

# 魔法态与 Code-Switching: 从 Bravyi–Kitaev 蒸馏到高保真逻辑魔法态

高健硕

北京大学物理学院

量子计算组会报告

# 报告结构

- 1 引言：为什么魔法态如此重要？
- 2 Clifford 计算为何不通用？
- 3 Bloch 球、稳定子多面体与魔法态
- 4 Magic States：定义与几何直观
- 5 由 Magic States 实现通用量子计算
- 6 魔法态蒸馏：5-to-1 协议的数学推导
- 7 为什么用 Code-Switching 来培育魔法态？
- 8 魔法态总结
- 9 控制系统

# 容错量子计算的两类门

- 在大多数容错架构中，门被大致分成两类：
  - Clifford 门：例如  $H, S, CNOT$ ，在许多编码上可以横向实现，容错成本低；
  - 非 Clifford 门：常选  $T$  门  $T = \text{diag}(1, e^{i\pi/4})$ ，容错实现开销巨大。
- 若只允许 Clifford 门 +  $|0\rangle$  准备 + Pauli 测量：
  - 演化保持在所谓的稳定子态 (stabilizer states) 集合中；
  - 可被经典计算机高效模拟 (Gottesman–Knill 定理)。
- 为了通用量子计算，必须引入额外的“非 Clifford 资源”。

# Bravyi–Kitaev 模型：魔法态的角色

- Bravyi–Kitaev (2004) 的思路：

## 资源理论视角

- 允许的“免费操作”：所有 Clifford 电路 + Pauli 测量 + 准备  $|0\rangle$ ；
- 受限资源：一批有噪声的单比特纯化资源态  $\rho$ 。
- 若  $\rho$  的 Bloch 向量沿某个魔法方向的极化足够大，可以通过魔法态蒸馏协议：

$$\rho^{\otimes n} \xrightarrow{\text{Clifford+ 测量 +postselect}} \rho_{\text{out}} \approx |\psi_{\text{magic}}\rangle\langle\psi_{\text{magic}}|,$$

将多份低保真资源，转化为少量高保真魔法态。

- 一旦有了高保真魔法态 + Clifford 门，就能实现通用容错门集。

# Pauli 群与 Clifford 群

## 定义 (Pauli 群与 Clifford 群)

- $n$  比特 Pauli 群:

$$\mathcal{P}_n = \{\pm 1, \pm i\} \cdot \{I, X, Y, Z\}^{\otimes n}.$$

- Clifford 群  $\mathcal{C}_n$  定义为所有满足

$$U\mathcal{P}_n U^\dagger = \mathcal{P}_n$$

的  $n$  比特酉算符集合。

- 生成集: 在 1 比特上,

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$$

生成  $\mathcal{C}_1$ ; 再加 CNOT 生成  $\mathcal{C}_n$ 。

- 性质: Clifford 门按共轭作用保留 Pauli 结构:

$$UXU^\dagger \in \{\pm X, \pm Y, \pm Z\}, \text{ 等。}$$

## 证明思路 (一): 单比特 Clifford 群是有限的

- 将单比特态看作 Bloch 球上的向量:

$$\rho = \frac{1}{2} (\mathbb{I} + \mathbf{r} \cdot \boldsymbol{\sigma}), \quad \mathbf{r} \in \mathbb{R}^3, \quad |\mathbf{r}| \leq 1.$$

- 任何单比特酉算符  $U$  对应 Bloch 球上的一个转动  $R_U \in SO(3)$ :

$$U(\mathbf{r} \cdot \boldsymbol{\sigma}) U^\dagger = (R_U \mathbf{r}) \cdot \boldsymbol{\sigma}.$$

- 若  $U$  是 Clifford, 则

$$UXU^\dagger, \quad UYU^\dagger, \quad UZU^\dagger$$

仍是 Pauli 矩阵 (带符号), 即 Bloch 球上的坐标轴被 **permute** 并翻转。

- 因此  $R_U$  必须是保持立方体 (或八面体) 对称的转动群元素,

$$\mathcal{C}_1 \cong \text{立方体旋转群}, \quad |\mathcal{C}_1| = 24.$$

- 结论: **只用 Clifford 生成的单比特门集合是一个有限群, 无法逼近所有单比特酉算符。**

## 证明思路 (二): Gottesman–Knill 定理

### 定义 (稳定子态)

一个  $n$  比特纯态  $|\psi\rangle$  若存在一个 Abel 群  $S \subset \mathcal{P}_n$  满足:

$$\forall P \in S, \quad P|\psi\rangle = |\psi\rangle,$$

且  $S$  最大, 称  $|\psi\rangle$  为 **稳定子态**。

### 定理 (Gottesman–Knill (简述))

由以下操作生成的任何量子过程可在经典多项式时间内模拟:

- ① 准备计算基态 (如  $|0\rangle^{\otimes n}$ );
- ② 施加任意 *Clifford* 电路  $U \in \mathcal{C}_n$ ;
- ③ 测量任意 *Pauli* 算符, 并按结果做 *Clifford* 纠正。

# Gottesman–Knill 的经典模拟算法（略证）

- 想法：不用显式存  $|\psi\rangle$ ，只存它的稳定子群  $S$ 。
- 表示： $S$  可由  $n$  个独立生成元  $\{g_1, \dots, g_n\} \subset \mathcal{P}_n$  描述，每个是  $n$  比特 Pauli，附带一个符号 ( $\pm 1$ )。
- Clifford 门  $U$  的作用：

$$Ug_jU^\dagger \in \mathcal{P}_n,$$

仍是 Pauli 运算，且可用线性代数模 2 运算快速更新生成元。

- 测量某 Pauli  $M$ ：
  - 若  $M$  与所有  $g_j$  对易，则测量结果已经由  $S$  决定；
  - 否则，适当替换生成元集合，可在  $O(n^2)$  时间更新  $S$ ，并生成经典比特结果。
- 整个演化过程都在一个大小为  $O(n^2)$  的经典表上进行更新，因此可多项式时间模拟。

