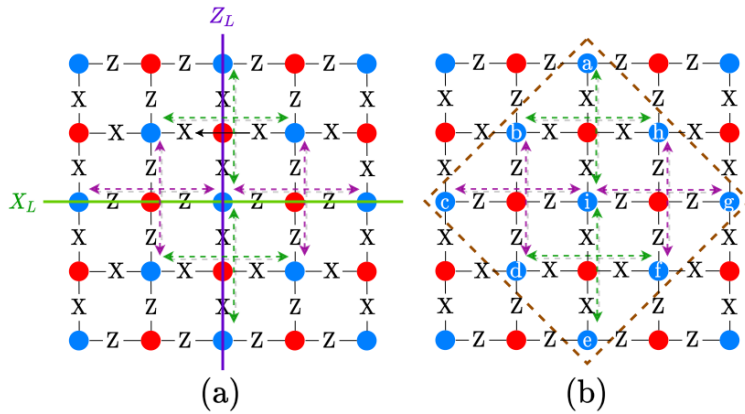
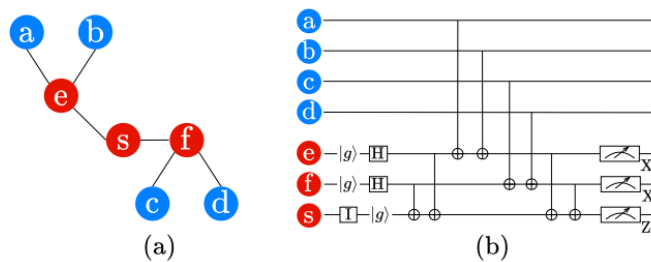


背景

Surface code 是一种将多个在平面上分布的物理量子比特编码为逻辑比特的容错编码方法。当前处于领导地位的超导量子计算机采用二维平面图的机制布局量子比特, 亦是为了便于应用 Surface code。Surface code 将物理量子比特划分为数据比特和测量比特 (又称为 Syndrome, 一个 Syndrome 测量对应一个 Stabilizer)。测量有两种, 分别是 X 测量和 Z 测量, 前者可以检测 Z 错误而后者可以检测 X 错误。Surface code 的纠错能力和它的码距相关。码距定义为执行一次逻辑 X/Z 门至少需要操作多少个物理量子比特。如下图所示, 这两种编码的码距都是 3。由于(b)棕色框所示的编码需要更少的物理量子比特, 因此它也被称为**紧凑的 surface code**, 直观上可以理解在普通 surface code 基础上删除不太重要的边角。



当前超导量子计算机的量子比特连通度小, IBM 的技术路线显示他们的设备采用六边形连通图, 即一个比特连接三个其他比特, 这使得传统的 surface code 无法直接适配这种连通图。之前的工作尝试将物理量子比特的连接设置为可调的, 这种方法的代价很大; 设计适应特定设备的纠错码, 这种方法不是自动的, 不够普适。于是基于 flag-bridge 线路的 surface code 应运而生, 简单来说, 通过引入辅助比特来连接测量比特和数据比特, 如下图所示: a,b,c,d 是数据比特, s 是测量比特, e,f 是辅助比特, 由于引入的辅助比特亦有可能出错, 因此增加了两个 X 测量来检测这种错误。



但目前缺乏将 bridge 线路技术应用于大规模 surface code 编码的方法，本文提出了在超导量子比特连通图上的 surface code 线路合成方法。

