

量子计算系统软件研究综述*

谢磊, 翟季冬

(清华大学 计算机科学与技术系, 北京 100084)

通信作者: 翟季冬, E-mail: zhajidong@tsinghua.edu.cn



摘要: 量子计算理论上有望解决诸多经典难解问题, 近年来量子计算机的快速发展正推动这一理论进入实践. 然而, 当前硬件中繁多的错误会造成计算结果出错, 严重限制了量子计算机解决实际问题的能力. 量子计算系统软件位于应用与硬件之间, 充分挖掘系统软件在硬件错误减缓方面的潜力, 对于近期实现有实用价值的量子计算而言至关重要. 由此, 近期涌现了一批量子计算系统软件研究工作. 将这些工作归纳入编译器、运行时系统和调试器 3 个范畴, 通过对它们的分析总结, 梳理量子计算系统软件的研究现状, 揭示其在硬件错误减缓方面的重要作用. 并对未来的研究方向进行展望.

关键词: 量子计算; 系统软件; 嘈杂中规模量子; 编译; 运行时; 调试

中图法分类号: TP311

中文引用格式: 谢磊, 翟季冬. 量子计算系统软件研究综述. 软件学报, 2024, 35(1): 1-18. <http://www.jos.org.cn/1000-9825/6908.htm>

英文引用格式: Xie L, Zhai JD. Survey on Quantum Computing System Software. Ruan Jian Xue Bao/Journal of Software, 2024, 35(1): 1-18 (in Chinese). <http://www.jos.org.cn/1000-9825/6908.htm>

Survey on Quantum Computing System Software

XIE Lei, ZHAI Ji-Dong

(Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China)

Abstract: Quantum computing is expected to solve many typical and difficult problems in theory. The rapid development of quantum computers in recent years is pushing the theory into practice. However, numerous errors in current hardware can cause incorrect computational results, which severely limit the ability of quantum computers to solve practical problems. Quantum computing system software lies between applications and hardware. In addition, tapping the full potential of the system software in mitigating hardware errors is crucial to realizing practical quantum computing in the near future. As a result, many research works on quantum computing system software have recently emerged. This study classifies them into three categories: compilers, runtime systems, and debuggers. Through an in-depth analysis of these works, the study sorts out the research status of quantum computing system software and reveals their important roles in mitigating hardware errors. This study also looks forward to future research directions.

Key words: quantum computing; system software; noisy intermediate-scale quantum; compilation; runtime; debugging

量子计算被视为下一代信息处理技术, 理论上可有效求解诸多经典难解问题^[1,2]. 近年来, 量子计算机迅猛发展, 推动量子计算从理论步入实践. 例如, 自 2016 年以来, IBM 已先后发布 20 余台超导量子计算机, 所操纵的量子比特数目也达到了 65 个之多^[3]; 2019 年, Google 发布 53 量子比特超导量子计算机, 并首次实现了量子计算发展的里程碑事件——“量子优越性”(或称“量子霸权”)^[2]. 我们迎来了嘈杂中规模量子 (noisy intermediate-scale quantum, NISQ) 时代^[4].

嘈杂中规模量子时代的主要特点是: 量子计算机规模有限、错误繁多. 这给量子计算应用的执行带来了诸多

* 基金项目: 国家自然科学基金 (62225206)

收稿时间: 2022-04-08; 修改时间: 2022-09-17; 采用时间: 2023-01-19; jos 在线出版时间: 2023-07-05

CNKI 网络首发时间: 2023-07-06

挑战. 图 1 展示了量子计算的系统栈, 从上至下可分为应用、系统软件、体系结构和硬件 4 层. 其中, 系统软件旨在挖掘应用特点、利用硬件特性, 以在硬件上更好地运行应用. 在嘈杂中规模量子时代, 针对硬件错误特点设计系统软件, 以增强应用的错误容忍能力, 对于减缓硬件错误的影响、终而实现有实用价值的量子计算而言至关重要.

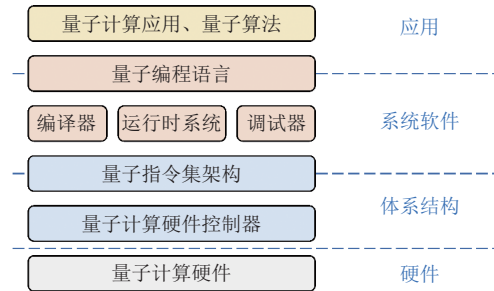


图 1 量子计算系统栈

正因如此, 近期涌现了一批量子计算系统软件相关研究工作, 这些工作以提高计算正确性为目标, 着重针对嘈杂中规模量子时代的特点进行设计. 这些工作可归入编译器、运行时系统和调试器 3 个范畴, 这三者的关系如下.

- 为实现量子计算应用与量子算法, 人们使用高级语言编写量子计算程序. 调试器可检测程序编写过程中的编程错误, 从而确保编写的程序符合预期逻辑.
- 在使用高级语言编写程序后, 编译器负责将其翻译为低级语言表示, 并进行编译优化、错误减缓处理等.
- 最后, 编译好的程序交由运行时系统执行, 运行时系统通过某种可减缓错误的运行策略, 调控程序的具体执行过程, 以获得更加准确的计算结果.

本文旨在通过对这些工作的总结分析, 梳理量子计算系统软件的研究现状, 揭示系统软件在硬件错误减缓方面的重要作用, 并展望未来的研究方向. 出于完整性考虑, 本文在必要时也将引用早期研究工作的成果.

本文第 1 节概述量子计算基本概念和嘈杂中规模量子时代硬件的发展现状, 着重指出嘈杂特点对量子计算的严重影响. 第 2-4 节分别分析在量子计算编译器、运行时系统和调试器上的研究工作, 梳理研究现状. 第 5 节展望未来研究方向. 第 6 节总结全文.

1 量子计算概述

本节概述量子计算基本概念、典型的实现技术以及当前时代量子计算机的特点.

1.1 量子计算

量子计算通过一系列量子门 (quantum gate) 操纵若干量子比特 (qubit) 完成. 量子比特是量子信息的承载物, 与经典比特取值 0、1 类似, 单个量子比特可取值 $|0\rangle$ 、 $|1\rangle$, 不同的是, 量子比特的状态还可取为二者的叠加, 即 $a|0\rangle + b|1\rangle$, 其中 a 、 b 为复数且 $|a|^2 + |b|^2 = 1$. 类似地, n 个量子比特的状态可表示为 2^n 个基本状态的叠加: $\sum_{i_1 i_2 \dots i_n} a_{i_1 i_2 \dots i_n} |i_1 i_2 \dots i_n\rangle$, 其中 $i_1 i_2 \dots i_n$ 遍历所有 n 位 0、1 组合. 量子比特这种多个基本状态叠加的能力使其状态空间规模远超同等个数经典比特的状态空间, 蕴含量子计算在某些问题上优于经典计算的可能.

量子门用于改变量子比特状态. 常见的单比特门有非门 X (可使 $|0\rangle$ 、 $|1\rangle$ 状态互相转换), 哈达玛 (Hadamard) 门 H (可创造基本状态的叠加) 等. 常见的两比特门有受控非门 CNOT (基于控制比特状态决定是否在受控比特上施加非门), 交换门 SWAP (交换两个量子比特的状态) 等. 当对于任意给定的量子门, 其都可以由一组固定的量子门组合实现时, 称这组门构成了一个通用量子计算门集. 所有单比特门加上 CNOT 门即是一个典型的通用门集.

量子计算的结果通过量子测量 (quantum measurement) 获取. 量子测量具有两个主要性质: 一是结果具有随机性, 二是测量会改变量子比特的状态. 以单个量子比特 $a|0\rangle + b|1\rangle$ 为例, 测量结果或 0 或 1, $|a|^2$ ($|b|^2$) 为测得 0 (1) 的概率, 在测量后, 量子比特的状态会相应地塌缩到 $|0\rangle$ ($|1\rangle$) 态. 类似地, n 个量子比特的测量结果为一个 n 位 0、

1 串 $i_1 i_2 \dots i_n$, 相应的概率为 $|a_{i_1 i_2 \dots i_n}|^2$, 测量后塌缩到 $|i_1 i_2 \dots i_n\rangle$. 尽管测量具有随机性, 但一个好的量子算法会最大化测得问题解的概率.

量子计算流程通常采用量子电路 (quantum circuit) 模型描述. 图 2 展示了 1 个例子: 横线表示量子比特, 共有 q_1 、 q_2 、 q_3 这 3 个量子比特; 首先在 q_1 上施加哈达玛门 H , 而后在 q_1 和 q_2 上施加 CNOT 门, 其次在 q_2 和 q_3 上施加 SWAP 门, 最后对 3 个量子比特进行测量获得计算结果. 根据门之间的依赖关系, 可定义量子电路的深度为依赖关系图中最长路径的长度, 例如示例中的电路深度为 4 (包含测量).

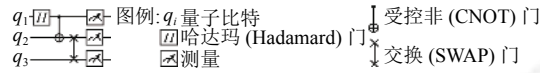


图 2 量子电路示例

读者可参考文献 [5] 进一步了解量子计算相关概念.

1.2 量子计算的若干种实现技术

当下主流的量子计算实现技术有超导 [6]、离子阱 [7]、光子 [8]、量子点 [9] 等, 其中超导技术近年来发展最为迅速. 以下以超导技术为例简要介绍量子计算的实现方式.

图 3 展示了一种典型的超导量子计算实现方案 [10], 方框内为量子比特, 实质是某种固态电路, 其在低温超导环境下的量子力学描述符合量子比特的数学性质. 每个量子比特连接若干条控制线 (例如图中的 XY 控制线和 Z 控制线) 和读取谐振腔, 通过控制线输入的电磁脉冲用于实现量子门, 而读取谐振腔用以进行量子测量. 为了实现两比特门, 两个量子比特间还需要通过电容、电感或总线等方式进行耦合, 例如图中采用总线的方式 (另一个参与耦合的量子比特未画出). 总结而言, 基于上述方案执行量子计算的过程为: 利用控制线输入特定的电磁脉冲以实现一系列的量子门, 最终通过读取谐振腔获得量子比特的状态.

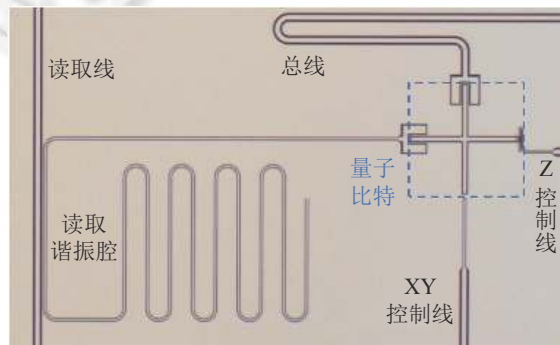


图 3 基于 Xmon 的超导量子计算实现方案 [10]

通过上述描述可以发现, 超导量子计算与经典计算中的存内计算 (processing in memory) 相似: 量子信息 (数据) 存放于量子比特 (内存) 中, 所有的操作直接作用于量子比特. 实际上, 这一特点并非超导量子计算所特有, 而是主流量子计算实现技术的通性, 后文中将会看到, 该特点影响了量子计算系统软件的设计.

1.3 嘈杂中规模量子时代的特点

当前量子计算的发展处于嘈杂中规模量子时代, 这一时代的量子计算机具有两大特点 [4]: 1) 中规模: 量子比特数目较少, 通常在数十至数百; 2) 嘈杂: 噪声繁多, 导致各种操作错误横生. 其中, 嘈杂的特点对量子计算机能力的限制尤为严重 [4,11]. 下面着重介绍近期量子计算机中存在的主要错误类型.