## 结论

如果只允许上述 Clifford+ 测量流程，那么其计算能力  $\subseteq \text{P}$ （至多稍强），与通用量子计算的 BQP 相差巨大。

# 因此仅有 Clifford 门不通用

- 综合两个观点：
  - ① 单比特 Clifford 群是有限群，无法逼近任意单比特旋转；
  - ② 带任意测量和经典控制的 Clifford 电路仍可被经典多项式时间模拟。
- 若只使用 Clifford 门 + Pauli 测量：
  - 所有可达态都是稳定子态的 convex 组合；
  - 所有过程都可用稳定子形式模拟。
- 要获得**真正的量子优势**、实现通用量子计算，需要引入**非 Clifford 资源**：
  - 方式一：直接实现容错 T 门（通常非常昂贵）；
  - 方式二：引入**魔法态**并通过魔法态注入来实现 T 门。

# Bloch 球与极化向量

- 任意单比特密度矩阵都可写成：

$$\rho = \frac{1}{2}(\mathbb{I} + r_x\sigma_x + r_y\sigma_y + r_z\sigma_z),$$

其中  $\mathbf{r} = (r_x, r_y, r_z) \in \mathbb{R}^3$ ,  $|\mathbf{r}| \leq 1$ 。

- $\mathbf{r}$  称为**极化向量**, 对应 Bloch 球上的点。
- Pauli 算符的本征态：

$$|0\rangle, |1\rangle \ (\mathcal{Z}), \quad |\pm_x\rangle = \frac{|0\rangle \pm |1\rangle}{\sqrt{2}} \ (\mathcal{X}), \quad |\pm_y\rangle = \frac{|0\rangle \pm i|1\rangle}{\sqrt{2}} \ (\mathcal{Y}),$$

对应 Bloch 球上的 6 个顶点：

$$(\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1).$$

# 稳定子八面体: Clifford 可达态的凸包

- Bravyi–Kitaev 证明: 若  $\rho$  仅通过 Clifford 操作和经典随机过程制备, 则其极化向量满足

$$|r_x| + |r_y| + |r_z| \leq 1.$$

- 这是一个以原点为中心的正八面体  $O$ :

$$O = \text{conv}\{(\pm 1, 0, 0), (0, \pm 1, 0), (0, 0, \pm 1)\}.$$

- 解释:

- 八面体的 6 个顶点就是 Pauli 本征态;
- 若  $r$  在八面体内部, 则  $\rho$  是它们的凸组合

$$\rho = \sum_k p_k |\phi_k\rangle\langle\phi_k|,$$

即稳定子态的混合。

- 几何意义: 八面体  $O$  中的每个点都是 6 个 Pauli 本征态的概率混合。
- 反之, 若  $r$  落在八面体外, 则  $\rho$  不可能是稳定子混合, 必然含有非 Clifford 资源。

# Gottesman–Knill 的几何版表述

## Theorem (几何版 Gottesman–Knill)

若辅助态  $\rho$  的极化向量  $(\rho_x, \rho_y, \rho_z)$  满足

$$|\rho_x| + |\rho_y| + |\rho_z| \leq 1,$$

则  $\rho$  可以用稳定子态的经典随机混合制备；

使用操作基  $O_{\text{ideal}} \cup \{\rho\}$  的任何自适应量子计算，都可以在经典概率图灵机上高效模拟。

- $O_{\text{ideal}}$ : 制备  $|0\rangle$ 、任意 Clifford 门、任意单比特 Pauli 测量。
- 几何直观: 八面体  $\mathcal{O}$  对应“稳定子锥”，处于其中的噪声资源不会带来真正的量子优势。

# Magic states 的 Pauli 展开

- 定义两类特殊单比特纯态（在 Bloch 球上落在八面体对称轴方向）：

$$|T\rangle\langle T| = \frac{1}{2} \left( I + \frac{1}{\sqrt{3}} (\sigma_x + \sigma_y + \sigma_z) \right),$$

$$|H\rangle\langle H| = \frac{1}{2} \left( I + \frac{1}{\sqrt{2}} (\sigma_x + \sigma_z) \right).$$

- 极化向量：

$$\vec{r}_T = \frac{1}{\sqrt{3}} (1, 1, 1), \quad \vec{r}_H = \frac{1}{\sqrt{2}} (1, 0, 1).$$

- 它们都落在八面体  $\mathcal{O}$  的对称轴上：沿着“远离稳定子混合锥”的方向。

## Magic states 的两种类型

- **T-type magic states**: 所有  $\{U|T\rangle : U \in C_1\}$ 。
- **H-type magic states**: 所有  $\{U|H\rangle : U \in C_1\}$ 。
- 其中  $C_1$  是单比特 Clifford 群。
- 在 Bloch 球上:
  - T-type: 对应八面体绕顶点的  $120^\circ$  旋转对称轴 (共 8 个态, 忽略全局相位)。
  - H-type: 对应八面体绕棱中点的  $180^\circ$  旋转对称轴 (共 12 个态)。
- 这些态都不能写成稳定子态的混合, 是“非稳定子”资源。

# Magic states 的两重角色

- 角色 1: 给定 magic states 的完美制备能力 + Clifford 操作  $\Rightarrow$  可以实现通用量子计算。
- 角色 2: 仅利用 Clifford 操作, 可以从“带噪声的 magic states”中进行蒸馏, 得到更干净的 magic states。
- 因此, magic states 既是“非 Clifford 资源的承载者”, 又是“自身可被 Clifford 操作自救”的特殊方向。
- 本文后半部分就是围绕“magic state 蒸馏 + gate injection”展开。

# 一般的 gate injection 思路

- 定义一族单比特补助态：

$$|A_\theta\rangle = \frac{1}{\sqrt{2}}(|0\rangle + e^{i\theta}|1\rangle), \quad \theta \not\equiv k\frac{\pi}{2}.$$

- 目标：仅利用  $O_{\text{ideal}}$  和若干个  $|A_\theta\rangle$ ，实现相位门

$$\Lambda(e^{i\theta}) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{pmatrix}.$$

