

[MICRO2023] QuCT: A Framework for Analyzing Quantum Circuit by Extracting Contextual and Topological Features

邓皓亭

2024 年 1 月 9 日

1 Introduction and background

本文指出，在 NISQ 时代，现有的分析模型只能在精准度限制和高计算复杂度之中进行权衡。因此推出 quct 框架用于提取、分析和优化量子线路。quct 的主要创新点是将每个量子门向量化，定量地描述相邻量子门之间相互作用的程度。从这个向量化模型中还可以拓展出两个模型，分别可以用作 fidelity prediction 和 unitary decomposition。

Fidelity prediction: 评估噪声环境下获取量子线路的正确结果的可能性。以超导量子计算机为例，错误可以分为两类：由退相干和错误部署导致的错误、在将 qubit 信息读取到经典设备中造成的测量错误。目前流行的 fidelity 模型有 cross-entropy benchmarking (XEB) 和 randomized benchmarking (RB)，运行时间较短（多项式时间），但准确度较低；另一种方法是 simulate the Master's equation with density matrix，可以得到精确结果但拥有较高的计算复杂度（在 A100 GPU 上最多支持 34 量子比特）。

unitary decomposition: 将一个酉矩阵分解为一系列的可执行的量子门。Column-by-Column Decomposition (CCD) 和 Quantum Shannon Decomposition (QSD) 等数学方式可以快速分解酉矩阵，但分解结果往往含有较多的量子门；基于搜索方式的 QFAST 拥有较好的分解结果，但分解时间往往较长。

上述方式的不足基本都源于缺少提取和保存量子线路特征的分析方式。XEB 和 RB 在进行 fidelity prediction 时准确性较低，主要因为不能对量子

门交互产生的错误进行建模；CCD 和 QSD 在进行 unitary decomposition 时速度慢，主要因为没有考虑到量子线圈结构的特征。因此推出了 quct，其结构如图 1 所示。在上流模型中，每个量子门被转化为包含临近线圈特征的向量；在下流模型中，将向量作为输入，可以进行多种分析任务（如 fidelity prediction 和 unitary decomposition）。

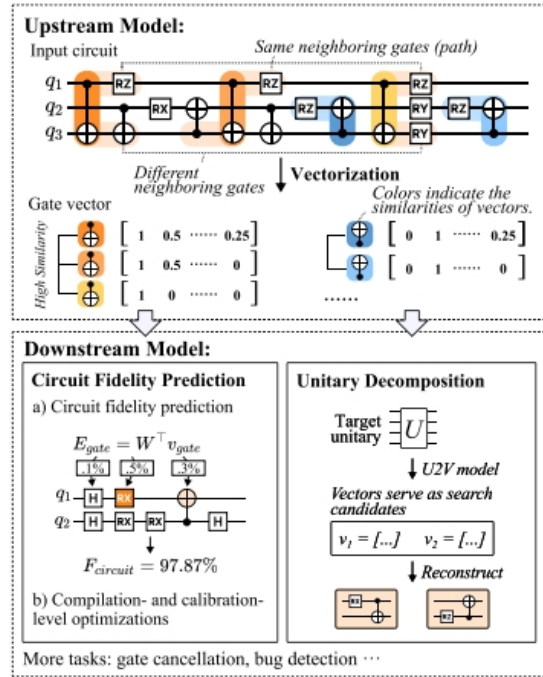


图 1: Overview of QuCT

2 Upstream model: gate vectorization

path: 描述一个量子门与其相邻量子门之间关系，如量子门类型（例如 CX、RZ 等）与执行顺序（例如并行、依赖等）。

使用向量化模型的优势：将无结构的量子线圈转化为一维向量的集合，可以在保留文字特征（如量子门种类、作用的 qubit）和拓扑特征的同时有效减少计算复杂度。

在向量化模型中，A path is a chained relation between multiple gates.

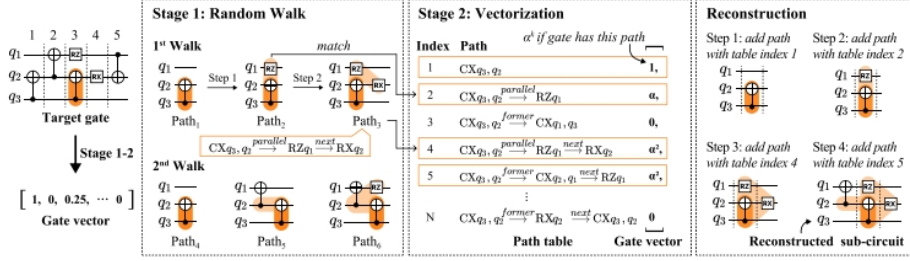


图 2: Generate paths sample

A k -step path is formulated as follows

$$path = g_1 \xrightarrow{r_1} g_2 \xrightarrow{r_2} \dots \xrightarrow{r_k} g_{k+1} \quad (1)$$

其中, r_i 表示两个量子门之间的关系 (一共三种: former、next、parallel)。可以看出, 文字特征纪录在参数 g 中, 拓扑特征记录在 $path$ 的关系中。

