

量子比特reset操作支持下的量子程序并行化优化

撰写：叶之帆

讨论与审核：孙晨寅，吴浩洋

全体组员：叶之帆（18级计算机学院，组长），孙晨寅（17级物理学院），吴浩洋（18级信息学院）

量子比特reset操作支持下的量子程序并行化优化

摘要

调研/相关工作

动态的量子比特reset操作

量子机器后端实现：针对批量提交的任务

借助reset操作的并行化优化

时间开销的抽象

研究思路

目的

手段

所有权机制

拟开展工作

物理学依据

量子程序调研

算法调研

建模

Basic Algorithm: 特征提取算法

Fully-connected Algorithm:

Partly-connected Algorithm:

测试集与验证

More

分工

摘要

以往的量子程序优化主要考虑针对单个量子程序的优化，这种优化建立在这样一种模型上：量子机器逐个执行被提交的每个量子程序。

我们考虑的模型是：多个量子程序可以在同一个量子机器上并行执行，以量子比特作为量子机器的基本资源单位，一个量子机器具有的全部量子比特在同一时刻可能被划分给不同的量子程序。当一个量子比特的所有权从一个量子程序转移到另一个量子程序时，采用reset操作重置该量子比特。根据较新的研究，这种reset操作的耗时有望达到量子门操作耗时的相近数量级。

本组考虑的优化手段主要从 量子比特的映射 和 量子程序间执行顺序的重排 两方面着手，试图提高量子机器的并行度。

调研/相关工作

动态的量子比特reset操作

量子比特的初始化是实现量子计算机的一个关键问题，传统的初始化方法是等待一段很长的时间，使量子比特自然回到初始态：

Preparation of a qubit into a well-defined initial state is one of the key requirements for any quantum computational algorithm. The conventional passive initialization protocol relies on the relaxation of the qubit to a thermal state determined by the residual coupling to the environment. This protocol is inherently slow because the relaxation rate has to be minimized to decrease the probability of errors in a coherent quantum computation.

引自: [Natrure: Efficient protocol for qubit initialization with a tunable environment](#)

另一种方法是主动使一个量子比特被 reset 到初始状态 $|0\rangle$ ，一些文章讨论了如何设计一个 reset 协议：

[Demonstrating a Driven Reset Protocol for a Superconducting Qubit](#)

这篇文章讨论的 reset 方法可以在 $3\ \mu\text{s}$ 中 reset 一个量子比特，并且达到至少 99.5 % 的可信度。 $3\ \mu\text{s}$ 的 reset 时间已经和超导体体系下的量子门操作处于同一量级

其他关于动态reset操作的研究：

[Digital Feedback in Superconducting Quantum Circuits](#)

[Fast and Unconditional All-Microwave Reset of a Superconducting Qubit](#)

在 Qiskit 中，已经加入了对单个量子比特的 reset 操作的支持：

[Conditional Reset on IBM Quantum Systems](#)

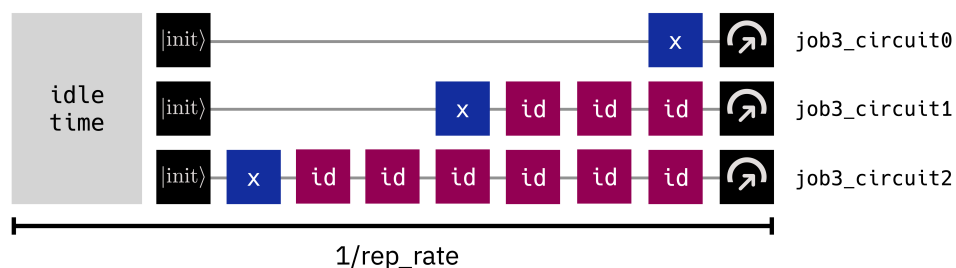
IBM 的动态reset方法是：测量 + 基于测量结果的初始化

The `reset` instruction is now supported in Qiskit on IBM Quantum systems. This allows users to reset qubits into the ground state ($|0\rangle$) with high-fidelity. These instructions are comprised of a not-gate conditioned on the measurement outcome of the qubit.

