ur、R. Lamprecht 和 S. Heusler，《迈向量子互联网》，《欧洲物理学杂志》，第 38 卷第 4 期，页码 043001，2017年5月。[6] C. Simon，《迈向全球量子网络》，《自然·光子学》，第 11 卷第 11 期，第 678–680 页，2017年。[7] S. Wehner、D. Elkouss 和 R. Hanson，《量子互联网：未来发展愿景》，《科学》，第 362 卷第 6412 期，2018年。[8]M. Zomorodi-Moghadam、M. Houshmand 和 M. Houshmand，《优化分布式量子电路中的隐形传态成本》，《国际理论物理学杂志》，第 57 卷，第 848–861 页，2018年。[9] M. Caleffi、A. S. Cacciapuoti 和 G. Bianchi，《量子互联网：从通信到分布式计算！》，载于第五届美国计算机协会纳米尺度计算与通信国际会议论文集，2018年，特邀论文。[10] M. Caleffi、D. Chandra、D. Cuomo、S. Hasaanpour 和 A. S. Cacciapuoti，《量子互联网的兴起》，《*IEEE* 计算机》，第 53 卷，第 67–72 页，2020年。[11] L. Gyongyosi 和 S. Imre，《量子互联网的纠缠浓缩服务》，《量子信息处理》，第 19 卷第 8 期，页码 221，2020年。[12] ——，《量子互联网中可扩展路由的路由空间探索》，《科学报告》，第 10 卷第 1 期，页码 11874，2020年。

[13] M. Amoretti 和 S. Carretta，《遭篡改节点的量子网络纠缠验证》，*IEEE*通信领域精选期刊，第 38 卷第 3 期，第 598–604 页，2020年。

[14]A. S. Cacciapuoti、M. Caleffi、R. Van Meter 和 L. Hanzo，“当纠缠遇上经典通信：量子互联网中的量子隐形传态”，*IEEE*通信汇刊，第68卷第6期，2020年，第3808–3833页。

[15]A. Yimsiriwattana 和 S. Lomonaco，“广义GHZ态与分布式量子计算”，当代数学，第381卷，2005年。

[16] R. Van Meter、K. Nemoto、W. J. Munro 和 K. M. Itoh，“量子多计算机上的分布式算术”，载于第*33*届国际计算机体系结构研讨会（*ISCA'06*），2006年，第354–365页。

[17]R. Van Meter、W. J. Munro 和 K. Nemoto，“实现Shor算法的量子多计算机体系结构”，载于量子计算、通信与密码学理论，Y. Kawano 和 M. Mosca 编。柏林、海德堡：Springer Berlin Heidelberg，2008年，第105–114页。

[18] P. Andr´es-Mart´ınez 和 C. Heunen，“通过超图划分实现量子电路的自动化分布”，物理评论 *A*，第 100 卷，页码 032308，2019年9月。

[19]R. G. Sundaram、H. Gupta 和 C. R. Ra makrish- nan，《量子电路的高效分布》，载于第 *35* 届国际分布式计算研讨会（*DISC 2021*），2021 年。

[20] ——，“在通用量子网络上分布量子电路”，载于 *2022 IEEE*量子计算与工程国际会议（*QCE*），2022年，第415–425页。

[21] O. Daei、K. Navi 和 M. Zomorodi-Moghadam，“优化的量子电路划分”，《国际理论物理学杂志》，第59卷，第12期，第3804–3820页，2020年12月。

[22] Z. Davarzani、M. Zomorodi-Moghadam、M. Housh- mand 和 M. Nouri-baygi，《通过二分图分配量子电路的动态规划方法》，《量子信息处理》，第 19 卷，2020年。

[23] D. Dadkhah、M. Zomorodi 和 S.E.Hosseini，《分布式量子电路优化的新方法》，《国际理论物理学杂志》，第 60 卷，第 9 期，第 3271–3285 页，2021 年 9 月。

[24]E. Nikahd、N. Mohammadzadeh、M. Sedighi 和 M. S. Zamani，《量子电路的自动化窗口式划分》，《*Phys. Scr.*》，第 96 卷第 3 期，035102 页，2021 年 1 月。

[25]D. Cuomo、M. Caleffi、K. Krsulich、F. Tramonto、G. Agliardi、E. Prati 和 A. S.Cacciapuoti，《面向分布式量子计算的优化编译器》，《美国计算机协会量子计算汇刊》，第 4 卷第 2 期，2023 年 2 月。[26] A. Ovide、S. Rodrigo、M. Bandic、H. Van Someren、S. Feld、S. Abadal、E. Alarcon 和 C. G. Almudever，《将量子算法映射到多核量子计算架构》，*arXiv:2303.16125*，2023 年。

[27] J. M. Baker、C. Duckering、A. Hoover 和 F. T. Chong，《面向模块化架构的分时量子电路划分》，载于第 *17* 届美国计算机协会国际计算前沿会议论文集，CF '20 系列。美国纽约：美国计算机协会，2020年，第 98–107 页。