1.3.1 退相干

由于量子比特与环境相互作用, 随着时间推移, 量子比特的状态会逐渐遭到破坏, 这称为退相干 (decoherence). 退相干有两个主要表现 [12]: 退回基态 ($|0\rangle$) 和退相, 可分别由能量弛豫时间 (T_1) 和相位弛豫时间 (T_2) 来刻画 (统称

为相干时间). 相干时间越短, 退相干效应就越明显, 可用于量子计算的时间就越短. 根据近期量子计算机的相干时间以及门执行时间^[11], 可得近期可执行的量子电路深度通常在数十至数百, 能解决的问题规模非常受限.

1.3.2 量子门错误、状态初始化与测量错误

环境的干扰和控制的不精确等会导致实际执行的量子门与期望之间存在差别, 这称为量子门错误. 量子门错误可用不保真度^[5] (通常也称为错误率) 来刻画, 不保真度越低, 则量子门错误越少. 对于近期的量子计算机, 单比特门不保真度通常在 10^{-3} – 10^{-2} 量级, 两比特门错误较多, 不保真度在 10^{-2} – 10^{-1} 量级.

除量子门外, 量子比特的初始化和测量亦会发生错误. 人们通常关心其中的翻转错误, 以测量为例, 单个量子比特有“测量|0⟩却得到1”和“测量|1⟩却得到0”两种情况. 对于近期的量子计算机而言, 这两种情况的错误率通常在 5%–10%; 且由于量子比特退相干等因素, |1⟩有退回到|0⟩的倾向, 因而测量|1⟩的错误率通常会高于测量|0⟩的错误率.

1.3.3 串扰

串扰 (crosstalk) 指系统的不同组成部分之间发生不期望的相互干扰. 例如在超导量子计算机中, 典型的串扰有^[13]: 1) 不同传输线上的信号相互影响; 2) 量子比特间存在不期望的耦合; 3) 不同量子比特的读取腔发生耦合等. 串扰的主要危害在于恶化同时执行的多个操作和造成相关性错误 (correlated error). 例如, 已有研究表明, 在 IBMQ 机器上同时执行两个 CNOT 门可导致门错误率增加 20 倍^[14], 而相关性错误可降低量子纠错的能力, 阻碍容错量子计算的发展^[15].

综上所述, 近期量子计算的整个过程伴随诸多错误发生. 图 4 显示了这些错误的综合效果: 不仅会大幅降低量子计算的正确性, 导致结果错误, 还会限制计算规模, 使有价值的问题难以得到解决. 因而, 近期量子计算系统软件的主要目标之一即是设法降低这些错误对量子计算的影响, 提高量子计算在嘈杂硬件上的正确率.

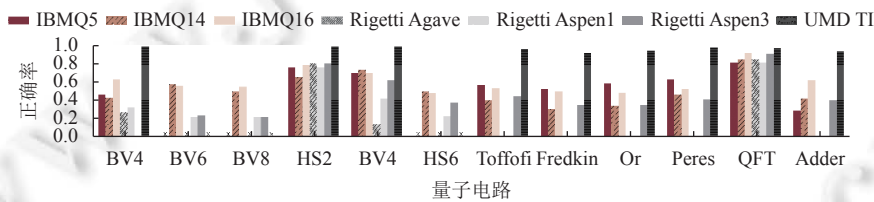


图 4 量子电路在不同量子计算机上执行的正确率 (所用电路量子比特数目 ≤ 8 , 量子门数目 ≤ 48)^[11]

2 量子计算编译器

近期有关量子计算系统软件的研究主要集中于 3 类系统软件: 编译器、运行时系统和调试器. 本节介绍编译器方面的研究进展.

编译器旨在将高级语言描述的程序优化、翻译为低级语言表示. 近期编译优化工作主要针对量子电路模型表示的程序. 基于量子电路模型, 量子计算编译器的主要流程如图 5 所示.

以下以图 6 为例简要介绍编译流程. 首先, 量子门分解 (quantum gate decomposition) 将电路中硬件无法执行的门分解为可执行的门, 这里采用多数研究工作 (以及 IBMQ 机器) 的设定, 即任意单比特门和 CNOT 门均可被执行, 所以示例电路中没有需要分解的门. 之后, 量子比特映射 (qubit mapping) 将描述量子电路所用的逻辑量子比特映射到硬件上的物理量子比特, 这里将 q_i 映射到 i . 在这种映射方式下, $CNOT_{1,3}$ 由于硬件上不存在①与③间的耦合而无法执行, 量子比特路由 (qubit routing) 通过插入 $SWAP_{1,4}$ 交换①和④的状态 (此处直接将 SWAP 门分解为了 3 个 CNOT 门), 使得原本在①、③间的 CNOT 可在④、③间执行. 在解决执行可行性问题后, 量子电路会经历若干优化过程, 示例展示了一种典型的优化方法, 即连续单比特门合并为一个, 以减少量子门的数目. 最后, 量子门调度 (quantum gate scheduling) 安排每个门的执行时间, 此时的量子电路即可在量子计算机上执行了.

以下详细介绍各个编译流程及相关研究工作.

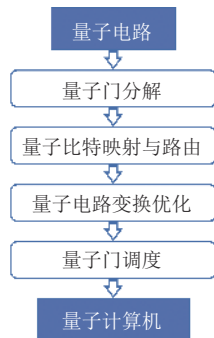


图5 量子计算编译器的主要流程

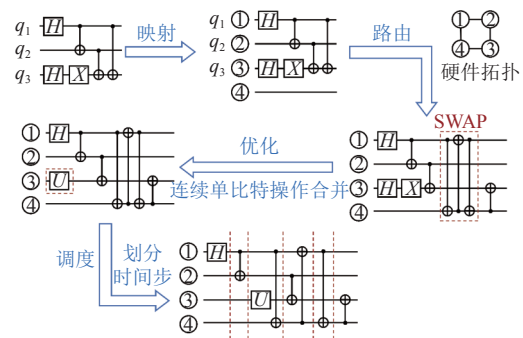


图6 量子电路编译过程示例

2.1 量子门分解

受制于实现难度和校准开销等因素,量子计算机并不直接支持任意 n 比特门的执行,而仅支持某个通用量子计算门集.例如,表1列出了目前若干量子计算机支持执行的门,可以看出,只有少数单比特门和两比特门可被直接执行.对于任意 n 比特门,需要将其分解为硬件支持的门.

表1 若干量子计算机直接支持执行的门

量子计算机	架构	单比特门	两比特门
IBMQ ^[3]	超导	U_1, U_2, U_3	CNOT
Google ^[2,16]	超导	PhasedXZ	fSim
Rigetti Aspen ^[17]	超导	R_x, R_z	CZ, XY
IonQ ^[18,19]	离子阱	GPI, GPI2, GZ	MS

量子门分解通常包含两个步骤:首先将任意 n 比特门分解为单比特门和 CNOT 门,而后将分解出来的单比特门和 CNOT 门进一步转换为硬件上可执行的单、两比特门.在这一流程中,单比特门和 CNOT 门作为硬件无关门集,构建了不同分解方法比较的统一标准,为多数工作所用.此外,这一硬件无关门集到硬件支持门集的转换也较为容易(通常采用预定义规则).以下主要探讨任意 n 比特门到这一硬件无关门集的分解.

由于 CNOT 门较之单比特门更难实现且错误更多,最坏情况下分解所需的 CNOT 门数即成为衡量不同分解方法优劣的主要指标.目前已知所需 CNOT 门数的最高下界为 $\lceil (4^n - 3n - 1)/4 \rceil$ ^[20], 仍未有达到此理论下界的通用分解方法.

主流的分解方法均为递归方法:通过将 n 比特门的分解转化为 $n-c$ 比特门 (c 为常数) 的分解,从而递归解决.根据转化方法的不同,分为 QR 分解,正余弦分解 (cosine-sine decomposition, CSD) 和量子香农分解 (quantum Shannon decomposition, QSD).文献 [21] 首次给出了分解任意 n 比特门的方法,核心思路是将量子门的矩阵表示通过矩阵 QR 分解逐步分解,该方法所需 CNOT 数目为 $O(n^3 4^n)$ (文献 [5] 第 4.5 节有对该过程的详细描述,读者可进一步参考阅读).在此基础上,文献 [22] 通过使用格雷码作为量子门矩阵表示的基向量,将所需 CNOT 数目降低至 $O(4^n)$,达到了理论下界的同阶复杂度,但其 4^n 前的系数较大(系数上界约为 11).为了进一步降低这一系数,文献 [23] 采用另一种矩阵分解方法——正余弦分解 (CSD),并配合反射格雷码等优化手段,将 CNOT 数目降低至 $4^n - 2^{n+1}$;而后,基于 CSD 方法的复杂度被进一步优化至 $\frac{1}{2} 4^n$ ^[24].与此同时,基于量子香农分解 (QSD) 的方法亦被提出^[25],达到了 $\frac{23}{48} 4^n$ 复杂度,是目前已知最有效的方法.

表2列出了目前若干编译器中采用的分解方法,其中 Euler 和 KAK 方法可视为 QSD 方法在单比特和两比特情形下的特例^[26],只是编译器实现时针对硬件支持门集做了特殊优化.除使用通用分解方法外,这些编译器还预定义了大量常用的门,其分解依靠事先编写的规则,由于利用了具体门的信息,分解性能有进一步的提升.

表 2 若干编译器采用的 n 比特量子门分解方法

编译器	$n \geq 3$	$n = 2$	$n = 1$
IBMQ Qiskit ^[27]	QSD	KAK	Euler
Google Cirq ^[28]	QSD (仅 $n = 3$)	KAK	Euler
Rigetti Qulic ^[29]	CSD、QSD	KAK	Euler

以上所有方法均为精确分解方法,对于当下量子计算机而言,它们分解产生的门数过多,且要求门参数可取任意精度,这加剧了噪声对量子计算的干扰.近期有不少工作^[30-32]在近似分解方面进行探索,其核心思想在于通过牺牲分解的精度,降低分解所用门数以及量子电路深度,提升量子电路在嘈杂量子计算机上执行的正确率.这些工作利用带待定参数的电路模板,以最小化模板所代表的操作与目标门之间的不保真度为目标调优待定参数,实现近似分解.文献[32]指出,近似分解手段可平均提升电路正确率 6%–30%.

除任意量子门分解外,相关研究也考虑具体类型的量子门分解.由于限制了门类型,这些研究通常考虑更多评价指标和分解场景设定.以 n 比特泛化托佛利 (Toffoli) 门为例 (n 比特泛化 Toffoli 门指的是具有 $n-1$ 个控制比特位、1 个目标比特位的受控非门, $n=2$ 即 CNOT 门),评价指标除了分解所需的基础门数外,往往还包括分解后电路的深度,在分解场景设定方面,相关工作也考虑额外引入辅助量子比特等情形.具体而言,在不使用额外辅助比特的情况下,文献[21]给出了一种门数和电路深度均为 $O(n^2)$ 的方案,后续研究进一步缩小了 n^2 前的系数^[33,34] (从 48 缩小至 2).辅助比特的引入可有效降低门数和电路深度:即使只引入 1 个辅助比特,门数和深度均可降低至 $O(n)$ ^[21],通过引入更多的辅助比特,这些指标还可进一步降低^[35,36],不过量级仍为 $O(n)$.文献[37]在引入辅助比特的基础上进一步引入测量操作,将电路深度降低至 $O(\log n)$.只是辅助比特和测量均代表了额外的资源消耗,孰优孰劣需要视具体情况而定.