量子机器后端实现：针对批量提交的任务

IBM 在 [System Circuits Execution](#) 提到量子机器对批量提交的量子程序的执行方式是：

Batched circuit execution



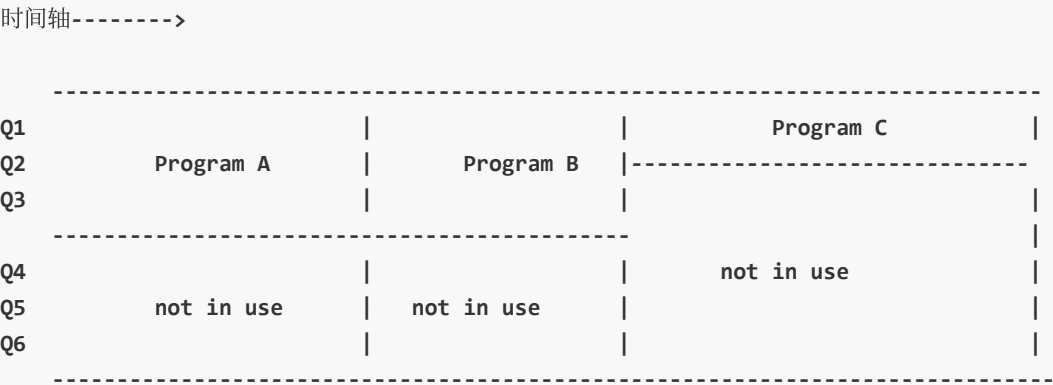
量子机器具有固定的执行速率 `rep_rate`，每一个量子程序在 $1/\text{rep_rate}$ 时间内执行，并且在这些时间内独占量子机器。这种逐个执行的方式阻止了批量提交的量子程序并行执行的可能。

注意到上图中标注了 idle time，idle time 是上一轮测量结束到下一轮执行开始之间的时间，设置较长的 idle time 将帮助将量子比特自然初始化到初态。借助动态reset的操作，无需等待 idle time，我们可以将部分量子比特快速、动态地重新初始化。

我们希望打破这种顺序执行的运行模型，我们建立的模型是：量子程序不必限定在 $1/\text{rep_rate}$ 时间内执行，也不必顺序执行批量提交的量子程序。这要求量子机器支持动态初始化和动态测量操作：我们的测量和初始化操作不限定固定周期，而且每次只对部分量子比特测量和初始化。

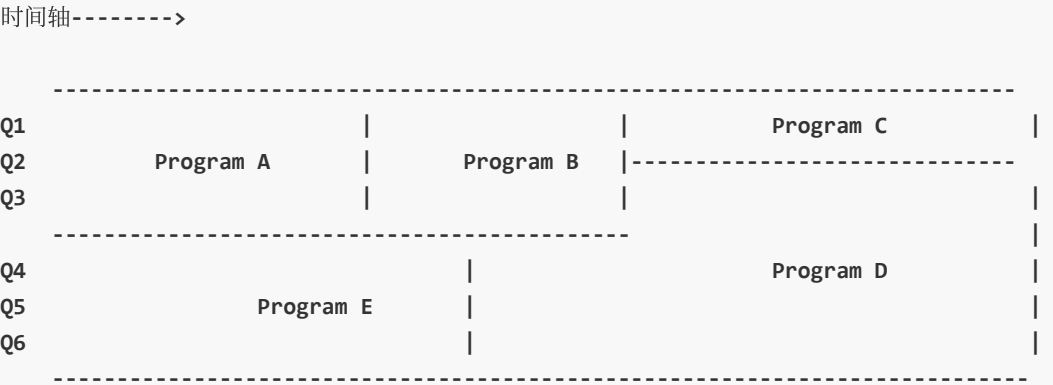
一个设想的执行范例：
{Q1,Q2,Q3,Q4,Q5,Q6}是量子比特资源，“|”处涉及测量和重新初始化的操作