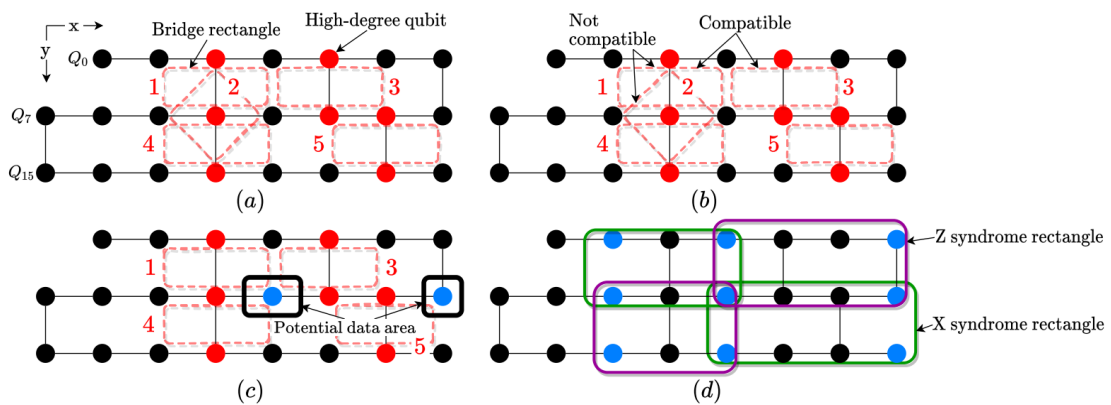
贡献点

1. 首次系统地形式化超导量子设备上的 surface code 合成问题，并提出了该问题的三大挑战：数据量子比特分配；测量线路合成；测量调度
2. 提出了第一个 surface code 自动线路合成框架 Surf-Stitch
3. 实验证明 Surf-Stitch 比手工设计的纠错码有更高的容错门槛

主要思路

数据比特分配

观察到 1、构造 bridge 线路的本质是构造一棵根节点为测量比特，叶节点为数据比特的树。
2、每棵树都拥有 4 个叶节点，因此它拥有两个度为 3 的节点或者一个度为 4 的节点。本文的数据比特分配方法的核心思想是：将比特连通图上的一个度为 4 的节点或者两个相邻的度为 3 的节点视为一棵树的枝干，尽量使得不相邻的树的枝干的数量更多，选定树的枝干之后再将其叶节点填补进去。



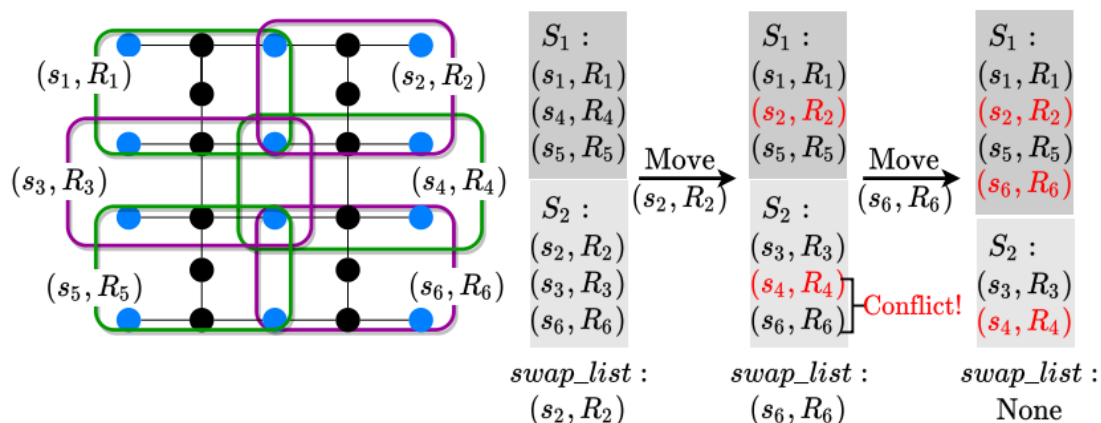
Bridge 树合成

测量线路和 bridge 树等价，因此将测量线路合成问题归约到 bridge 树合成问题。合成的目

标是最小化树的节点数，这有利于减少辅助比特的用量。关键思想是：1、找到四个数据比特间最短的和次最短的两两路径，将这两条路径上的节点加入树。2、如果这两条路径不连通，找到一条连接这两条路径的最短路径，将该路径的节点加入树。

测量调度

观察到不同的测量线路执行时长不同，用到相同辅助比特的测量线路不能并行。关键思想：尽量使得执行时长接近的线路在同一层，因为如果一层中两个线路的执行时长不同，其中较短的线路执行完毕后需要等待另一个。有可能本文没有考虑更细粒度的调度，而是规定层间存在屏障，因此有改进空间。



优缺点

1. 背景和动机写得细致入微，面面俱到，层层递进，将这篇文章的价值立于不败之地！
2. 解决方法描述较为模糊，但也能让人看到核心思想，让读者似懂非懂，高！实在是高！
3. 实验基于数值模拟，可见目前顶会也能容忍没有真机实验的高质量论文了。

对我的启发

- 1、编译优化的研究趋势是和实际应用结合，前有 Paulihedral，现有 Surf-Stitch
- 2、根据本文的引用，找到了一个噪声模型，之后的数值模拟试验可以采用该模型
- 3、背景(特别是 related works)和动机不要吝啬字数，为读者娓娓道来，将文章价值拔高，方法反而不用写的太细，但要突出核心思想，用例子，画图!!!