2.2 量子比特映射与路由

量子电路在构建时采用的是逻辑量子比特,如需运行于真实硬件上,则要将逻辑量子比特映射为硬件上的物理量子比特,这称为量子比特映射.此外,逻辑量子比特被认为任意两个比特间均可执行两比特门,然而,实际硬件往往只允许在特定量子比特间执行两比特门,故需要采用某些手段动态改变量子比特的映射关系,使不可执行的门变得可行,这称为量子比特路由(许多文献将“量子比特映射与路由”简称为“量子比特映射”).尽管改变映射关系的方法有很多,但绝大多数工作采用插入 SWAP 门的方法,本文也主要考虑这种路由方式.读者可再次参考图 6 了解量子比特映射与路由过程.

本文统计了近年来的相关工作并将其从“优化目标”和“所用方法”两个维度进行分类,结果如表 3 所示.以下先对这两个维度作详细阐述,而后分析相关工作.

优化目标可分为“最小化插入的 SWAP 数目或电路深度”和“最大化电路保真度”两类,二者的根本区别在于是否显式地考虑各种错误对量子计算的影响.针对 SWAP 数目或电路深度的工作认为,门数与量子门错误对计算的影响、电路深度与退相干对计算的影响均具有正相关关系,降低二者可等效地提升计算正确率.这一观点虽然正确,但其指导下的优化是间接的.另一类工作则显式考虑各种错误,典型的做法是通过每种操作的错误率估计整个电路执行时的保真度(正确率),并将保真度作为直接优化目标.

表 3 量子比特映射与路由相关工作分类

所用方法	优化目标	
	SWAP数目/电路深度	电路保真度
最优化	文献[38]: 动态规划 文献[39]: 时序规划、约束规划 文献[40]: 整数线性规划	文献[11,41]: 可满足性模理论 (satisfiability modulo theory)
启发式	文献[38]: 距离指标贪心策略 文献[42]: 深度指标贪心策略 文献[43]: 多种指标贪心策略 文献[44]: A*搜索	文献[45]: A*搜索 文献[41]: 贪心策略 文献[46]: 子图匹配

所用方法可分为“最优化方法”和“启发式方法”两类.最优化方法将映射与路由问题转化为最优化问题,而后采用相应的最优化算法求解.这类方法的核心在于问题转化,通常包括3个步骤^[41,47]:1)问题变量化:将量子比特映射关系、量子门执行时间、各种操作错误率等以变量形式表示;2)约束表达:构造等式、不等式或逻辑关系式等表达硬件对两比特门执行的约束以及量子门之间的依赖;3)构造优化目标:使用变量表达优化目标,通常需要一些额外变换以兼容求解算法.启发式方法则通常遵循“启发式代价函数+搜索”的模式,核心点在于代价函数的构造.以下,以方法维度为视角分析相关工作.

文献[38]依照量子门依赖关系逐门解决映射与路由问题,基于动态规划给出了一个可使SWAP数目最小化的最优解算法,该算法具有指数级复杂度.然而,SWAP最小化已被证明是NP完全问题,这一复杂度已经最优.文献[39]将SWAP数目最小化问题转化为时序规划和约束规划问题,并比较了两种转化的优劣:时序规划求解快但所需SWAP多,约束规划反之.为了综合二者优势,该工作进一步提出了一种混合算法,将时序规划的解作为约束规划的初始解,获得了求解时间和求解质量上的双重提升.文献[40]则以电路深度为优化目标,基于整数线性规划进行求解.而在以电路保真度为目标的相关工作中,文献[11,41]综合考虑了量子比特的相干时间、门的执行时间和错误率以及测量错误率,将电路保真度建模为所有过程(量子门、测量、量子比特闲置)的正确率之积,采用可满足性模理论进行求解.对于上述工作的最优性,[38]进行了最优性证明,其他工作的最优性取决于对应的最优化问题求解器的最优性.

在基于启发式方法的工作中,文献[38,42,43]均采用贪心策略最小化SWAP数或电路深度,主要区别在于代价函数的设计.其中文献[43]提出多种代价函数,设计理念囊括文献[38,42]的思想,并引入了前瞻、双向贪心等技术,所得解较优.这些贪心方法考虑的级别在量子门,文献[44]提出对量子电路分层,通过A*算法搜索在层间插入SWAP门的方案,以最小化SWAP数目.在此工作基础上,文献[45]将量子门的错误率引入到A*代价函数中,在IBMQ机器上实测发现获得了约2倍的电路保真度提升.同样以电路保真度为目标,文献[41,46]则着力改进初始映射:文献[41]采用贪心策略逐次选取单比特门或两比特门错误率最低的量子比特,直到所有量子比特映射完毕;而文献[46]基于已有映射与路由算法给出的映射,在硬件拓扑上寻找该映射的同构子图,从中选择最优者.在上述方式给出初始映射后,两个算法的路由过程均采用最小化SWAP数目的策略,未再进一步考量操作错误率.尽管如此,初始映射的优化依旧有效地提升了电路保真度.

2.3 量子电路变换优化

量子电路变换优化的核心是等价性规则,图7列出了若干种常见的等价性规则,可分为消除规则、合并与分解规则、交换规则等.以下介绍利用这些规则可进行的变换优化.

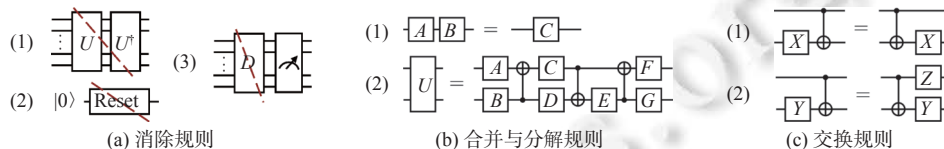


图7 量子电路变换优化可用的等价性规则

利用消除规则可进行量子电路的去冗,以减少操作数目,降低操作错误对量子计算的影响.图7(a)列出了3类常见的消除规则.(1)量子门的消除:当相邻的两个量子门互为共轭转置关系时,这两个量子门的等效效果为恒等变换($UU^\dagger = I$),恒等变换不改变量子比特状态,因而可消除这两个量子门.常见的例子有单比特门 $HH = I$ 、两比特门 $CNOT_{a,b}CNOT_{a,b} = I_{a,b}$ 等.(2)重置操作的消除:重置操作将量子比特状态置为 $|0\rangle$,如量子比特本身状态已经为 $|0\rangle$,则其后的重置操作可消除.重置操作通常用于量子电路执行期间重置辅助量子比特,由于重置操作耗时长,消除该操作能够大幅减少量子电路执行时间,减缓退相干错误.(3)测量前的对角门消除: n 比特对角门指的是在 $|i_1 i_2 \dots i_n\rangle$ 基上矩阵表示是对角阵的量子门,其中 $i_1 i_2 \dots i_n$ 遍历所有 n 位0、1组合,施加对角门后量子比特的状态在 $|i_1 i_2 \dots i_n\rangle$ 上的幅值分布不发生变化,因而施加前后测量结果一致,可消除该对角门.

图7(b)列出了两类合并与分解规则,分别对应单比特和两比特门.单比特门规则往往用于合并连续的门,以

减少量子门数目, 少有见到对单比特门拆分的情形, 这是由于目前的量子计算机能够执行任意单比特门, 所以拆分执行并没有意义. 而对于两比特门规则, 合并与分解通常是一同使用的. 例如, 一种典型的优化手段是首先合并两个量子比特上连续的若干个量子门, 形成一个两比特门 U , 而后再对 U 进行分解. 这样做的目的是, 由于任意的两比特门至多采用 3 个 CNOT 即可实现^[48], 当原本两个量子比特上连续的量子门使用多于 3 个 CNOT 时, 上述合并再分解的方法可以减少 CNOT 门的使用.

图 7(c) 列出了两类交换规则: (1) 是满足 $AB = BA$ 的严格交换规则. (2) 则允许交换后产生额外的单比特门, 称为泛化交换规则^[14]. 与上面减少量子门数目的规则不同, 交换规则的核心在于打破量子门之间的依赖性, 这种能力往往会带给其他优化更大的优化空间. 这里举 3 个典型应用. (1) 交换规则通常与其他规则一并使用: 例如, 综合使用交换与消除规则, 可以交换量子门以满足原本不满足的消除条件, 执行消除. (2) 交换规则也可用于错误减缓: 例如文献 [14] 利用交换规则重排量子门, 消除同时执行 CNOT 门的模式, 减缓串扰错误. (3) 在量子门调度方面, 交换规则对依赖性的打破可提升量子门执行的并行度^[42,49], 这点将在后续“量子门调度”一节展开叙述.

2.4 量子门调度

量子门调度旨在安排各个量子门的执行时间. 调度的一个基本目标是最小化量子电路的总执行时间, 这有利于减缓退相干对计算的影响. 达成此目标的关键在于挖掘量子门执行的并行性. 一个基本的策略是采用“并行调度”, 即最大化每时刻执行的量子门数目, 该策略是 Qiskit、Quilc 等编译器的默认策略.

然而, 并行调度严格遵循量子电路中量子门的依赖关系, 为了进一步提升并行度, 需要打破该依赖关系, 交换性被广泛用于实现这一目标. 文献 [42] 利用量子门间的交换性 (如图 7(c)(1) 所示), 在去除了原本电路中不必要的依赖关系后采用“并行调度”策略进行调度. 文献 [49] 则进一步探索程序块间的交换性 (程序块即若干个量子门合并成的整体, 其可视为一个大的量子门, 交换性通过定义判定), 在程序块层面执行“并行调度”策略.

不过, 当考虑目前量子计算机存在的错误时, 一味地提升并行度往往并非最优选择: 并行度提升在减缓退相干的同时也加剧了串扰的影响, 文献 [50] 指出, IBMQ 机器同时执行 CNOT 操作可导致门错误率增加数十倍. 因而文献 [50] 提出的调度器适当地在量子电路中插入“屏障 (barrier)”, 以串行执行某些量子门. 文献 [51] 则发现串扰与量子比特频率之间存在关系, 可通过调节频率改变串扰的影响, 其提出的调度器同时调控量子门执行的并行度和量子比特频率, 以取得退相干和串扰之间的平衡.

2.5 小结

量子计算硬件的特点极大影响了量子计算编译器的设计. 由于硬件可直接执行的门类型有限, 编译器需要对不可执行的量子门进行分解; 由于计算 (门) 直接发生在数据 (量子比特) 上, 编译器需要建立逻辑比特与物理比特之间的映射; 由于硬件上只有部分量子比特间可执行两比特门, 编译器又需要通过路由手段动态改变映射关系. 此外, 由于近期量子计算机错误繁多, 提升电路执行保真度成为了编译优化的首要目标. 上述种种塑造了量子计算编译器与经典计算编译器的主要不同, 也使得编译器研究成为近期量子计算系统软件研究的一个重要方面.