对于提交到量子计算机上的任务**{Program A,Program B,Program C,Program D,Program E}**，传统的执行方式是：



在上面这种执行模型中，每个量子程序独占一段执行时间，即使有闲置的量子比特也不能划分给其他量子程序使用

我们期望的执行方式是：



在这种执行模型中，量子程序的执行时间互不冲突地嵌在一起，充分利用闲置资源

借助reset操作的并行化优化

单个量子程序可能只占用了量子计算机中的一些量子比特，而非全部量子比特。并且，对于量子程序占用的量子比特，每一个量子比特被占用的时间长短也是不尽相同的（它们开始投入使用的时间可能不同）。

基于以上事实，多个量子程序可以并行执行，有效利用空置的量子比特资源。在进行这种并行化优化之前，我们需要提取一个量子程序占用量子比特的特征（占用了多少量子比特；对于占用的每个量子比特，开始使用该量子比特及结束使用该量子比特的时间是怎样的）

时间开销的抽象

由于需要计算量子比特被占用的时间长短，我们必须抽象每一个量子比特操作的执行时间（也包括 reset 操作的时间）

一个可行的抽象是：

参考：[CODAR: A Contextual Duration-Aware Qubit Mapping for Various NISQ Devices](#)

定义量子时钟周期(quantum clock cycle)为 τ ，所有门操作的时间近似为 τ 的整数倍，相同的门操作所用的时间是一个固定值 $n\tau$ ，其中 n 取决于具体的底层实现

研究思路

目的

通过引入 reset 操作，考虑量子程序并行执行和连续执行时的优化，优化对象是一串量子程序（作为工作负载）的总执行时间

手段

开发一个类似于[CODAR: A Contextual Duration-Aware Qubit Mapping for Various NISQ Devices](#)的近似算法，优化量子比特的mapping方案，来达到优化目的；

必要的情况下，可以调整多个量子程序的执行顺序，这对量子程序的正确性不产生影响

这里涉及两次映射优化：

- A)（为了减小错误率和单个量子程序的执行时间）单个量子程序的映射优化
- B)（为了减少总执行时间）多个量子程序的映射问题

我们需要思考这两次优化之间的关系：

- A) 不改变针对单个量子程序的映射优化的结果（或者从单量子程序映射算法提供的多个最优结果中挑选一个），在此基础上做多个量子程序的映射优化，这保证了对现存的各种“单量子程序映射算法”的兼容性
- B) 综合考虑全部优化目标与优化手段，在单个量子程序优化的同时开展跨程序的映射优化

所有权机制

我们提出量子比特的“所有权”概念，量子比特在任一时刻最多只能被一个量子程序所有，量子比特所有权的转移必须涉及一个reset操作。量子程序假定量子比特在被取得所有权的初始时刻是位于 $|0\rangle$ 状态的。

拟开展工作

物理学依据

1. 考察 reset 操作在不同量子计算机体系结构下的开销和最新进展
2. 为算法提供理论支撑
3. 核查、评估算法的正确性与可行性
4. 调查 reset 操作用于初始化的可信度，为量子计算引入了何种误差