- 做法（单比特 gate injection）：

- ① 待操作比特处于未知态  $|\psi\rangle = a|0\rangle + b|1\rangle$ 。
- ② 附加一辅助比特在  $|A_\theta\rangle$ ，整体态  $|\Psi_0\rangle = |\psi\rangle \otimes |A_\theta\rangle$ 。
- ③ 测量稳定子  $S_1 = \sigma_z \otimes \sigma_z$ ，并自适应地做 CNOT 等 Clifford 操作。

# 随机游走式实现 $\Lambda(e^{i\theta})$

- 一次“尝试”的结果：
  - 以概率  $1/2$  得到  $\Lambda(e^{i\theta})$ ；
  - 以概率  $1/2$  得到  $\Lambda(e^{-i\theta})$ ；
  - 测量结果告诉我们是  $+\theta$  还是  $-\theta$ 。
- 多次重复：相当于在指数  $p_k \in \mathbb{Z}$  上做随机游走，序列的作用为

$$\Lambda(e^{ip_1\theta}), \Lambda(e^{ip_2\theta}), \dots$$

- 经典随机游走几乎必然访问到  $p_k = 1$ ，因此最终能够实现期望的  $\Lambda(e^{i\theta})$ 。
- 若  $\theta$  为  $2\pi$  的有理倍数，则是在有限群  $\mathbb{Z}_q$  上的随机游走，收敛更快。

# H-type magic state 与 $\Lambda(e^{-i\pi/4})$

- 把  $|H\rangle$  写成计算基:

$$|H\rangle = \cos \frac{\pi}{8} |0\rangle + \sin \frac{\pi}{8} |1\rangle.$$

- 存在 Clifford 门  $H, K$  使得

$$HK|H\rangle = e^{i\pi/8} |A_{-\pi/4}\rangle.$$

- 结论:

- 一旦能理想制备  $|H\rangle$ , 只用 Clifford 操作即可得到  $|A_{-\pi/4}\rangle$ ;
- 进一步通过 gate injection, 实现非 Clifford 门  $\Lambda(e^{-i\pi/4})$ 。

- 已知: Clifford 群 +  $\Lambda(e^{-i\pi/4})$  + CNOT 构成通用门集。

# T-type magic state 与 $\Lambda(e^{-i\pi/6})$

- $|T\rangle$  的计算基展开：

$$|T\rangle = \cos \beta |0\rangle + e^{i\pi/4} \sin \beta |1\rangle, \quad \cos(2\beta) = \frac{1}{\sqrt{3}}.$$

- 使用两份  $|T\rangle$ ：

- ① 先制备  $|\Psi_0\rangle = |T\rangle \otimes |T\rangle$ ;
- ② 测量  $S_1 = \sigma_z \otimes \sigma_z$ , 仅在测得 +1 时继续 (期望用掉约 3 份  $|T\rangle$ );
- ③ 通过 CNOT + Hadamard 等 Clifford 门, 把结果态变换为

$$|A_{-\pi/6}\rangle.$$

- 再通过 gate injection, 实现  $\Lambda(e^{-i\pi/6})$ , 而

$$\text{Clifford} + \Lambda(e^{-i\pi/6})$$

也构成通用门集。

# 小结：为什么「magic + Clifford」是通用的？

- 对任一 magic state (H- 或 T-type):
  - 通过有限个 Clifford 门把它变成某个  $|A_\theta\rangle$ ;
  - 用 gate injection 把  $|A_\theta\rangle$  转换成相位门  $\Lambda(e^{i\theta})$ 。
- 选取合适的  $\theta$  (例如  $-\pi/4$  或  $-\pi/6$ )，再配合 Clifford 群：

$\langle C_1, \text{CNOT}, \Lambda(e^{i\theta}) \rangle$  在  $U(2^n)$  中致密.

- 因此：

$O_{\text{ideal}} + \{\text{理想 magic states}\} \Rightarrow \text{通用量子计算 (UQC)} .$

# 魔法态的定义与 Bloch 球表示

定义 (Bravyi–Kitaev 的 T/H 型魔法态)

定义单比特纯态，使得

$$|T\rangle = \frac{1}{2} \left( \mathbb{I} + \frac{1}{\sqrt{3}} (\sigma_x + \sigma_y + \sigma_z) \right),$$

$$|H\rangle = \frac{1}{2} \left( \mathbb{I} + \frac{1}{\sqrt{2}} (\sigma_x + \sigma_z) \right).$$

- 因此  $|T\rangle$ 、 $|H\rangle$  的 Bloch 向量分别为

$$\mathbf{r}_T = \frac{1}{\sqrt{3}} (1, 1, 1), \quad \mathbf{r}_H = \frac{1}{\sqrt{2}} (1, 0, 1).$$