生成 $path$ 使用 random walk 算法, $path$ 的总数由每次 walk 的 step 数 (N_{step}) 和 walk 的数量 (N_{walk}) 决定, 这些都是需配置的参数。其中, N_{walk} 决定作为量子线路特征采样的相邻子线路的数量, N_{step} 决定在那些子线路中量子门的最大数量。因此, 一个量子门的 $path$ 数上限为 $N_{step}N_{walk} + 1$, 其中 1 代表 0-step。

在生成 $path$ 之后, 通过将 $path$ 与静态的 $path$ table (通过穷举得到) 进行比对来完成量子门的向量化。

图 2 展示了生成 $path$ 的过程。

对于所有 $path$, 出现的所有门都应该与 $path$ 中起始门物理相连。因此 $path$ 表的大小主要由设备拓扑图的复杂性决定, 而不是 qubit 的数量决定。

3 Downstream model: Fidelity prediction and optimization

对于 Fidelity Prediction, quct 采用的 fidelity 公式为:

$$E(v_i) = W^T v_i$$

$$F_{circuit} = \prod_{g_i \in G} (1 - E(v_i)) \prod_{q \in Q} MF_q$$

其中 W 由 stochastic gradient descent algorithm 根据 fidelity 数据集训练得出（数据集是硬件依赖的）。

训练的数据集由在量子设备上执行的一系列随机量子线路得到的 fidelity 数据组成。其中 fidelity 数据由 Hellinger fidelity function 得到：

$$F_{circuit}^{groundtruth} = 1 - \frac{1}{\sqrt{2}} \left\| \sqrt{P_{measured}} - \sqrt{P_{ideal}} \right\|$$

在编译层面，经典的流程可分为 routing 和 scheduling 两个部分。对于 routing，通过精确预测 fidelity，quct 可以与现有的编译器集成，用于以最佳 fidelity 寻找 routing 的解决方案（指插入 SWAP 门使物理的量子线路适应程序拓扑）。对于 scheduling，quct 也可拓展调度的方案（指调节各层的量子门执行顺序）使其能获得最佳的 fidelity。

4 Downstream model: unitary decomposition

在 unitary decomposition 方面，quct 与 QFAST 结合，使用 A^* 递归算法进行酉分解。具体过程如图 3 所示。

与 QFAST 的穷举搜索不同，QUCT 将量子门向量作为搜索对象，大大减少了搜索空间。量子门向量发挥了在酉门和量子线圈中中间表示层的作用。在 U2V 模型中，以 $\langle unitary, \{vectors\} \rangle$ 的形式构建数据。

对于 Unitary-to-Vector 模型，维护了一个经过 Qiskit transpiler 处理过的 U2V 数据集，并根据该数据集训练了一个 k 决策树的随机森林模型。对于过大的数据量，做出了以下优化来减小搜索空间、加速分解过程：a) 在生成 unitary-vector 对时，只选择源于第一层的门中的 vector，这是因为分解是一个递归插入的过程，每次都是从量子线路的起始处开始进行；b) 使用特征分解的方式将原来的酉矩阵投影到低维度。

在 QFAST 中，搜索空间包含酉矩阵的全部 qubits 组合，大小为 $\sum_{i=2}^{N_q-1} C_{N_q}^i$ ；在 QuCT 中，搜索空间等于模型中 trees(k) 的数量，经验上 $k = 2N_q$ 就足够得到合适的结果。

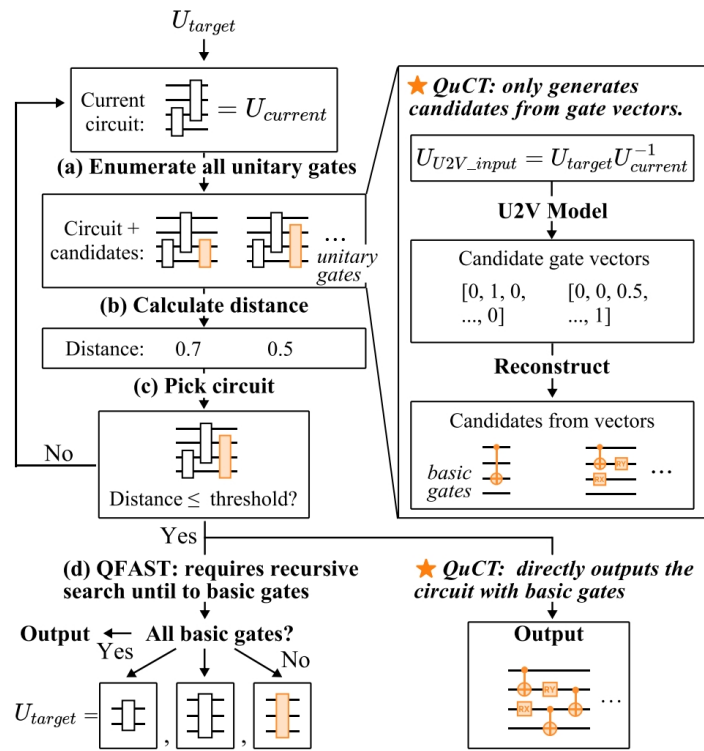


图 3: Workflow of the Unitary decomposition