量子程序调研

调研一些常见的量子程序，分析程序的特征，特别是使用量子比特的特征（占用所有权的初始时刻、结束时刻）

算法调研

考察现有的各种量子程序优化算法，论证我们开发的跨量子程序的优化算法与单个程序优化算法的兼容性

建模

建立一个良好的抽象模型：抽象量子机器的各项特征（执行各项操作的时间、错误率、底层量子比特联通性）

Basic Algorithm: 特征提取算法

针对一个量子程序，计算它所需要使用的每个量子比特开始被用到的时机和可以被释放的时机，即：被占用所有权的时机

这一步的算法的目的是提取量子程序使用量子比特的特征，为后续优化算法提供必要的信息。

对于全联通假设下的量子机器，由于不需要考虑映射问题，当一个量子程序的OpenQasm给定时，它使用量子比特的特征就已经被确定了。

而对于一个量子比特部分联通的量子机器，给定了一个量子程序的OpenQasm之后，我们还要考虑映射问题以及插入的 SWAP操作，这会影响到量子程序占用量子比特的情况。从比较简单的角度考虑，因为针对单个量子程序的优化通常从减少单个量子程序执行时间以及减小错误率的角度出发，具有比较高的重要性，所以我们可以优先保证单个量子程序的优化，即按照以下顺序开展总体的优化工作：

运行针对单个量子程序的优化算法 -> 找到单个量子程序的一个（或多个最优映射方案）
运行Basic Algorithm（特征提取算法） -> 提取上述最优映射条件下量子程序占用量子比特的特征
基于提取的特征，运行Partly-connected Algorithm -> 得到最终的最优映射方案

Fully-connected Algorithm:

底层量子比特全联通假设下的算法开发：

此时无需考虑具体的量子比特映射方法，而只需要计算量子程序的执行顺序

输入：一批量子程序 $\{P_1, P_2, P_3, P_4, \dots\}$ ，量子机器的抽象特征 (T, N)
 P_i := 量子程序的OpenQasm
 N := 全部量子比特数
 T := $\{t_1, t_2, t_3, \dots\}$ ，各种量子门操作、reset操作、measure操作的耗时
输出： $\{O_1, O_2, O_3, O_4, \dots\}$
 O_i := (t_{si}, te_i)
 t_{si} := P_i 开始执行的时机
 te_i := P_i 结束执行的时机

Partly-connected Algorithm:

针对部分联通的量子计算机，计算的内容除了每个量子程序的执行顺序和开始执行的时机，还涉及如何调整每个量子程序的mapping方案

输入：一批量子程序 $\{P_1, P_2, P_3, P_4, \dots\}$, 量子机器的抽象特征 (T, M, N)

P_i := 量子程序的OpenQasm

N := 全部量子比特数

T := $\{t_1, t_2, t_3, \dots\}$, 各种量子门操作、reset操作、measure操作的耗时

M := 量子比特的联通性信息，可以用邻接矩阵表示

输出： $\{O_1, O_2, O_3, O_4, \dots\}$

O_i := (f_1, f_2, f_3, \dots)

f_j := 时间 t 到底层量子比特 q 的映射函数 ($f_j := t \rightarrow q$)，表示第 i 个程序要用到的第 j 个量子比特在 t 时刻被映射到底层的量子比特 q

测试集与验证

我们难以预见将来量子计算机的工作负载，所以测试集（一系列量子程序的集合，假想它们被同时提交到一个量子计算机上）将考虑多种可能情况。我们可以猜想测试集的一个重要参数是：每个量子程序占用的平均量子比特数 M 与量子计算机的总量子比特数 N 之比，也可以考虑量子程序占用的量子比特数的方差

暂时没有找到实际实现了动态reset和动态测量的量子机器，IBM声称会在近期给出符合条件的机器：

Reset provides an initial foray into real-time computation on IBM Quantum systems, and will be followed by a release of mid-circuit measurements in the near future.

我们可能必须通过自己建立的模型估测优化效果。

More

论证我们提出的优化手段在未来量子机器上的应用前景；

讨论reset操作的可信度，如何在兼顾错误率的情况下进行优化；

跨程序的执行优化是一个比较新的方向，我们将考虑这一方向的其他优化方案与发展前景

分工

暂定：

调研与选题：叶之帆

物理背景调研：孙晨寅

量子算法调研：全员

建模：全员

算法开发与测试：叶之帆，吴浩洋

讨论与评价：全员