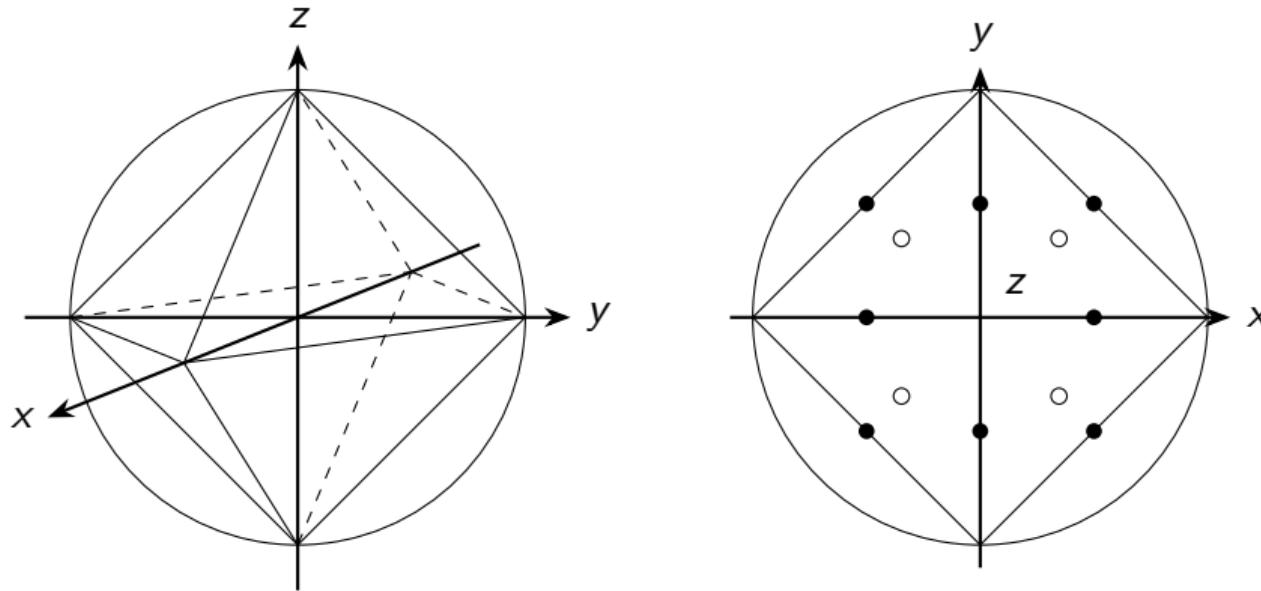
- 它们显然满足

$$|r_x| + |r_y| + |r_z| > 1,$$

即位于稳定子八面体之外。

- 对，施加任意单比特 Clifford  $U \in \mathcal{C}_1$ ，得到的轨道上的态称为 T 型 / H 型魔法态。

# Bloch 球与稳定子八面体 $O$



左: 带稳定子八面体  $O$  的 Bloch 球示意; 右:  $O$  在  $x-y$  平面的投影, 空心圈为 T 型魔法态, 实心点为 H 型魔法态。

# 噪声模型与“魔法方向”极化

- 以 T 型魔法态为例，设目标态为  $|T\rangle$ ，对应 Bloch 向量  $\mathbf{n}$ ：

$$\mathbf{n} = \mathbf{r}_T = \frac{1}{\sqrt{3}}(1, 1, 1), \quad |T_0\rangle = |T\rangle, \quad |T_1\rangle = \text{与 } |T\rangle \text{ 正交的态.}$$

- 物理上我们准备到的往往是退相干后的混合态：

$$\rho = (1 - \epsilon) |T_0\rangle\langle T_0| + \epsilon |T_1\rangle\langle T_1|,$$

其中  $\epsilon$  是沿魔法方向的错误率。

- 在 Bloch 球上，这对应

$$\rho = \frac{1}{2} (\mathbb{I} + (1 - 2\epsilon) \mathbf{n} \cdot \boldsymbol{\sigma}),$$

即极化大小  $|\mathbf{r}| = |1 - 2\epsilon|$  沿方向  $\mathbf{n}$ 。

- 蒸馏的目标：构造一个 Clifford+ 测量协议，将

$$\rho^{\otimes 5} \longrightarrow \rho_{\text{out}}$$

使得  $\epsilon_{\text{out}} < \epsilon$ ，且反复迭代后  $\epsilon_k \rightarrow 0$ 。

## 第 0 步：去相干 (twirling) 把噪声对角化

- 为了简化分析，先对每个物理比特施加去相干变换：

$$\mathcal{D}(\eta) = \frac{1}{3}(\eta + T\eta T^\dagger + T^\dagger\eta T),$$

其中  $T$  是单比特 Clifford 运算。

- 性质：

$$\mathcal{D}(|T_0\rangle\langle T_1|) = 0, \quad \mathcal{D}(|T_1\rangle\langle T_0|) = 0,$$

即在  $\{|T_0\rangle, |T_1\rangle\}$  本征基下把  $\rho$  对角化。

- 因此去相干后可以假设

$$\rho = (1 - \epsilon)|T_0\rangle\langle T_0| + \epsilon|T_1\rangle\langle T_1|$$

而不丢失一般性。

- 这一步只用随机选  $\{I, T, T^\dagger\}$  并不依赖精确噪声模型。

# 第1步：五份资源态的张量积表示

- 取五份独立同分布的资源态：

$$\rho_{\text{in}} = \rho^{\otimes 5}.$$

- 在  $\{|T_0\rangle, |T_1\rangle\}$  基底下，可以写成

$$\rho_{\text{in}} = \sum_{x \in \{0,1\}^5} \epsilon^{|x|} (1 - \epsilon)^{5 - |x|} |T_x\rangle\langle T_x|,$$

其中

$$x = (x_1, \dots, x_5), \quad |x| = \sum_j x_j, \quad |T_x\rangle = \bigotimes_{j=1}^5 |T_{x_j}\rangle.$$

- 直观理解：

- $|x|$  表示 5 份资源中有多少份是“错误态”  $|T_1\rangle$ ；
- 系数  $\epsilon^{|x|} (1 - \epsilon)^{5 - |x|}$  就是对应的概率。

## 第 2 步: 5 量子比特码与投影算符

- 选用著名的 5 量子比特稳定子码, 其 4 个生成元可写作:

$$S_1 = XZZXI,$$

$$S_2 = IXZZX,$$

$$S_3 = XIXZZ,$$

$$S_4 = ZXIXZ.$$

- 码空间  $L$  定义为

$$L = \{|\Psi\rangle \in \mathcal{H}_2^{\otimes 5} \mid S_j |\Psi\rangle = |\Psi\rangle, j = 1, \dots, 4\}.$$

- 投影到码空间的投影算符:

$$\Pi = \frac{1}{16} \prod_{j=1}^4 (\mathbb{I} + S_j).$$

- 该码具有如下关键性质:

- $(X^{\otimes 5}, Y^{\otimes 5}, Z^{\otimes 5})$  在  $L$  上作为逻辑 Pauli 作用;
- $T^{\otimes 5}$  在  $L$  上作为逻辑 T 门作用 (差一个整体相位)。

### 第3步：测量稳定子并选取“零综合”

- 在五比特态  $\rho_{\text{in}}$  上测量  $S_1, \dots, S_4$ , 只保留测得平凡校验子的轮次。
- 相当于应用投影：

$$\rho_s = \Pi \rho_{\text{in}} \Pi.$$

- 直接代入展开式

$$\rho_{\text{in}} = \sum_x \epsilon^{|x|} (1 - \epsilon)^{5-|x|} |T_x\rangle\langle T_x|,$$