3 量子计算运行时系统

编译好的量子程序将交由运行时系统执行. 近期的量子计算运行时系统主要功能有两个: 一是负责协调各个模块完成程序执行的整个过程. 例如, IBM Qiskit 的运行时系统提供了向云端量子计算机提交程序和从云端获取执行结果的功能: 编译好的量子程序经由运行时系统打包后发送至云端, 执行完毕后可向云端询问执行结果. 另一个主要功能是实现某些运行策略, 即执行一个给定量子程序的方式. 在嘈杂中规模量子时代, 这些运行策略对降低硬件错误对程序执行的影响而言尤为重要, 本节主要对不同的运行策略进行分析比较.

运行时系统和量子电路的通用交互模式如图 8(a) 所示, 即在量子电路执行前、中、后均有交互发生. 在嘈杂中规模量子时代, 由于量子比特的相干时间较短, 在电路执行过程中进行交互难以实现, 故现有的系统主要在执行前后进行交互 (见图 8(b)), 所实现的运行策略也遵循“执行前处理电路+执行后处理结果”的模式. 依据所基于的理论, 现有运行策略可分为“基于统计方法”和“基于物理理论”两类. 表 4 总结了近期主要的量子程序运行策略, 以下展开分析.

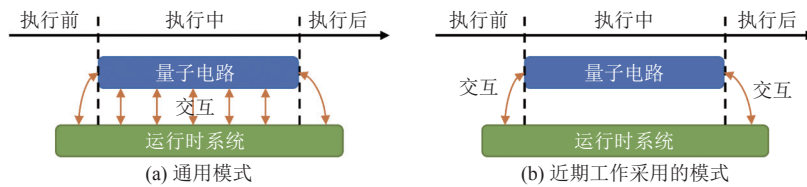


图8 运行时系统和量子电路的交互模式

表4 近期主要的量子程序运行策略

运行策略	简介	优缺点	相关工作
极大似然估计	多次执行给定电路,取测量结果最大频率者为最终结果	适用于随机错误,不适用系统性错误	基础方法
基于统计方法	挖掘错误与可控因素的相关性	在可控因素的多个不同取值下分别执行给定电路,综合这些电路的结果	适用于系统性错误
构建错误的统计模型	预先构建错误的统计模型,利用模型处理测量结果	黑盒方法,对错误类型不敏感;但泛化能力有待提升	[54-57]
外推至零噪声极限 (zero-noise extrapolation)	依据给定电路,构建拥有不同噪声水平的电路,外推这些电路的结果至零噪声水平	电路构建简单;适用于求某个可观测量的均值	[58-60]
基于物理理论	概率错误消除 (probabilistic error cancellation)	对于给定电路,采样错误模型构造多个电路,分别执行并综合结果	适用于求某个可观测量的均值,白盒方法,所需错误模型较难构造
电路切分	切分给定电路为多个小电路,分别执行并综合结果	综合结果的时空开销较大	[62,63]

3.1 基于统计方法的运行策略

一种基础的运行策略是多次执行给定电路,统计每次执行的结果,选取出现频率最大者为最终结果.这是极大似然估计的思想,可以有效消除随机错误的影响,然而对于系统性错误无能为力.由于该策略简单通用、无需修改给定电路,目前多数系统,如 IBM Qiskit、Google Cirq 等的运行时系统,均默认采用之.此外,该运行策略还是其他策略的基础:所有其他策略在执行某一特定电路时,采用的都是极大似然估计方法.

为了减轻系统性错误的影响,研究人员尝试挖掘错误与可控因素之间的相关性,通过调控可控因素,产生不同表现的系统性错误,从而将系统性错误“随机化”,以使得可采用类似“极大似然估计”的方法进行消除.例如,文献 [52] 发现,同一种量子比特映射与路由方法往往会导致特定模式的错误结果,因而研究人员提出编译时采用不同的映射与路由方法,生成多个电路,然后加权平均这些电路的执行结果分布,以消除只用同一种方法带来的系统性错误.再如,文献 [53] 指出,量子比特的测量错误率与其状态有关,测量 $|1\rangle$ 态出错的概率比测量 $|0\rangle$ 态大,故研究人员提出可在测量前适当翻转量子比特状态.然而由于事前不知晓哪个量子比特状态为 $|1\rangle$ (否则就知道了计算结果),研究人员进一步提出了随机翻转和模式翻转等策略,这些策略实质上是在“随机化”测量中的系统错误.

除了分析具体错误的类型、起因,针对性地减缓其影响外,也有不少工作通过构建统计模型,以黑盒方法减轻错误的影响.例如,针对测量错误,IBMQ 系统采用了一种基于贝叶斯定理的减缓方法^[54],其先构建“测量 $|i\rangle$ 态得到结果 j ”的先验概率,而后根据电路的实际测量结果,依据贝叶斯定理反推出后验结果(即无错误发生时的结果).再如,文献 [55-57] 为电路执行期间发生的所有错误的综合效果构建统计模型,理论上该模型在输入实际测量结果时可输出无错误发生时的结果.这些工作均采用机器学习模型,主要区别在于训练所用电路类型或电路特征不同.在电路类型方面,文献 [55,56] 均采用随机电路,而文献 [57] 采用由 Clifford 门构建的电路,由于 Clifford 门可被经典有效地模拟,所以文献 [57] 可适用的电路规模更大.在电路特征方面,文献 [55] 将电路以类似图片的方式编码,文献 [56] 则着重使用了给定电路的逆电路中的特征.综合而言,这些基于错误统计模型的方法只关心错误的表现,而不深究导致错误的原因,理论上对错误类型不敏感,但在实际应用中表现欠佳,泛化能力有待进一步提升^[55].

3.2 基于物理理论的运行策略

除了在统计意义上研究噪声外,研究者还从量子计算的物理理论出发,设计了一系列减缓错误的运行策略.外

推至零噪声极限 (zero-noise extrapolation) 和概率错误消除 (probabilistic error cancellation) 是求某个可观测量 (observable) 的均值时常用的两种运行策略 (求可观测量均值通常用于模拟量子系统, 其与其他量子计算任务的主要区别在于, 对于其他任务而言, 电路的某一具体输出结果是问题的解, 而量子系统模拟需要电路输出结果分布的均值). 电路切分将给定电路切分成多个量子比特数更少的电路, 它们可在小规模量子计算机上执行, 由于当下小规模量子计算机的噪声水平较之大规模者往往更低, 如此执行也可一定程度减缓错误的影响.

外推至零噪声极限的理论基础是, 量子电路在某噪声水平 λ 下的执行结果 $E(\lambda)$ 与电路执行的理想结果 E^* 和噪声水平 λ 具有关系 $E(\lambda) = E^* + \sum_{k=1}^n a_k \lambda^k + R_{n+1}(\lambda)$, 其中 $R_{n+1}(\lambda)$ 是 λ 至少 $n+1$ 阶的高阶余项, 通过一系列不同噪声水平下的电路执行结果 $E(\lambda_i)$, 可拟合上式, 并获得理想结果 E^* . 因而该运行策略依据给定电路, 构建拥有不同噪声水平但功能与给定电路一致的多个电路, 外推这些电路的结果得到理想结果. 文献 [58,59] 提出了这一方法, 读者可参考作深入了解, 文献 [60] 在实验上进行了验证. 该运行策略的优点在于电路构建简单, 便于实验实现, 但其对噪声做了弱噪声和马尔可夫性假设, 导致外推主要在低深度电路的情况下有效 [59,61].

概率错误消除的理论基础是, 一个量子电路执行的理想结果 E^* 可写作一系列量子电路 α 在有噪声情况下执行结果 $E(\alpha)$ 的概率组合 $E^* = \gamma \sum_{\alpha \sim P(\alpha)} E(\alpha)$, 其中 γ 是一个常量, $P(\alpha)$ 是一个概率分布, 二者与给定电路和噪声相关. 该运行策略对于给定电路, 采样 $P(\alpha)$ 生成多个量子电路 α , 求和这些电路结果得到理想结果. 文献 [59] 给出了概率错误消除的理论基础和特殊情况下电路采样的方式, 文献 [61] 结合量子门集层析 (gate set tomography) 和完全集分解等方法给出了局部马尔可夫噪声条件下电路采样的方式.

电路切分的理论基础如图 9(a) 所示, 量子电路中的任一量子比特在不执行操作时可作切分, 切分结果包含 8 组电路, 每组电路包含一个对切分点之前电路的测量和一个对切分点之后电路的重新初始化 [62]. 图 9(b) 给出了一个切分示例, 对 \times 处进行切分后, 原本需要 5 个量子比特执行的电路现在只需要在 3 个量子比特上执行即可 (只是还需要后处理子电路的结果). 当小规模量子计算机的噪声水平较大规模者较低时, 采用这种运行策略执行电路可获得更高的正确率. 然而, 该策略的主要缺点在于后处理的时空开销较大: 不当的切分会使后处理的时间开销随切分次数呈指数级增加, 后处理的空间开销则是原电路量子比特数目的指数倍. 文献 [63] 提出将切分问题转换为混合整数规划问题求解, 以最小化后处理的时间开销; 对于空间开销, 其提出了一种动态查询方法, 不再一次性算出所有测量结果的概率, 而是聚焦于概率前 k 大的结果, 允许调整 k 以平衡结果精度和空间规模.

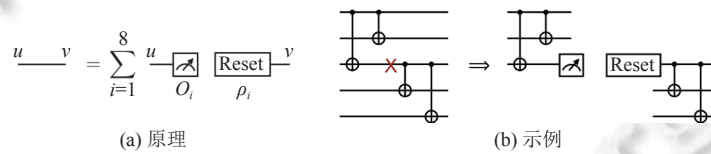


图 9 量子电路切分

3.3 小结

由于编译器已经解决了量子电路在硬件上运行的种种限制, 运行时系统则更加纯粹地思考如何提升电路执行的正确率. 此外, 相比于编译器的静态决策而言, 运行时系统可以利用运行时信息, 从而在错误减缓方面的优化空间更大. 然而, 受制于近期量子计算机有限的相干时间, 现有运行时系统尚未在电路执行过程中与之交互. 随着硬件发展, 相干时间提升后, 运行时系统定会更加多样, 策略也会更加有效.

4 量子计算调试器

调试器旨在定位与纠正编程错误. 近期量子计算调试器相关研究主要集中于对量子比特状态的断言, 具体可分为对状态类型和“值”的断言.

4.1 断言量子比特状态类型

对于量子比特状态类型, 通常有两种分类角度.

(1) 从状态叠加的角度分类, 量子比特状态可分为经典态 (classical state) 或叠加态 (superposition state). 如第 1

节所述,对于 n 个量子比特组成的系统,其状态可表示为 $|\psi\rangle = \sum_{i_1 i_2 \dots i_n} a_{i_1 i_2 \dots i_n} |i_1 i_2 \dots i_n\rangle$, 其中 $i_1 i_2 \dots i_n$ 遍历所有 n 位 0、1 组合. 当只有一个系数 $a_{i_1 i_2 \dots i_n}$ 为 1 时 (此时,其他系数均为 0), 称 $|\psi\rangle$ 为经典态, 因在该状态下, 每个量子比特或 $|0\rangle$ 或 $|1\rangle$. 若至少两个系数 $a_{i_1 i_2 \dots i_n}$ 非 0, 则称 $|\psi\rangle$ 为叠加态, 视为对经典态的叠加.

(2) 从子系统分解的角度分类, 量子比特状态可分为纠缠态 (entangled state) 或乘积态 (product state 或 separable state). 记由 m 个子系统组成的系统的状态为 $|\psi\rangle$ (每个子系统可包含若干量子比特, 不同子系统的量子比特无交集), 若存在 m 个状态 $|\psi_i\rangle$, 其中 $|\psi_i\rangle$ 是第 i 个子系统的状态, 使得 $|\psi\rangle = \otimes_{i=1}^m |\psi_i\rangle$, 即 m 个子系统各自状态的张量积, 则称 $|\psi\rangle$ 为乘积态, 否则为纠缠态.