[28] D. Ferrari、A.S.Cacciapuoti、M. Amoretti 和 M.Cal- effi，“分布式量子计算的编译器设计”，*IEEE*量子工程汇刊，第2卷，第1–20页，2021年。

[29]IBM，“IBM Quantum heavy hex 晶格”，https://research.ibm.com/blog/heavy-hex-lattice，2021。

[30] M. Takita、A. W. Cross、A. D. C´orcoles、J. M. Chow 和 J. M. Gambetta，《利用超导量子比特进行容错态制备的实验演示》，《物理评论快报》，第 119 卷，第 180501 页，2017 年 10 月。

[31] N. Sundaresan、I. Lauer、E.Pritchett、E.Magesan、P. Ju- rcevic 和 J. M. Gambetta，“通过优化的旋转回波减少交叉共振中的幺正误差和旁观者误差”，*PRX Quantum*，第 1 卷，第 020318 页，2020年12月。

[32] A. D. C´orcoles、E. Magesan、S. J. Srinivasan、A. W. Cross、M. Steffen、J. M. Gambetta 和 J. M. Chow，《使用由四个超导量子比特组成的方形晶格演示量子错误检测码》，《自然通讯》，第 6 卷，第 1 期，第 6979 页，2015 年 4 月。

[33] C. Chamberland、G. Zhu、T. J. Yoder、J. B. Hertzberg 和 A. W. Cross，《带标记量子比特的低度图上的拓扑码与子系统码》，《物理评论*X*》，第 10 卷，011022，2020 年 1 月。

[34] B. W. Kernighan 和 S. Lin，《一种高效的图划分启发式方法》，《贝尔系统技术期刊》，第 49 卷，第 2 期，第 291–307 页，1970 年。

[35]IBM，“IBM量子系统”，https: // quantum-computing.ibm.com/services/resources，2020年。

[36]A. Dahlberg、B. Van Der Vecht、C. Delle Donne、M. Skrzypczyk、I. Te Raa、W. Kozlowski 和 S. Wehner，《NetQASM——一种低级指令集

架构，用于量子互联网中的混合量子—经典程序》，量子科学与技术，第 7 卷第 3 期，页码 035023，2022 年 6 月。[在线]。网址：https://dx.doi.org/10.1088/2058-9565/ac753f

[37]D. Ferrari 和 M. Amoretti，“采用应用驱动基准评估的噪声自适应量子编译策略”，载于第 *19* 届美国计算机协会计算前沿国际会议论文集，丛书 CF ’22。美国纽约：美国计算机协会，2022 年，第 237–243 页。



DavideFerrari (GSM’20)于2019年获得意大利帕尔马大学计算机工程理学硕士学位。随后，他在帕尔马大学未来技术实验室担任研究学者，致力于量子编译高效算法的设计。在信息技术博士学位学习期间，他在该系的研究……

帕尔马大学工程与建筑系，他从事量子编译、量子优化和分布式量子计算研究。目前，他是帕尔马大学工程与建筑系的研究员。他参与帕尔马大学的量子信息科学（QIS）研究计划，并且是量子软件研究组的成员。2020年，他赢得了“IBM量子奖电路优化开发者挑战赛”。他的研究重点是量子优化应用，以及面向本地和分布式量子计算的高效量子编译。



Stefano Carretta 于2005年获得意大利帕尔马大学物理学博士学位，现任帕尔马大学物质物理学正教授。他的研究工作主要聚焦于磁性分子的量子行为理论建模和量子信息处理。他参与提出了最早一批关于利用磁性

将分子用作量子比特，并首次提出利用分子纳米磁体作为量子模拟器。他曾是2021—2027年意大利国家研究计划（PNR）量子技术专家委员会的成员，并参与多个涉及量子技术的国家及欧洲项目。目前，他是一个欧洲研究委员会（ERC）协同项目的首席研究员之一。他是150多篇发表于国际期刊的研究论文的合著者。2006年，他因在分子纳米磁体方面的研究，获颁《科学》（意大利版《科学美国人》）奖章和意大利共和国总统奖章。2011年，他因对分子磁性理论的贡献，获颁著名的奥利维耶·卡恩国际奖。



MicheleAmoretti （S’01-M’06-SM’19）于2006年获得意大利帕尔马帕尔马大学信息技术博士学位。他是帕尔马大学计算机工程副教授。2013年，他曾在法国格勒诺布尔的LIG实验室担任访问研究员。他目前的研究

他的研究兴趣主要包括量子计算、高性能计算和物联网。他在经过同行评审的国际期刊、会议论文集和书籍中发表或合著了130余篇研究论文。他担任以下期刊的副主编： IEEE《量子工程汇刊》和《国际分布式传感器网络杂志》。他参与帕尔马大学的量子信息科学（QIS）研究与教学计划，并领导量子软件实验室。他目前是欧洲HE项目量子互联网联盟的首席研究员之一。他是CINI联盟在UNI/CT 535“量子技术”UNINFO委员会中的代表；该委员会是ISO/IEC JTC 1 WG 14“量子信息技术”和CEN/CENELEC JTC 22“量子技术”在意大利的对应机构。