得到

$$\rho_s = \sum_x \epsilon^{|x|} (1 - \epsilon)^{5-|x|} \Pi |T_x\rangle\langle T_x| \Pi.$$

- 经过对称性分析（详见原文/附录），可以证明：

$$\Pi |T_x\rangle = 0, \quad \text{若} |x| = 1 \text{ 或} 4,$$

而  $|x| = 0, 2, 3, 5$  时有非零贡献，并可归并到两个逻辑本征态上。

## 第4步：输出逻辑态的显式系数

- 码空间  $L$  内选取两个归一化的逻辑态：

$$\left| T_0^L \right\rangle, \left| T_1^L \right\rangle,$$

分别对应逻辑“好”态与“坏”态。

- 投影后未归一化的输出态可以写成

$$\rho_s = a(\epsilon) \left| T_0^L \right\rangle \langle T_0^L | + b(\epsilon) \left| T_1^L \right\rangle \langle T_1^L |,$$

其中系数为

$$a(\epsilon) = \frac{1}{6} [\epsilon^5 + 5\epsilon^2(1-\epsilon)^3],$$

$$b(\epsilon) = \frac{1}{6} [(1-\epsilon)^5 + 5\epsilon^3(1-\epsilon)^2].$$

- 测得平凡校验子的**成功概率**为

$$p_s(\epsilon) = a(\epsilon) + b(\epsilon) = \frac{1}{6} [\epsilon^5 + 5\epsilon^2(1-\epsilon)^3 + 5\epsilon^3(1-\epsilon)^2 + (1-\epsilon)^5].$$

## 第 5 步：解码并得到新的单比特噪声参数

- 5 量子比特码有一个解码酉  $V$ :

$$VL = \mathcal{H}_2 \otimes |0000\rangle,$$

即将码空间  $L$  映射到 “1 个逻辑比特 + 4 个辅助比特”。

- 解码后仅保留第一个逻辑比特，其状态为单比特混合态

$$\rho_{\text{out}} = (1 - \epsilon_{\text{out}}) |T_0\rangle\langle T_0| + \epsilon_{\text{out}} |T_1\rangle\langle T_1|,$$

其中

$$\epsilon_{\text{out}} = \frac{a(\epsilon)}{a(\epsilon) + b(\epsilon)} = \frac{\epsilon^5 + 5\epsilon^2(1 - \epsilon)^3}{\epsilon^5 + 5\epsilon^2(1 - \epsilon)^3 + (1 - \epsilon)^5 + 5\epsilon^3(1 - \epsilon)^2}.$$

- 这是5-to-1 魔法态蒸馏的显式映射  $\epsilon \mapsto \epsilon_{\text{out}}$ 。

# 小噪声展开与阈值条件

- 对  $\epsilon_{\text{out}}(\epsilon)$  在  $\epsilon = 0$  处展开：

$$\epsilon_{\text{out}} = 5\epsilon^2 + 10\epsilon^3 + O(\epsilon^4).$$

- 物理含义：

- 对于小噪声，错误率从  $O(\epsilon)$  压到  $O(\epsilon^2)$ ；
- 每轮蒸馏显著提高魔法态的保真度。

- 阈值：解方程  $\epsilon_{\text{out}} = \epsilon$ ，可得物理区间内的非平凡解

$$\epsilon_{\text{th}} \approx 0.173.$$

- 若初始魔法态的错误率  $\epsilon < \epsilon_{\text{th}}$ ，则

$$\epsilon_{\text{out}} < \epsilon,$$

反复迭代  $\epsilon_k$  将单调下降并趋近 0。

- 对应 Bloch 极化：

$$|\mathbf{r}| = |1 - 2\epsilon|, \quad |\mathbf{r}|_{\text{th}} \approx 1 - 2 \times 0.173 \approx 0.654.$$

即极化超过某阈值，就可以通过蒸馏得到几乎纯净的魔法态。

- Bravyi–Kitaev 进一步证明：对于 Bloch 球上落在稳定子八面体  $O$  外的态，存在适当的稳定子码与蒸馏协议，使其可被蒸馏到高保真魔法态。
- 直观图像：
  - 八面体  $O$  内部：稳定子混合态，在 Clifford 模型下“免费”；
  - 八面体外部：具有魔法资源，可通过多份协同与 postselection 放大资源；
  - 多轮蒸馏：将“极化略高于阈值”的资源，逼近 Bloch 球表面上的“纯魔法态”。
- 这就是“如果  $\rho$  沿某些魔法方向的极化超过阈值，就可以通过魔法态蒸馏得到近乎纯净的魔法态”的具体数学实现。

# 传统蒸馏的瓶颈

- 理论上：多级魔法态蒸馏可以把误差压到  $10^{-10}$  以下。
- 实际上：
  - 每一层蒸馏都消耗大量物理比特和门操作；
  - 蒸馏电路本身很长，对门误差和读出误差极度敏感；
  - 在许多架构中，T 门的**空间-时间成本远高于 Clifford 门**。
- 因此一个新思路是：

“先把单个逻辑魔法态做得尽量好”

- 在一个拥有横向 T 门的编码中直接制备高保真逻辑魔法态；
- 再通过**码切换**把这个逻辑态转移到主计算码上；
- 让后续蒸馏轮数大幅减少，甚至部分场景免蒸馏。

# 两种代码的“分工合作”

- 设有两种容错编码：
  - ① 码 A (如 3D color code / 15 比特 qRM 码)：存在横向 T 门，适合制备魔法态；
  - ② 码 B (如 2D color code / 7 比特 Steane 码)：适合作为主计算码，但无横向 T。
- 策略：
  - ① 在码 A 中初始化  $|\bar{+}\rangle_A$ ，施加横向 T 门得到  $|\bar{T}\rangle_A$ ；
  - ② 在码 B 中初始化  $|\bar{0}\rangle_B$ ；
  - ③ 实现逻辑 CNOT  $\text{CNOT}_{A \rightarrow B}$ ，并在 A 上测量逻辑  $\bar{X}$ ；
  - ④ 根据测量结果对 B 做逻辑 Pauli 纠正；
  - ⑤ 结果：B 中得到逻辑魔法态  $|\bar{T}\rangle_B$ 。
- 这就是 Daguerre & Kim 提出的基于码切换的低开销魔法态制备的核心思路。