一种判定量子比特状态类型的方法是“统计断言”^[64], 其主要思想是大量测量待断言状态, 得到测量结果分布, 利用不同状态类型下分布的特点, 基于统计推断证实或否定断言. 具体如下.

1) 经典态下测量结果的分布应是单峰的, 且峰值位于预期状态处; 叠加态下分布应集中在预期的几个经典态上. 因而, 对于经典态和叠加态的判定, 可采用假设检验的方法, 如利用卡方检验, 判断实测分布与期望分布的偏离程度, 从而完成断言.

2) 纠缠态下不同子系统的测量结果具有相关性, 而乘积态下则相互独立. 因而, 这二者可利用列联表对测量结果进行相关性检验来区分.

4.2 断言量子比特状态的“值”

经典计算中, 我们常常会断言变量是否取某一特定值; 类似地, 在量子计算中, 我们期望断言系统状态 $|\psi\rangle$ 是否为特定状态 $|\psi_s\rangle$. 更一般地, 记系统状态空间为 \mathcal{H} , 我们期望断言系统状态是否属于状态空间的某一子空间, 即 $|\psi\rangle \in \mathcal{S}$, $\mathcal{S} \subset \mathcal{H}$. 在这种语义下, 对特定状态 $|\psi_s\rangle$ 的断言等价于断言 $|\psi\rangle \in \mathcal{S} = \text{span}\{|\psi_s\rangle\}$.

相关研究中实现对状态“值”断言的方法有两类: 基于投影测量 (projective measurement)^[65,66] 和基于非破坏性状态甄别 (non-destructive discrimination)^[66,67].

4.2.1 基于投影测量

给定状态空间 \mathcal{H} 的子空间 \mathcal{S} , 任意状态 $|\psi\rangle \in \mathcal{H}$ 都可唯一分解为 $|\psi\rangle = |\psi_{\mathcal{S}}\rangle + |\psi_{\mathcal{S}^\perp}\rangle$, 其中 $|\psi_{\mathcal{S}}\rangle \in \mathcal{S}$, $|\psi_{\mathcal{S}^\perp}\rangle \in \mathcal{S}^\perp$. 定义投影算子 $P_{\mathcal{S}}: \mathcal{H} \rightarrow \mathcal{S}$ 为 $P_{\mathcal{S}}|\psi\rangle = |\psi_{\mathcal{S}}\rangle$, 则对于所有 $|\psi_{\mathcal{S}}\rangle \in \mathcal{S}$, 均有 $P_{\mathcal{S}}|\psi_{\mathcal{S}}\rangle = |\psi_{\mathcal{S}}\rangle$, 即它们在投影算子的作用下保持不变. 依据上述性质, 可构造投影测量 $M_{\mathcal{S}} = \{P_{\mathcal{S}}, I - P_{\mathcal{S}}\}$, 在该测量下, 所有 $|\psi_{\mathcal{S}}\rangle \in \mathcal{S}$ 均测得 0, 因而通过验证测量结果是否为 0 可以判定断言是否成立^[65]. 在状态 $|\psi\rangle$ 上执行投影测量 $M = \{P_0, P_1, \dots, P_{n-1}\}$, $\sum_i P_i = I$: 测量结果为 $0, 1, \dots, n-1$ 之一, 测得结果 i 的概率为 $|P_i|\psi\rangle|^2$, 测量后状态塌缩到 $\frac{P_i|\psi\rangle}{|P_i|\psi\rangle}$. 详细解释请参考文献 [5].

在近期量子计算机上实现投影测量方法需要解决两个问题. 一是近期量子计算机只允许在 $|i_1 i_2 \dots i_n\rangle$ 上的测量 ($i_1 i_2 \dots i_n$ 遍历所有 n 位 0、1 组合), 断言所需的测量 $M_{\mathcal{S}} = \{P_{\mathcal{S}}, I - P_{\mathcal{S}}\}$ 需要转换到 $|i_1 i_2 \dots i_n\rangle$ 上. 二是断言 (及其所需测量) 往往发生在计算过程中, 但近期多数量子计算机并不支持计算过程中测量, 为了断言后能够继续计算, 需要引入一些额外机制. 以下进一步阐述这两个问题的解决方法.

测量转换的实质为空间基的变换. 假设存在一个变换 U , 使得 \mathcal{S} 的基转换为 $|i_1 i_2 \dots i_n\rangle$, 则对状态 $|\psi\rangle$ 的投影测量 $M_{\mathcal{S}}$ 等价于对状态 $|\psi\rangle$ 先施加变换 U 后, 再在 $|i_1 i_2 \dots i_n\rangle$ 上进行测量. 该方法的电路示意图如图 10 所示, 其中 U^\dagger 的作用是在断言后恢复原状态 $|\psi\rangle$, 以便可以继续计算. U 的构造依据 \mathcal{S} 的维数分为 3 种情况.

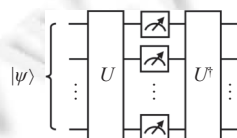


图 10 基于投影测量进行断言的流程

(1) 当 $|\mathcal{S}| = 2^m, m \in [0, n-1]$ 时, 一种构造方法如图 11 所示, 即将 \mathcal{S} 的基映射到前 2^m 个 $|i_1 i_2 \dots i_n\rangle$, 将 \mathcal{S}^\perp 的基映射到其余的 $|i_1 i_2 \dots i_n\rangle$, 这样, 当测量得到前 2^m 个 $i_1 i_2 \dots i_n$ 结果时, 即认为原状态 $|\psi\rangle \in \mathcal{S}$ 、断言成立; 否则失败.

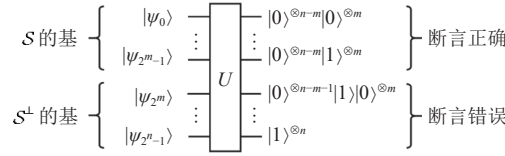


图 11 基于投影测量的断言方法中 U 的构造 ($|\mathcal{S}|=2^m, m \in [0, n-1]$)

(2) 当 $2^{m-1} < |\mathcal{S}| < 2^m, m \in [0, n-1]$ 时, 可以寻找多个包含 \mathcal{S} 的空间 $\{\mathcal{T}_1, \mathcal{T}_2, \dots\}$, 每个 \mathcal{T}_i 的维度均为 2^m , 并且这些空间之交为 \mathcal{S} . 这样, 即可利用 $|\mathcal{S}| = 2^m$ 时的方法, 为 \mathcal{T}_i 构造相应的变换和断言 $|\psi\rangle \in \mathcal{T}_i$, 原断言成立当且仅当所有的 $|\psi\rangle \in \mathcal{T}_i$ 断言均成立.

(3) 当 $2^{n-1} < |\mathcal{S}| < 2^n$ 时, 可通过引入一个辅助量子比特, 将原状态空间扩展到 2^{n+1} 维, 这样, \mathcal{S} 相对于扩展后的空间属于上面提到的两种情况之一, 可相应处理.

上述构造方法 (图 10) 需要在测量后的状态上继续计算, 对于不支持计算过程中测量的量子计算机, 文献 [66] 提出了一种交换机制, 如图 12(a) 所示, 通过引入 $n-m$ 个辅助量子比特, 将需要测量的状态从原电路中交换出来, 这样测量可以在计算全部完成后进行. 此外, 这种交换机制还可以简化 $2^{n-1} < |\mathcal{S}| < 2^n$ 时的电路构造, 如图 12(b) 所示, 相关细节可参考文献 [66].

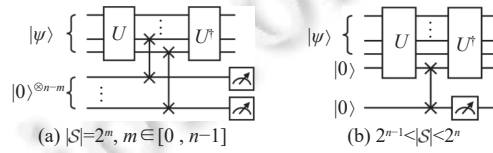


图 12 用于投影测量方法的交换机制

4.2.2 基于非破坏性状态甄别

图 13 显示了非破坏性甄别的流程, 首先在状态 $|\psi\rangle$ 上施加受一个辅助量子比特控制的 U 门, 而后测量该辅助量子比特. 通过构造 U 使之具有 $+1$ 、 -1 两个特征子空间, 上述测量等价于投影测量 $M = \{P_{+1}, P_{-1}\}$ [5], 其中 $P_{\pm 1}$ 是到 ± 1 特征子空间的投影算子. 对于断言 $|\psi\rangle \in \mathcal{S}$, 当构造 U 的 $+1$ 特征子空间为 \mathcal{S} 时, 非破坏性状态甄别的测量结果 0 、 1 可分别指示断言成功与否. 具体而言, 给定断言 $|\psi\rangle \in \mathcal{S}$, 可找到 \mathcal{S} 的一组基 $|\psi_i\rangle$, 和 \mathcal{S}^\perp 的一组基 $|\varphi_j\rangle$, U 可构造为 $U = \sum_i |\psi_i\rangle\langle\psi_i| - \sum_j |\varphi_j\rangle\langle\varphi_j|$.

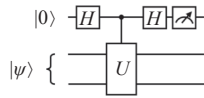


图 13 基于非破坏性甄别进行断言的流程

4.3 不同断言方法的比较分析

表 5 汇总了本节提到的相关工作, 以下从断言后状态、所需辅助量子比特数目、方法复杂度主要来源及出错情况这 4 个角度对它们进行分析比较.

表 5 不同断言方法比较分析

方法	语义	断言后状态	辅助量子比特	复杂度主要来源	出错情况
统计断言	类型	无法用于后续计算	0	大量重复电路执行与测量	假阳性、假阴性
投影测量			0		
交换机制	值	当断言成功时, 状态为期望状态	0~n	U 的构造与实现	假阳性
非破坏性状态甄别			1		

• 断言后状态. 由于统计断言直接在待断言状态上进行测量, 且量子测量会改变被测者状态, 因而断言后的状态无法用于后续计算, 需要从头重新计算. 基于投影测量或非破坏性状态甄别的方法通过构造变换 U , 使得测量等

效为 $M_S = \{P_S, I - P_S\}$, 当断言成功时, 因 $P_S|\psi_S\rangle = |\psi_S\rangle$, 断言后状态为期望状态, 可直接用于后续计算. 可用于后续计算的意义在于, 除了能够避免原本计算中断外, 在程序有多个断言时, 能够允许一次执行即可判定所有断言, 否则需要分别判定每个断言, 计算代价高.

- 所需辅助量子比特数目. 统计断言及不使用交换机制的投影测量方法直接在待断言状态上操作, 无需额外辅助量子比特. 交换机制使用的辅助量子比特数目与 $|S|$ 有关, 当 $|S| = 1$ 时, 最多需要与待断言状态相同数目 (n) 的辅助量子比特. 非破坏性状态甄别固定需要 1 个辅助量子比特.

- 方法复杂度主要来源. 统计断言的复杂度主要来源于统计推断所需的大量测量结果, 每次测量都要从头重新执行电路, 时间代价较高; 此外, 由于统计断言后的状态无法用于后续计算, 当有多个断言时, 要分别判定, 又引入了更多的时间开销. 基于投影测量或非破坏性状态甄别的方法的复杂度主要来源于 U 的构造与实现: 通过 S 的一组基构造 U , 而后利用量子门分解手段将其拆分成硬件可执行的门; 从基构造 U 的复杂度、量子门分解最坏情况下所需的 CNOT 数目, 二者均为所涉及的量子比特数目的指数级.

