

Example 4 — COPY 与 XOR 张量的数学推导

本文档把论文 Example 4 中的 COPY（复制）和 XOR（奇偶/加法模 2）张量的定义与性质写成数学推导，并展示如何通过张量收缩得到 CNOT 门。

1. COPY 张量的定义（秩 3）

在二值（0/1）情形下，定义三阶 COPY 张量 COPY^i_{jk} （我们写作输出为上标，输入为下标的约定只是记号）为：

$$\mathrm{COPY}^i_{jk} = \begin{cases} 1, & i = j = k, \\ 0, & \text{否则.} \end{cases}$$

直观地，COPY 把一个二值输入复制到两个输出（或相反方向，取决于你如何把索引配对）。在图形表示中，COPY 节点的三个腿当值都相同（全 0 或全 1）时取 1，其它配置为 0。

因此，当把 COPY 的一个腿作为输入（例如值 0 或 1）并把其它两腿接到外部时，会得到两个相同的输出（公式见论文的图 (23)-(24)）。

2. XOR（parity）张量的定义（秩 3）

XOR（或 parity）张量在论文中用带圈的加号表示。它在索引分配包含偶数个 1 时取值 1，否则 0。等价地，若输出等于输入的按位 XOR，则输出为 1：

$$\mathrm{XOR}^r_{qs} = \begin{cases} 1, & r = q \oplus s, \\ 0, & \text{否则.} \end{cases}$$

在二值域上，XOR 的取值可列举（三条腿的所有配置，若有偶数个 1 则为 1）。

论文还给出了组件的代数表达（28b），在做一般化计数时很有用。对于本处的二值示例，上述定义已足够。

3. COPY 与 XOR 之间的 Hadamard 关系

论文给出 COPY 与 XOR 之间通过 Hadamard 基变换（局部 H 门）相互对应的关系：

$$\frac{1}{\sqrt{2}} \text{XOR tensor} = (H \otimes H \otimes I) \cdot \text{COPY tensor} \cdot (H \otimes H \otimes I)$$

直观含义：在 Hadamard 基 ($|+\rangle, |-\rangle$) 下，XOR 行为像复制操作（或其比例缩放形式），所以 XOR 可以看成是在另一基下的“复制”。论文写成更紧凑的图形等式（公式 (26)-(27)）。

4. 通过收缩重建 CNOT（公式 (29)）

论文中给出 CNOT 可以视为两个三阶张量（COPY 与 XOR）在中间索引 m 上收缩得到的秩 4 张量：

$$\sum_m \text{COPY}_i^{qm} \text{XOR}_{mj}^r = \text{CNOT}_{ij}^{qr}.$$

这里的指标排列与论文一致：左边的 COPY 把输入索引 i 和中间索引 m 以及输出 q 联系起来；右边的 XOR 把中间索引 m 与输入 j 和输出 r 联系起来。把 m 求和（收缩）后，得到一个关于 q, r, i, j 的四阶张量，其分量就是 CNOT 在这些输入输出索引下的分量。

用更计算机友好的数组表示（COPY[out, i, m], XOR[out, m, j]），可以用 Python/NumPy 写成：

$$T_{qrij} = \sum_m \text{COPY}[q, i, m] \cdot \text{XOR}[r, m, j].$$

然后把 T 重塑为矩阵（行索引复合为 (q, r) ，列索引复合为 (i, j) ），即可比较标准 4×4 的 CNOT 矩阵。

在二值情形下，按此收缩确实会恢复标准 CNOT（论文公式 (29)）；这是因为 COPY 负责把控制位值沿中间连线分发，而 XOR 根据控制值与目标值的异或结果在输出处生成结果——两者合并即实现控制翻转逻辑。

5. 分量形式（论文的 (28a) 与 (28b)）

论文给出了适度一般化的分量表达式：

$$\text{COPY}_{jk}^i = (1-i)(1-j)(1-k) + ijk,$$

以及

$$\text{XOR}_{rs}^q = 1 - (q + r + s) + 2(qr + qs + rs) - 4qrs,$$

这些多项式在二值 $\{0,1\}$ 上等价于前面的取值定义（仅输出为 0/1 时生效）。

解释：

- COPY 的多项式表示把全 0 $((1-i)(1-j)(1-k)=1)$ 和全 1 $(ijk=1)$ 情形单独选出并赋 1。
 - XOR 的多项式形式是把奇偶性用布尔多项式写出，能在代数消去或符号推导时提供方便。
-

6. 数值验证（用 NumPy）

在代码实现 `example4.py` 中，我实现了上述的 COPY 与 XOR，并用 Einstein-summation（`np.einsum`）对中间索引 m 求和，重塑为 4×4 矩阵，与标准 CNOT 比较，Frobenius 范数差为 0：这数值上验证了论文的等式（29）。

示例（伪代码）：

```
COPY = ... # shape (2,2,2)
XOR  = ... # shape (2,2,2)
T = np.einsum('qim,rmj->qrij', COPY, XOR)
T_mat = T.reshape(4,4)
assert np.allclose(T_mat, standard_cnot)
```

7. 小结

- COPY 在原基下实现“复制”两个比特的值；XOR 在计算基对应奇偶检查/异或运算。
- 通过 Hadamard 基转换，XOR 与 COPY 在不同基之间互相关联，这一点在量子电路分析与张量网络的基变换中特别有用。
- 用张量收缩把 COPY 与 XOR 组合起来可以还原出 CNOT 的逻辑，这揭示了量子门可以分解为更简单的张量构件（便于符号推导和网络计算）。

该文档已保存为 `docs/example4_derivation.md`，并且配套的数值实现位于 `example4.py`。