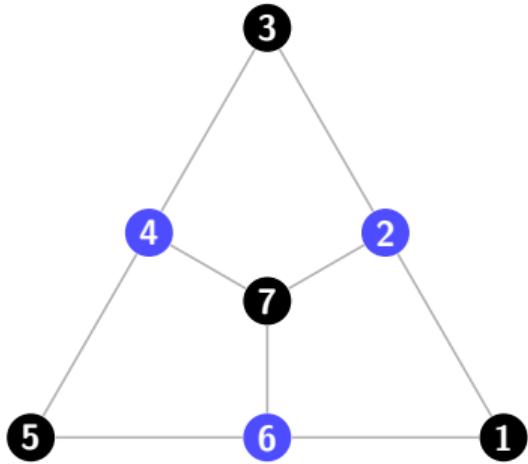
# 码切换方案的优势

## 与纯蒸馏方案相比

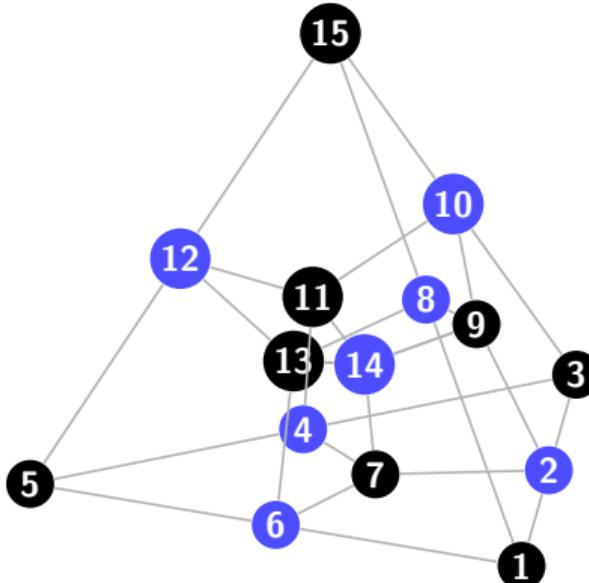
- 不必在主计算码上实现复杂的多层蒸馏电路；
- 利用“横向 T 门”所在的 3D 码做一次高质量准备，再切换到 2D 码；
- 只需几十个物理比特，就能把逻辑魔法态误差压到  $10^{-4}$ – $10^{-5}$  级别；
- 结合少量蒸馏轮数，可显著降低整体 T 门成本。

## 与实验条件的兼容

- Daguerre 等人用囚禁离子处理器实验演示，在 28 个物理比特上实现从 qRM 到 Steane 的码切换；
- 测得 Steane 码中逻辑魔法态失真  $\lesssim 5 \times 10^{-4}$ ，成功率约 80%；
- 显示真实器件噪声已经 **低于该协议的伪阈值**，逻辑层面的魔法态质量优于物理层。



Steane code  $[[7, 1, 3]]$



qRM code  $[[15, 1, 3]]$

# Steane 与 qRM 上的横向门集

- 两个代码在横向门上的能力不同，是码切换方案的出发点。

	Steane [[7, 1, 3]]	qRM [[15, 1, 3]]
几何维度	2D 三色码	3D 三色码 (Reed-Muller)
横向 Clifford	是: $H, S, CNOT$ 等	是
横向 T 门	否	是: $T^{\otimes 15}$ 实现逻辑 $\bar{T}$
适合作为	主计算码、容错 Clifford 网络	短暂“借用”来种魔法态

- 想法：在 qRM 上先做

$$|\bar{+}\rangle_{qRM} \xrightarrow{\bar{T}} |\bar{T}\rangle_{qRM},$$

再码切换到 Steane，使逻辑魔法态落在 2D 主计算码上。

# 码切换的本质：子空间嵌入与规范自由度

- 两个颜色码可视为嵌入在同一物理 Hilbert 空间上的**稳定子码**：

$$\mathcal{H} = (\mathbb{C}^2)^{\otimes 15},$$

$$\mathcal{C}_{\text{qRM}} = \{|\psi\rangle : S_i^{(15)} |\psi\rangle = |\psi\rangle\},$$

$$\mathcal{C}_{\text{Steane}} = \{|\phi\rangle : S_j^{(7)} |\phi\rangle = |\phi\rangle\} \otimes |\text{aux}\rangle.$$

- 经过适当的**编码等价转换**，可以让 Steane 码被视为 qRM 码的一个**子码**：

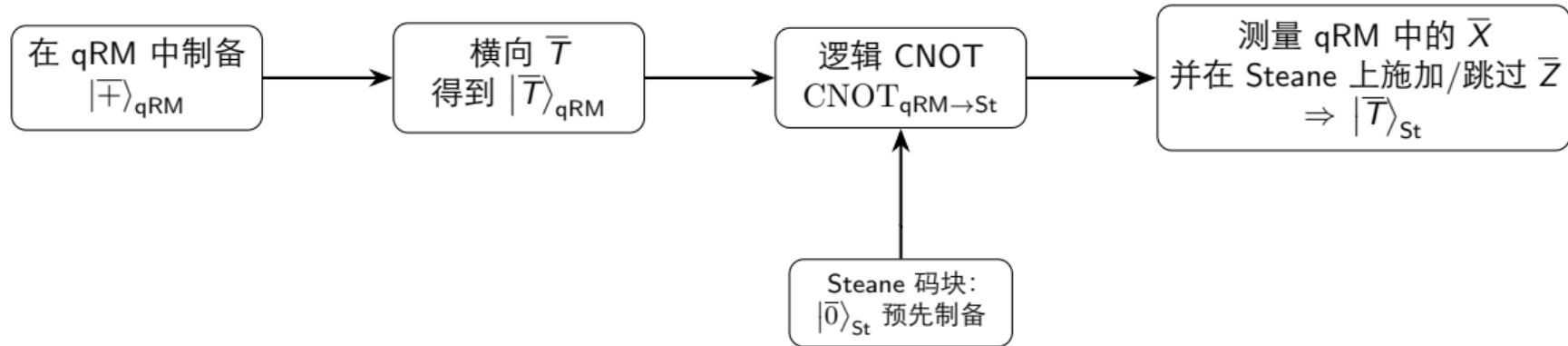
$$\mathcal{C}_{\text{Steane}} \otimes \mathcal{H}_{\text{gauge}} \subseteq \mathcal{C}_{\text{qRM}}.$$