- 出错情况. 本节提到的所有方法实现的断言并非理想断言, 存在一定概率出错. 统计断言方法采用统计推断, 结果是在一定置信度下给出的, 可能存在假阳性 (判定结果成立, 但实际不成立) 和假阴性 (判定结果不成立, 但实际成立) 情况. 断言量子比特状态“值”的方法均等效于进行投影测量 $M_S = \{P_S, I - P_S\}$, 由于任意状态 $|\psi\rangle$ 可分解为 $|\psi_S\rangle + |\psi_{S^c}\rangle$, $|\psi_S\rangle$ 分量的存在使得有一定概率测得断言成立结果, 因而会存在“假阳性”情况, 文献 [65] 分析了断言结果的置信区间. 但这些方法不会产生“假阴性”结果, 因为上述分解是唯一的, 如果原本状态 $|\psi\rangle \in S$, 则其分解中 $|\psi_{S^c}\rangle = 0$, 不会测出断言失败结果.

4.4 小结

经典计算中, 变量的状态是完整可知的, 但在量子计算中, 量子测量只能获知量子比特状态的部分信息, 这一差异导致了经典计算中容易实现的断言在量子计算中需要花费许多努力. 除断言量子比特状态外, 对过程的调试、对错误的定位等也是编程调试的重要组成部分, 量子计算的特点对它们带来的挑战和可能的解决方案还需要进一步探索.

5 量子计算系统软件研究展望

本节从近期工作涉及的编译器、运行时系统、调试器这 3 个方面出发展望未来量子计算系统软件的研究.

5.1 编译器

- 硬件错误的考虑. 在 4 个主要编译流程中, 量子比特映射与路由、量子门调度的近期相关工作对硬件错误考虑较多; 但量子门分解和量子电路变换优化的工作在这方面较为欠缺. 究其原因, 主要是分解和变换相比映射、调度而言, 对硬件的依赖性不强, 故多数工作在硬件出现之前就已开展, 自然没有考虑当下硬件的错误. 这些工作多以最小化门个数或电路深度为目标, 寻求理论上的最优解, 但对于当下硬件, 这一目标往往并非最优, 以下举两个例子说明. 一是近期量子计算机通常提供多种硬件直接支持的门 (表 1), 但实际上不同类型的门错误率并不同, 不考虑门类型而单纯优化门的个数或电路深度会适得其反 [32]. 二是, 由于噪声和控制误差等原因, 近期量子计算机可实现的门精度有限, 或者说, 两个参数接近的门实现的效果相差不大, 但相关工作往往只考虑精确分解/变换, 为此, 不得不使用许多高精度的门, 这对于近期量子计算而言有害无益. 因此, 探索系统性的优化方法, 综合考虑门的类型、对应的错误率、使用的数目/电路深度等, 并尝试近似分解/变换手段, 可能是近期量子电路编译优化的一个突破点.

- 模块化程序. 同经典程序一样, 量子程序也是模块化的: 包含语句块、过程等语义. 然而, 目前使用的量子电路模型对计算的描述是扁平化的, 即一个操作接着下一个操作, 没有模块化语义. 基于这一模型的研究工作往往直接面向整个电路进行优化, 这样一是全局优化的效率较低、扩展性差, 二是模块化语义的丢失限制了优化能力, 导致优化结果不优. 例如, 量子计算过程中通常需要临时使用辅助量子比特, 使用结束后需要进行反计算 (uncomputation) 来还原辅助量子比特的状态, 这可称为辅助量子比特的“分配”和“释放”. 在没有模块化语义的描述中, 何时分配和

释放通常由编程者在程序编写时指定, 写好后反计算逻辑与其他计算逻辑混在一起, 编译器很难区分并进行优化. 如果采用模块化描述, 则编译器可以根据上下文确定释放时机, 从而在运行时间和辅助量子比特数之间进行权衡^[68,69]. 此外, 随着量子计算机的发展, 日后的量子程序规模会越来越大, 必然需要在模块化程序的语义下讨论编译优化.

- 经典-量子协同. 除了模块化描述缺失外, 量子电路模型也缺乏对经典-量子混合逻辑的描述支持, 如分支、循环等控制流, 经典计算逻辑等. 然而, 许多算法依赖于上述描述, 例如迭代相位估计 (iterative phase estimation)^[70]、量子纠错^[71]、循环至成功 (repeat-until-success)^[72]等. 目前有若干工作正在探索如何支持并优化具有混合逻辑的程序^[73-75], 这些工作多从编程语言的角度入手, 对编译优化的考虑仍较为初步. 在嘈杂中规模量子计算的背景下, 由于量子比特寿命有限, 如何通过编译优化最大程度地降低经典逻辑处理时间, 减少其对量子部分执行的影响是一个重要问题.

5.2 运行时系统

- 运行过程中交互. 目前的运行时系统仅在量子电路执行前后与之交互, 一是因为量子比特的相干时间较短, 二是因为当下多数硬件无法在计算过程中测量, 从而无法在运行时获知电路运行状态, 也就失去了交互的意义. 近期量子计算机逐渐实现了运行中测量功能^[76,77], 使得闭环控制成为可能: 根据实时测量得到的状态, 决定下一步的操作. 这为制定错误减缓策略带来了更大的空间^[78], 然而, 运行中测量同样会发生错误, 闭环控制也会增加电路执行时间, 设计有效的策略仍面临诸多挑战.

- 消除错误模型依赖. 现有运行策略通常依赖对硬件错误的建模, 在本文提到的两类运行策略中, 基于统计方法的策略常常构建错误的统计模型^[54-57], 而基于物理理论的策略甚至需要更精细的错误建模^[59,61]. 使用错误模型会引入 3 个问题: 一是错误模型构建耗时. 例如, 测量具有仅 15 个量子比特的量子计算机的串扰错误率就需要大约 8 h ^[50], 对于稍大规模的量子计算机或更为复杂的错误类型, 建模时间难以接受. 二是模型时效性差. 当下量子计算机硬件错误率随时间变化, 通常 1 天后就需要重新测量^[45]. 三是模型表征不完全. 建模通常基于一些假设和近似, 这就导致模型与实际存在差别. 这些问题阻碍了依赖错误模型的方法在实际中的应用. 机器学习近年来发展迅速, 并且是一种黑盒方法, 有望用来消除错误模型依赖. 近期若干工作在此方向上进行了初步尝试, 如文献^[79]通过监督式机器学习方法绕开错误模型直接构建可实现特定任务、并对错误容忍的量子电路, 文献^[80]通过增强学习方法隐式学习硬件错误并生成错误纠正逻辑. 但这些工作仍较为初步, 效果有待进一步提高.

5.3 调试器

- 断言代价. 一个方法的代价如何影响其是否可在实际中使用, 但目前相关工作缺乏对断言代价的分析研究. 具体来说, 统计断言方法需要大量重复的电路执行与测量, 但断言的精度与执行次数之间的关系并不明确, 这会造成实际应用时不清楚需要执行的次数. 对于基于投影测量或非破坏性状态甄别的方法而言, 目前只明确了各自方法的复杂度 (所涉及量子比特数目的指数级), 但更重要的是比较不同方法实现同一断言时的代价, 从而选取最优方法, 然而尚未有相关分析. 此外, 为了降低断言代价, 文献^[65,66]均提出了扩大目标子空间的策略, 即对于断言 $|\psi\rangle \in \mathcal{S}$, 扩大空间 \mathcal{S} , 但扩大的程度与断言代价之间的关系亦不明确.

- 其他调试类型. 除断言外, 其他调试类型同样重要, 这里举两个可能的研究方向. 一是考虑如何对过程进行调试. 经典计算中, 我们通常将某些常用功能封装成过程以便复用, 这就需要单元测试或形式化验证等手段确保过程的正确性. 量子计算也有这样的需求, 但量子计算的状态空间比经典计算的状态空间更大, 验证过程正确性更加困难. 二是考虑如何定位错误位置. 断言和过程验证仅可判断程序是否出错, 确认出错后还需要定位错误的具体原因和位置以纠正错误. 经典计算中, 我们可以采用跟踪调试并打印变量信息等方式进行定位, 但在量子计算中“打印信息”即获取量子比特状态, 这需要进行量子测量. 然而, 测量会引入更多问题: 一方面测量并不能直接输出量子比特状态 (测量具有随机性), 另一方面测量后量子比特状态会发生变化, 通常无法继续进行计算或尝试其他调试手段. 一种想法是大量复制量子比特、在复制的量子比特上进行测量来规避上述问题, 但不可克隆定理表明在不知道量子比特状态的情况下无法复制该量子比特^[5], 使得这一想法并不可行. 以上种种大幅增加了错误定位的难度.

6 总结

目前量子计算的发展进入了嘈杂中规模量子时代,该时代硬件错误繁多的特点严重限制了量子计算在实际中解决问题的能力。量子计算系统软件上承应用,下接硬件,充分挖掘系统软件在错误减缓方面的潜力对近期实现量子计算的实用价值而言至关重要。本文将近期相关研究工作归入编译器、运行时系统和调试器这3个范畴,并分别进行了分析总结。可以看到,这些工作以提高计算正确性为主要目标,着重对硬件中存在的错误设计相应的减缓策略,并取得了不错的效果。但同时也能看到,量子计算系统软件的发展较为初步,现有工作存在诸多不足,未来仍有较大发展空间。

References:

- [1] Shor PW. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 1997, 26(5): 1484–1509. [doi: [10.1137/S0097539795293172](https://doi.org/10.1137/S0097539795293172)]
- [2] Arute F, Arya K, Babbush R, *et al.* Quantum supremacy using a programmable superconducting processor. *Nature*, 2019, 574(7779): 505–510. [doi: [10.1038/s41586-019-1666-5](https://doi.org/10.1038/s41586-019-1666-5)]
- [3] IBM. IBM quantum. 2022. <https://quantum-computing.ibm.com>
- [4] Preskill J. Quantum computing in the NISQ era and beyond. *Quantum*, 2018, 2: 79. [doi: [10.22331/q-2018-08-06-79](https://doi.org/10.22331/q-2018-08-06-79)]
- [5] Nielsen MA, Chuang IL. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. New York: Cambridge University Press, 2010.
- [6] Huang HL, Wu DC, Fan DJ, Zhu XB. Superconducting quantum computing: A review. *Science China Information Sciences*, 2020, 63(8): 180501. [doi: [10.1007/s11432-020-2881-9](https://doi.org/10.1007/s11432-020-2881-9)]
- [7] Bruzewicz CD, Chiaverini J, McConnell R, Sage JM. Trapped-ion quantum computing: Progress and challenges. *Applied Physics Reviews*, 2019, 6(2): 021314. [doi: [10.1063/1.5088164](https://doi.org/10.1063/1.5088164)]
- [8] O'Brien JL. Optical quantum computing. *Science*, 2007, 318(5856): 1567–1570. [doi: [10.1126/science.1142892](https://doi.org/10.1126/science.1142892)]
- [9] Jazaeri F, Beckers A, Tajalli A, Sallese JM. A review on quantum computing: From qubits to front-end electronics and cryogenic MOSFET physics. In: *Proc. of the 26th Int'l Conf. on Mixed Design of Integrated Circuits and Systems*. Rzeszow: IEEE, 2019. 15–25. [doi: [10.23919/MIXDES.2019.8787164](https://doi.org/10.23919/MIXDES.2019.8787164)]
- [10] Barends R, Kelly J, Megrant A, Sank D, Jeffrey E, Chen Y, Yin Y, Chiaro B, Mutus J, Neill C, O'Malley P, Roushan P, Wenner J, White TC, Cleland AN, Martinis JM. Coherent Josephson Qubit suitable for scalable quantum integrated circuits. *Physical Review Letters*, 2013, 111(8): 080502. [doi: [10.1103/PhysRevLett.111.080502](https://doi.org/10.1103/PhysRevLett.111.080502)]
- [11] Murali P, Linke NM, Martonosi M, Abhari AJ, Nguyen NH, Alderete CH. Full-stack, real-system quantum computer studies: Architectural comparisons and design insights. In: *Proc. of the 46th Int'l Symp. on Computer Architecture*. Phoenix: Association for Computing Machinery, 2019. 527–540. [doi: [10.1145/3307650.3322273](https://doi.org/10.1145/3307650.3322273)]
- [12] Kwon S, Tomonaga A, Lakshmi Bhai G, Devitt SJ, Tsai JS. Gate-based superconducting quantum computing. *Journal of Applied Physics*, 2021, 129(4): 041102. [doi: [10.1063/5.0029735](https://doi.org/10.1063/5.0029735)]
- [13] Sarovar M, Proctor T, Rudinger K, Young K, Nielsen E, Blume-Kohout R. Detecting crosstalk errors in quantum information processors. *Quantum*, 2020, 4: 321. [doi: [10.22331/q-2020-09-11-321](https://doi.org/10.22331/q-2020-09-11-321)]
- [14] Xie L, Zhai JD, Zheng WM. Mitigating crosstalk in quantum computers through commutativity-based instruction reordering. In: *Proc. of the 58th ACM/IEEE Design Automation Conf.* San Francisco: IEEE, 2021. 445–450. [doi: [10.1109/DAC18074.2021.9586145](https://doi.org/10.1109/DAC18074.2021.9586145)]
- [15] Aliferis P, Gottesman D, Preskill J. Quantum accuracy threshold for concatenated distance-3 codes. *Quantum Information & Computation*, 2006, 6(2): 97–165.
- [16] Google. Google devices. 2022. <https://quantumai.google/cirq/google/devices>
- [17] Rigetti. Rigetti quantum computing platform. 2021. <https://qcs.rigetti.com/qpus>
- [18] Wright K, Beck KM, Debnath S, Amini JM, Nam Y, Grzesiak N, Chen JS, Piseni NC, Chmielewski M, Collins C, Hudek KM, Mizrahi J, Wong-Campos JD, Allen S, Apisdorf J, Solomon P, Williams M, Ducore AM, Blinov A, Kreikemeier SM, Chaplin V, Keesan M, Monroe C, Kim J. Benchmarking an 11-qubit quantum computer. *Nature Communications*, 2019, 10(1): 5464. [doi: [10.1038/s41467-019-13534-2](https://doi.org/10.1038/s41467-019-13534-2)]
- [19] IonQ. Best practices for using IonQ hardware. 2022. <https://ionq.com/best-practices>
- [20] Shende VV, Markov IL, Bullock SS. Minimal universal two-qubit controlled-NOT-based circuits. *Physical Review A*, 2004, 69(6): 062321. [doi: [10.1103/PhysRevA.69.062321](https://doi.org/10.1103/PhysRevA.69.062321)]

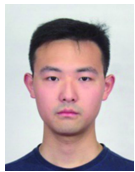
- [21] Barenco A, Bennett CH, Cleve R, DiVincenzo DP, Margolus N, Shor P, Sleator T, Smolin JA, Weinfurter H. Elementary gates for quantum computation. *Physical Review A*, 1995, 52(5): 3457–3467. [doi: [10.1103/PhysRevA.52.3457](https://doi.org/10.1103/PhysRevA.52.3457)]
- [22] Vartiainen JJ, Möttönen M, Salomaa MM. Efficient decomposition of quantum gates. *Physical Review Letters*, 2004, 92(17): 177902. [doi: [10.1103/PhysRevLett.92.177902](https://doi.org/10.1103/PhysRevLett.92.177902)]
- [23] Möttönen M, Vartiainen JJ, Bergholm V, Salomaa MM. Quantum circuits for general multiqubit gates. *Physical Review Letters*, 2004, 93(13): 130502. [doi: [10.1103/PhysRevLett.93.130502](https://doi.org/10.1103/PhysRevLett.93.130502)]
- [24] Möttönen M, Vartiainen JJ. Decompositions of general quantum gates. In: Shannon S, ed. *Trends in Quantum Computing Research*. New York: Nova Science Publishers, 2006.
- [25] Shende VV, Bullock SS, Markov IL. Synthesis of quantum-logic circuits. *IEEE Trans. on Computer-aided Design of Integrated Circuits and Systems*, 2006, 25(6): 1000–1010. [doi: [10.1109/TCAD.2005.855930](https://doi.org/10.1109/TCAD.2005.855930)]
- [26] Drury B, Love P. Constructive quantum Shannon decomposition from Cartan involutions. *Journal of Physics A: Mathematical and Theoretical*, 2008, 41(39): 395305. [doi: [10.1088/1751-8113/41/39/395305](https://doi.org/10.1088/1751-8113/41/39/395305)]
- [27] Qiskit Developers. Qiskit: An open-source framework for quantum computing. 2022. <https://qiskit.org>
- [28] Cirq Developers. Cirq. 2022. <https://quantumai.google/cirq>
- [29] Rigetti Quilc Developers. Rigetti quilc. 2022. <https://docs.rigetti.com/qcs>
- [30] Jones T, Benjamin SC. Robust quantum compilation and circuit optimisation via energy minimisation. *Quantum*, 2022, 6: 628. [doi: [10.22331/q-2022-01-24-628](https://doi.org/10.22331/q-2022-01-24-628)]
- [31] Davis MG, Smith E, Tudor A, Sen K, Siddiqi I, Iancu C. Heuristics for quantum compiling with a continuous gate set. arXiv:1912.02727, 2019.
- [32] Lao LL, Murali P, Martonosi M, Browne D. Designing calibration and expressivity-efficient instruction sets for quantum computing. In: *Proc. of the 48th ACM/IEEE Annual Int'l Symp. on Computer Architecture*. Valencia: IEEE, 2021. 846–859. [doi: [10.1109/ISCA52012.2021.00071](https://doi.org/10.1109/ISCA52012.2021.00071)]
- [33] Szykowski M, Kerntopf P. Low quantum cost realization of generalized Peres and Toffoli gates with multiple-control signals. In: *Proc. of the 13th IEEE Int'l Conf. on Nanotechnology*. Beijing: IEEE, 2013. 802–807. [doi: [10.1109/NANO.2013.6721034](https://doi.org/10.1109/NANO.2013.6721034)]
- [34] Abdollahi A, Saeedi M, Pedram M. Reversible logic synthesis by quantum rotation gates. *Quantum Information and Computation*, 2013, 13(9–10): 771–792. [doi: [10.26421/QIC13.9-10-3](https://doi.org/10.26421/QIC13.9-10-3)]
- [35] Amy M, Maslov D, Mosca M. Polynomial-time T-depth optimization of Clifford+T circuits via matroid partitioning. *IEEE Trans. on Computer-aided Design of Integrated Circuits and Systems*, 2014, 33(10): 1476–1489. [doi: [10.1109/TCAD.2014.2341953](https://doi.org/10.1109/TCAD.2014.2341953)]
- [36] Maslov D. Advantages of using relative-phase Toffoli gates with an application to multiple control Toffoli optimization. *Physical Review A*, 2016, 93(2): 022311. [doi: [10.1103/PhysRevA.93.022311](https://doi.org/10.1103/PhysRevA.93.022311)]
- [37] He Y, Luo MX, Zhang E, Wang HK, Wang XF. Decompositions of n -qubit Toffoli gates with linear circuit complexity. *Int'l Journal of Theoretical Physics*, 2017, 56(7): 2350–2361. [doi: [10.1007/s10773-017-3389-4](https://doi.org/10.1007/s10773-017-3389-4)]
- [38] Siraichi MY, dos Santos VF, Collange C, Pereira FMQ. Qubit allocation. In: *Proc. of the 2018 Int'l Symp. on Code Generation and Optimization*. New York: Association for Computing Machinery, 2018. 113–125. [doi: [10.1145/3168822](https://doi.org/10.1145/3168822)]
- [39] Booth KEC, Do M, Beck JC, Rieffel E, Venturelli D, Frank J. Comparing and integrating constraint programming and temporal planning for quantum circuit compilation. In: *Proc. of the 28th Int'l Conf. on Automated Planning and Scheduling*. Delft: AAAI Press, 2018. 366–374.
- [40] Bhattacharjee D, Saki AA, Alam M, Chattopadhyay A, Ghosh S. MUQUT: Multi-constraint quantum circuit mapping on NISQ computers: Invited paper. In: *Proc. of the 2019 IEEE/ACM Int'l Conf. on Computer-aided Design*. Westminster: IEEE, 2019. 1–7. [doi: [10.1109/ICCAD45719.2019.8942132](https://doi.org/10.1109/ICCAD45719.2019.8942132)]
- [41] Murali P, Baker JM, Javadi-Abhari A, Chong FT, Martonosi M. Noise-adaptive compiler mappings for noisy intermediate-scale quantum computers. In: *Proc. of the 24th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems*. Providence: Association for Computing Machinery, 2019. 1015–1029. [doi: [10.1145/3297858.3304075](https://doi.org/10.1145/3297858.3304075)]
- [42] Guerreschi GG, Park J. Two-step approach to scheduling quantum circuits. *Quantum Science and Technology*, 2018, 3(4): 045003. [doi: [10.1088/2058-9565/aacfb0](https://doi.org/10.1088/2058-9565/aacfb0)]
- [43] Li GS, Ding YF, Xie Y. Tackling the qubit mapping problem for NISQ-Era quantum devices. In: *Proc. of the 24th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems*. Providence: Association for Computing Machinery, 2019. 1001–1014. [doi: [10.1145/3297858.3304023](https://doi.org/10.1145/3297858.3304023)]
- [44] Zulehner A, Paler A, Wille R. An efficient methodology for mapping quantum circuits to the IBM QX architectures. *IEEE Trans. on Computer-aided Design of Integrated Circuits and Systems*, 2019, 38(7): 1226–1236. [doi: [10.1109/TCAD.2018.2846658](https://doi.org/10.1109/TCAD.2018.2846658)]