- 码切换的操作可以理解为：

- 改变一组稳定子（增加或减少其中一部分）；
- 相当于**固定/释放**一部分 gauge 自由度；
- 在保持逻辑信息不变的前提下，改变我们“采用哪一套编码”。

- 在本协议中：从“3D qRM 编码 + 8 个 gauge 比特”切换到“2D Steane 编码 + 其余比特”。

# 码切换协议：逻辑电路流程图



这里给出逻辑层流程图；真正的物理实现还需配合 flag qubit 的综合提取和 postselection。

# flag qubit：保证码切换过程的容错性

- 码切换中有两类关键子过程：
  - ① 逻辑态初始化： $|\bar{0}\rangle_{\text{St}}$ 、 $|\bar{+}\rangle_{\text{qRM}}$ ；
  - ② 跨码逻辑 CNOT 所需的稳定子测量 / syndrome 提取。
- 若直接用传统稳定子测量电路：
  - 单个两比特门错误可能扩散成高权重错误，超出  $d = 3$  码可纠范围；
  - 会在码切换时引入不可纠正的逻辑错误。
- Daguerre & Kim 的做法：
  - 使用 flag qubit：在综合电路的关键位置插入额外辅助比特；
  - 一旦发生危险错误模式，flag 比特以高概率翻转；
  - 检测到 flag 或综合不满足条件时，整轮丢弃 (postselection)；
  - 在允许单故障模型下，剩余轮次的错误被限制在可纠范围内。
- 因此整个码切换协议是单故障容错的，对应一个伪阈值。

# 理论数值结果： $10^{-3}$ 噪声下的逻辑魔法态

- 噪声模型：均匀电路级噪声，两比特门后跟随强度为  $p = 10^{-3}$  的局部完全去极化，单比特门同量级，测量与初始化也按同一  $p$  模型处理。
- 数值模拟使用专门的稳定子模拟器，通过将 T 门编码为适当的非 Clifford 噪声通道来有效模拟。
- 主要结果（在 Steane 码上得到的逻辑魔法态）：

$$1 - F_{\text{logical}} \approx (4.6 \pm 1.6) \times 10^{-5},$$
$$P_{\text{accept}} \approx 84\%.$$

- 若在最终阶段再叠加一次逻辑层 postselection（仅接受综合全零的输出）并做多点拟合外推：

$$1 - F_{\text{logical}} \sim 5.1 \times 10^{-7},$$

即逻辑魔法态 fidelity 达到 0.9999995 的级别。

- 对比传统多级魔法态蒸馏，这个成本在小型处理器上已经非常可观。

# 实验实现：28 比特离子阱处理器上的码切换

- 硬件平台：一台可编程的囚禁离子量子处理器。
  - 线性离子链，几乎全连通两比特门；
  - 支持中途测量与重置，适合实现 flag 方案和 syndrome 提取。
- 资源使用：
  - 15 个物理比特：编码 qRM 逻辑比特；
  - 7 个物理比特：编码 Steane 逻辑比特；
  - 剩余比特：作为 flag qubit、辅助测量、空闲缓冲；
  - 总共使用 28 个比特（包含所有辅助）。
- 实验流程（高度概括）：
  - ① 在 qRM 和 Steane 两个块中分别初始化逻辑态；
  - ② 运行包含 flag 的综合提取电路，完成横向 T 与跨码 CNOT；
  - ③ 根据 flag 与综合结果接受/拒绝该轮；
  - ④ 对接受轮进行逻辑态层析（包括单拷贝与双拷贝）。

# 实验结果：逻辑魔法态质量与伪阈值

- 通过误差纠正后的逻辑态层析得到的逻辑 infidelity 上界：

$$1 - F_{\text{logical}} \leq 5.1(2.7) \times 10^{-4},$$

这是目前为止编码魔法态的最优实验结果之一。

- 与底层器件噪声的对比：

$$\rho_{2Q} \approx 1.05(8) \times 10^{-3},$$

$$\rho_{\text{SPAM}} \approx 1.38(12) \times 10^{-3},$$

即逻辑魔法态的 infidelity 比两比特门错误率还低约一倍。

- 这说明：

- 器件噪声已经低于码切换协议的伪阈值；
- 通过适当的码切换和 postselection，可以在当前硬件上真实获得“逻辑优于物理”的魔法态资源；
- 这为后续叠加少量魔法态蒸馏、构建小规模容错 T 门库奠定了实验基础。

# 四个核心问题的统一回答

## ① 为什么魔法态重要？

因为 Clifford 门集合本身是有限且可经典模拟的，魔法态是非 Clifford 资源的标准货币，通过魔法态注入得到 T 门，从而实现通用容错量子计算。

## ② 为什么要用这种方法来培育魔法态？

传统多轮蒸馏资源开销巨大。利用具有横向 T 门的编码 + 码切换，在小型设备上就能直接制备高保真逻辑魔法态，显著减少整体资源。

## ③ 这种方法具体怎么操作？

数学上可分解为：在码 A 中制备  $|\bar{T}\rangle_A$ ，再通过逻辑 CNOT + 测量 + Pauli 框架纠正实现逻辑一比特传输，把态转移到码 B。

## ④ 有什么优势？

前端用码切换拉高单个逻辑魔法态的保真度，后端蒸馏轮数大幅减少甚至可免；在现有平台（囚禁离子、中性原子等）上更贴近实验条件。

# 中性原子量子计算控制系统：整体架构

- 分层结构：

- 上位机软件：量子程序与实验序列描述（Python / QUA 等）
- 实时控制硬件：FPGA + AWG/DAC + DDS + 数字 I/O
- 实验装置：真空腔体、冷却与囚禁光路、Rydberg 激光、成像系统

- 典型工作流：

- 上位机编译脉冲序列  $\Rightarrow$  下载到控制硬件
- FPGA 时序引擎触发 AWG/DDS、激光快门、相机曝光等
- 实验结束后：相机 / 探测器读出  $\Rightarrow$  上位机分析

- 时序与同步：

- 公共 10 MHz 参考时钟 + 硬件触发 (TTL) 统一各设备
- 目标：纳秒级抖动、微秒级反馈（测量-反馈-继续运算）

# 上层软件与编程接口：Pulser / ARTIQ / OPX

- Pasqal 软件栈
  - **Pulser**: 开源 Python 库, 直接在脉冲层描述中性原子量子程序
  - 支持定义原子阵列几何、Rydberg/拉曼脉冲序列、数模混合算法
  - Pulser Studio: GUI 版, 无代码拖拽设计中性原子量子电路
- 通用实验控制框架
  - **ARTIQ + Sinara**: Python DSL 写实验序列, 内核编译到 FPGA
  - 强项: 确定性时序 (ns 级)、多通道 TTL/DAC/DDS 模块化扩展
  - **labscript suite**: Python + GUI, 偏“实验室级”多仪器调度
- 集成控制箱 (Quantum Machines / QuEra 等)
  - **Quantum Machines OPX(+)**:
    - 使用 QUA 语言在“脉冲级”编程
    - 内部 FPGA + PPU 自动处理时序、循环、条件分支与实时反馈
  - **QuEra**: 对外主要通过云 (如 AWS Braket) 提供接口, 内部同样采用“高层电路  $\Rightarrow$  脉冲编译器  $\Rightarrow$  控制箱”的结构

# 控制硬件 I: AWG / DAC / DDS 与激光调制

- 作用: 产生驱动 AOM/AOD/EOM 的射频与微波波形
  - 冷却 / MOT 光、光镊阵列、Raman / Rydberg 激光的幅度 / 频率 / 相位调制
  - 典型时间尺度:  $\mu\text{s}$  甚至  $\text{ns}$  级脉冲整形
- 常见方案
  - 商用多通道 AWG (Spectrum / Keysight / Tektronix)
    - 采样率:  $\sim 0.5\text{--}1.25 \text{ GS/s}$ , 14–16 位分辨率
    - 多卡同步: 共享 10 MHz 时钟和触发, 抖动可做到  $< 10 \text{ ns}$
  - DDS + 放大器驱动 AOM/AOD
    - DDS 频率几十–数百 MHz, 用于频移与多音 (多陷阱) 控制
    - 射频功放输出功率典型  $+30 \text{ dBm}$  级
- 开源/低成本板卡举例
  - **Red Pitaya STEMlab 125-14**
    - FPGA: Xilinx Zynq-7010 SoC (双核 ARM + FPGA)
    - DAC/ADC:  $2 \times 14 \text{ bit}$ , 125 MS/s
    - 数字 I/O:  $16 \times 3.3 \text{ V}$  GPIO
    - 通过以太网 + SCPI / 自定义协议远程控制

# 控制硬件 II: FPGA / 数字 I/O / 成像与同步

- **FPGA 时序引擎**

- 负责执行“时间表”：在精确时间点输出 TTL、更新 AWG/DDS 寄存器
- 通常集成在：
  - ARTIQ Kasli 控制板 + Sinara TTL/DDS/DAC 模块
  - QM OPX(+) 等集成控制箱

- **数字 I/O 与触发**

- 多路 TTL 输出：控制快门、相机曝光、磁场线圈开关、锁频启停
- 多路数字输入：计数光电探测器、时间标记事件，实现测量反馈
- 典型接口：3.3V TTL / LVDS，部分板卡支持 ns 级边沿时间戳

- **典型控制箱参数例子**

- **QM OPX+** (大致)：
  - 模拟输出：~10 通道，1 GS/s，16 位，~400 MHz 带宽
  - 数字输出：~10 通道，1 GS/s，3.3V TTL
  - 模拟输入：若干通道，~1 GS/s，用于快速测量与反馈

- **Sinara / ARTIQ**：

- Kasli：FPGA 主控 + 多路 TTL I/O
- Urukul：4× DDS 输出 (射频)
- Zotino：多通道 DAC (0–10V) 控制激光功率、电压偏置等

# 科研搭建：需要准备的软硬件与环境（概览）

- 物理平台

- UHV 真空系统：真空腔体（带窗口）、离子泵/涡轮泵、真空规
- 原子源：铷 / 铯蒸气源与加热控制
- MOT 磁场线圈 + 快速电源，冷却用激光光路

- 光学与激光

- 冷却 / 再泵激光、光镊/光晶格激光、Rydberg 激发激光、Raman 或微波源
- 光学平台 + 高 NA 物镜、AOM/AOD/SLM、偏振与功率稳定链路
- 成像系统：EMCCD / sCMOS + 相机接口卡（CameraLink / CoaXPress / GigE）

- 电子与控制

- AWG/DDS/DAC：激光/射频调制，多通道同步
- FPGA/控制箱：ARTIQ+Sinara、Red Pitaya 阵列，或 QM OPX(+)
- 数字 I/O：TTL 触发、计数模块、时间标记/计时器
- 通信：以太网 / USB / PXI；统一 10 MHz 时钟 + 触发分配网络

- 软件与上层栈

- Python 环境：Pulser、ARTIQ/labscript、PyVISA、相机 SDK 等
- 若使用 OPX：QUA + 官方 SDK
- 数据管理与分析：Jupyter/MATLAB/Mathematica 等

# 谢谢大家！

欢迎讨论与提问～