- [45] Tannu SS, Qureshi MK. Not all qubits are created equal: A case for variability-aware policies for NISQ-Era quantum computers. In: Proc. of the 24th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. Providence: Association for Computing Machinery, 2019. 987–999. [doi: [10.1145/3297858.3304007](https://doi.org/10.1145/3297858.3304007)]
- [46] Ash-Saki A, Alam M, Ghosh S. QURE: Qubit re-allocation in noisy intermediate-scale quantum computers. In: Proc. of the 56th ACM/IEEE Design Automation Conf. Las Vegas: IEEE, 2019. 1–6.
- [47] Dou XL, Liu L, Chen YT. An investigation into quantum program mapping on superconducting quantum computers. Journal of Computer Research and Development, 2021, 58(9): 1856–1874 (in Chinese with English abstract). [doi: [10.7544/issn1000-1239.2021.20210314](https://doi.org/10.7544/issn1000-1239.2021.20210314)]
- [48] Vatan F, Williams C. Optimal quantum circuits for general two-qubit gates. Physical Review A, 2004, 69(3): 032315. [doi: [10.1103/PhysRevA.69.032315](https://doi.org/10.1103/PhysRevA.69.032315)]
- [49] Shi YN, Leung N, Gokhale P, Rossi Z, Schuster DI, Hoffmann H, Chong FT. Optimized compilation of aggregated instructions for realistic quantum computers. In: Proc. of the 24th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. Providence: Association for Computing Machinery, 2019. 1031–1044. [doi: [10.1145/3297858.3304018](https://doi.org/10.1145/3297858.3304018)]
- [50] Murali P, McKay DC, Martonosi M, Javadi-Abhari A. Software mitigation of crosstalk on noisy intermediate-scale quantum computers. In: Proc. of the 25th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. Lausanne: Association for Computing Machinery, 2020. 1001–1016. [doi: [10.1145/3373376.3378477](https://doi.org/10.1145/3373376.3378477)]
- [51] Ding YS, Gokhale P, Lin SF, Rines R, Propson T, Chong FT. Systematic crosstalk mitigation for superconducting qubits via frequency-aware compilation. In: Proc. of the 53rd Annual IEEE/ACM Int'l Symp. on Microarchitecture. Athens: IEEE, 2020. 201–214. [doi: [10.1109/MICRO50266.2020.00028](https://doi.org/10.1109/MICRO50266.2020.00028)]
- [52] Tannu SS, Qureshi M. Ensemble of diverse mappings: Improving reliability of quantum computers by orchestrating dissimilar mistakes. In: Proc. of the 52nd Annual IEEE/ACM Int'l Symp. on Microarchitecture. Columbus: Association for Computing Machinery, 2019. 253–265. [doi: [10.1145/3352460.3358257](https://doi.org/10.1145/3352460.3358257)]
- [53] Tannu SS, Qureshi MK. Mitigating measurement errors in quantum computers by exploiting state-dependent bias. In: Proc. of the 52nd Annual IEEE/ACM Int'l Symp. on Microarchitecture. Columbus: Association for Computing Machinery, 2019. 279–290. [doi: [10.1145/3352460.3358265](https://doi.org/10.1145/3352460.3358265)]
- [54] Qiskit. Measurement error mitigation. 2022. <https://learn.qiskit.org/course/quantum-hardware/measurement-error-mitigation>
- [55] Zlokapa A, Gheorghiu A. A deep learning model for noise prediction on near-term quantum devices. arXiv:2005.10811, 2020.
- [56] Patel T, Tiwari D. Qraft: Reverse your quantum circuit and know the correct program output. In: Proc. of the 26th ACM Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. New York: Association for Computing Machinery, 2021. 443–455. [doi: [10.1145/3445814.3446743](https://doi.org/10.1145/3445814.3446743)]
- [57] Czarnik P, Arrasmith A, Coles PJ, Cincio L. Error mitigation with Clifford quantum-circuit data. Quantum, 2021, 5: 592. [doi: [10.22331/q-2021-11-26-592](https://doi.org/10.22331/q-2021-11-26-592)]
- [58] Li Y, Benjamin SC. Efficient variational quantum simulator incorporating active error minimization. Physical Review X, 2017, 7(2): 021050. [doi: [10.1103/PhysRevX.7.021050](https://doi.org/10.1103/PhysRevX.7.021050)]
- [59] Temme K, Bravyi S, Gambetta JM. Error mitigation for short-depth quantum circuits. Physical Review Letters, 2017, 119(18): 180509. [doi: [10.1103/PhysRevLett.119.180509](https://doi.org/10.1103/PhysRevLett.119.180509)]
- [60] Kandala A, Temme K, Córcoles AD, Mezzacapo A, Chow JM, Gambetta JM. Error mitigation extends the computational reach of a noisy quantum processor. Nature, 2019, 567(7749): 491–495. [doi: [10.1038/s41586-019-1040-7](https://doi.org/10.1038/s41586-019-1040-7)]
- [61] Endo S, Benjamin SC, Li Y. Practical quantum error mitigation for near-future applications. Physical Review X, 2018, 8(3): 031027. [doi: [10.1103/PhysRevX.8.031027](https://doi.org/10.1103/PhysRevX.8.031027)]
- [62] Peng TY, Harrow AW, Ozols M, Wu XD. Simulating large quantum circuits on a small quantum computer. Physical Review Letters, 2020, 125(15): 150504. [doi: [10.1103/PhysRevLett.125.150504](https://doi.org/10.1103/PhysRevLett.125.150504)]
- [63] Tang W, Tomesh T, Suchara M, Larson J, Martonosi M. CutQC: Using small Quantum computers for large quantum circuit evaluations. In: Proc. of the 26th ACM Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. New York: Association for Computing Machinery, 2021. 473–486. [doi: [10.1145/3445814.3446758](https://doi.org/10.1145/3445814.3446758)]
- [64] Huang YP, Martonosi M. Statistical assertions for validating patterns and finding bugs in quantum programs. In: Proc. of the 46th Annual Int'l Symp. on Computer Architecture. Phoenix: IEEE, 2019. 541–553.
- [65] Li GS, Zhou L, Yu NK, Ding YF, Ying MS, Xie Y. Projection-based runtime assertions for testing and debugging Quantum programs. Proc. of the ACM on Programming Languages, 2020, 4(OOPSLA): 150. [doi: [10.1145/3428218](https://doi.org/10.1145/3428218)]
- [66] Liu J, Zhou HY. Systematic approaches for precise and approximate quantum state runtime assertion. In: Proc. of the 2021 IEEE Int'l Symp. on High-performance Computer Architecture. Seoul: IEEE, 2021. 179–193. [doi: [10.1109/HPCA51647.2021.00025](https://doi.org/10.1109/HPCA51647.2021.00025)]

- [67] Liu J, Byrd GT, Zhou HY. Quantum circuits for dynamic runtime assertions in quantum computation. In: Proc. of the 25th Int'l Conf. on Architectural Support for Programming Languages and Operating Systems. Lausanne: Association for Computing Machinery, 2020. 1017–1030. [doi: [10.1145/3373376.3378488](https://doi.org/10.1145/3373376.3378488)]
- [68] Paradis A, Bichsel B, Steffen S, Vechev M. Unqomp: Synthesizing uncomputation in quantum circuits. In: Proc. of the 42nd ACM SIGPLAN Int'l Conf. on Programming Language Design and Implementation. New York: Association for Computing Machinery, 2021. 222–236. [doi: [10.1145/3453483.3454040](https://doi.org/10.1145/3453483.3454040)]
- [69] Ding YS, Wu XC, Holmes A, Wiseth A, Franklin D, Martonosi M, Chong FT. SQUARE: Strategic quantum ancilla reuse for modular quantum programs via cost-effective uncomputation. In: Proc. of the 47th ACM/IEEE Annual Int'l Symp. on Computer Architecture. Valencia: IEEE, 2020. 570–583. [doi: [10.1109/ISCA45697.2020.00054](https://doi.org/10.1109/ISCA45697.2020.00054)]
- [70] Dobšiček M, Johansson G, Shumeiko V, Wendin G. Arbitrary accuracy iterative quantum phase estimation algorithm using a single ancillary qubit: A two-qubit benchmark. Physical Review A, 2007, 76(3): 030306. [doi: [10.1103/PhysRevA.76.030306](https://doi.org/10.1103/PhysRevA.76.030306)]
- [71] Steane AM. Error correcting codes in quantum theory. Physical Review Letters, 1996, 77(5): 793–797. [doi: [10.1103/PhysRevLett.77.793](https://doi.org/10.1103/PhysRevLett.77.793)]
- [72] Paetznick A, Svore KM. Repeat-until-success: Non-deterministic decomposition of single-qubit unitaries. Quantum Information & Computation, 2014, 14(15–16): 1277–1301.
- [73] Fu X, Yu JT, Su X, Jiang HR, Wu H, Cheng FC, Deng X, Zhang JR, Jin L, Yang YH, Xu L, Hu CC, Huang AQ, Huang GY, Qiang XG, Deng MT, Xu P, Xu WX, Liu WW, Zhang Y, Deng YX, Wu JJ, Feng Y. Quingo: A programming framework for heterogeneous quantum-classical computing with NISQ features. ACM Trans. on Quantum Computing, 2021, 2(4): 19. [doi: [10.1145/3483528](https://doi.org/10.1145/3483528)]
- [74] Ittah D, Häner T, Kliuchnikov V, Hoefler T. QIRO: A static single assignment-based quantum program representation for optimization. ACM Trans. on Quantum Computing, 2022, 3(3): 14. [doi: [10.1145/3491247](https://doi.org/10.1145/3491247)]
- [75] Cross A, Javadi-Abhari A, Alexander T, De Beaudrap N, Bishop LS, Heidel S, Ryan CA, Sivarajah P, Smolin J, Gambetta JM, Johnson BR. OpenQASM 3: A broader and deeper quantum assembly language. ACM Trans. on Quantum Computing, 2022, 3(3): 12. [doi: [10.1145/3505636](https://doi.org/10.1145/3505636)]
- [76] Córcoles AD, Takita M, Inoue K, Lekuch S, Mineev ZK, Chow JM, Gambetta JM. Exploiting dynamic quantum circuits in a quantum algorithm with superconducting qubits. Physical Review Letters, 2021, 127(10): 100501. [doi: [10.1103/PhysRevLett.127.100501](https://doi.org/10.1103/PhysRevLett.127.100501)]
- [77] Pino JM, Dreiling JM, Figgatt C, Gaebler JP, Moses SA, Allman MS, Baldwin CH, Foss-Feig M, Hayes D, Mayer K, Ryan-Anderson C, Neyenhuis B. Demonstration of the trapped-ion quantum CCD computer architecture. Nature, 2020, 592(7853): 209–213. [doi: [10.1038/s41586-021-03318-4](https://doi.org/10.1038/s41586-021-03318-4)]
- [78] Botelho L, Glos A, Kundu A, Miszczak JA, Salehi Ö, Zimborás Z. Error mitigation for variational quantum algorithms through mid-circuit measurements. Physical Review A, 2022, 105(2): 022441. [doi: [10.1103/PhysRevA.105.022441](https://doi.org/10.1103/PhysRevA.105.022441)]
- [79] Cincio L, Rudinger K, Sarovar M, Coles PJ. Machine learning of noise-resilient quantum circuits. PRX Quantum, 2021, 2(1): 010324. [doi: [10.1103/PRXQuantum.2.010324](https://doi.org/10.1103/PRXQuantum.2.010324)]
- [80] Fösel T, Tighineanu P, Weiss T, Marquardt F. Reinforcement learning with neural networks for quantum feedback. Physical Review X, 2018, 8(3): 031084. [doi: [10.1103/PhysRevX.8.031084](https://doi.org/10.1103/PhysRevX.8.031084)]

附中文参考文献:

- [47] 窦星磊, 刘磊, 陈岳涛. 面向超导量子计算机的程序映射技术研究. 计算机研究与发展, 2021, 58(9): 1856–1874. [doi: [10.7544/issn1000-1239.2021.20210314](https://doi.org/10.7544/issn1000-1239.2021.20210314)]



谢磊(1996—), 男, 博士生, 主要研究领域为量子计算编译, 错误减缓.



翟季冬(1981—), 男, 博士, 副教授, 博士生导师, CCF 高级会员, 主要研究领域为高性能计算, 编译优化, 性能评测.