文章编号：2095-4980(2016)06-0860-07

具有珍珠项链结构的量子稳定子卷积码

邱鹏辉，陈晓光

(复旦大学 通信科学与工程系，上海 200433)

摘　要：阐述了量子和经典编码之间的关系，基于稳定子码的概念提供了一个高效的多项式来描述稳定子码，通过给定的生成元计算出其标准型。将传统的量子分组码的编码方法扩展到量子卷积码领域，根据量子电路的优化准则转换成具有高度结构化、电路易于实现的珍珠项链结构，避免了灾难性错误的传播，简化了量子编码最小存储的计算。

关键词：量子通信；量子卷积码；稳定子码；珍珠项链结构；Calderbank–Shor–Steane 码

中图分类号：TN911.2　　**文献标志码：**A　　**doi：**10.11805/TKYDA201606.0860

Quantum stabilizer convolutional code with pearl-necklace structure

QIU Penghui, CHEN Xiaoguang

(Department of Communication and Science Engineering, Fudan University, Shanghai 200433, China)

Abstract: The relationship between quantum and classical coding is introduced, and an efficient polynomial is provided to describe the stabilizer codes. Its standard is calculated to generate quantum convolutional codes. Then the method based on classic quantum coding is applied to the realm of convolutional codes. It is converted into a pearl-necklace structure according to the optimization criterion of quantum circuit. The circuit of the structure is easy to implement, which can avoid the catastrophic error propagation, and simplify the calculation of quantum coding minimum storage.

Keywords: quantum communication; quantum convolutional code; stabilizer code; pearl-necklace; Calderbank–Shor–Steane(CSS) coding

近年来，从无限稳定子矩阵中得到量子卷积码的基本理论逐渐发展起来[1]。随着研究的深入，量子纠错码理论随着各种编码方案的发展而不断得到完善和发展。其中至今仍发挥重要作用的一种方法是 1996 年由 Calderbank, Shor 和 Steane 采用经典线性分组纠错码，利用 2 个特殊的经典二元纠错码设计出量子纠错码系统的 CSS 码构造方法。受 Chau 提出的量子卷积码编码方案的启发[2-3]，本文依据量子稳定子码的特点和经典卷积码的编码技术，将 CSS 型编码结构扩展到量子卷积码，然后转换成珍珠项链结构的量子卷积码。

1　量子编码与经典编码的联系与转换

就量子纠错码而言，输入量子比特信息产生错误等效于对其进行泡利操作，可写成包含泡利矩阵 $\boldsymbol{X},\boldsymbol{Y},\boldsymbol{Z}$ 的形式。如，操作 $\boldsymbol{E}=\boldsymbol{ZXIIYI}=(\boldsymbol{IIIIXI})\cdot(\boldsymbol{ZIIIZI})$，可以根据如操作算子作用在量子比特上则将该位置 1，反之置 0 这一规则，将 $\boldsymbol{X}$ 操作表示为一个二进制序列串，$\boldsymbol{Z}$ 操作依据同一原理。因此上述式子可以转换为 $\boldsymbol{e}=[000010|100010]$，此序列的长度为 $2N$。同时为了方便，可以将 $\boldsymbol{E}$ 写成 $\boldsymbol{E}=\boldsymbol{Z}_1\boldsymbol{X}_2\boldsymbol{Y}_5$。

稳定子码的可交换性表现为生成编码的扭积行正交[4]。具体如下：如果生成项的第 k 行为：

$$r_k=(x_k\,|\,z_k) \tag{1}$$

那么第 k 行和第 l 行的扭积可以表示为：

$$r_k\bullet r_l=(x_kz_l+x_lz_k)\bmod 2 \tag{2}$$

式中 x_kz_l 定义为 $x_kz_l=\sum_j x_{kj}z_{lj}$ 的内积。要使得扭积为 0，当且仅当第 k 行和第 l 行具有偶数个操作算子不同。一个量子校验矩阵 $\boldsymbol{A}=(\boldsymbol{A}_1|\boldsymbol{A}_2)$ 对于所有的行满足 $\boldsymbol{A}_1\boldsymbol{A}_2^{\mathrm{T}}+\boldsymbol{A}_2\boldsymbol{A}_1^{\mathrm{T}}=0$，也就是说其满足扭积行正交。

收稿日期：2015-11-04；修回日期：2015-12-13

考虑到 $\boldsymbol{A}$ 是校验矩阵，$\boldsymbol{e}$ 为二进制错误矢量，那么带噪声的量子校验码恰好是经典码的 $\boldsymbol{eA}^{\mathrm{T}}$。在这种形式下，对误差矢量 $\boldsymbol{e}$ 和量子校验矩阵中对应的行进行经典的标量点积运算，若结果为 0，表明错误算子和响应的稳定子编码的行互易；若结果为 1，则表明它们反互易。因此，稳定子编码的性质可以从一些特殊的经典编码中推断出来。给定一个 $M\times 2N$ 的二进制矩阵，其具有任意 2 行的扭积为 0，那么等效于用 N 量子比特可以构建 $N-M$ 个量子比特的编码器。

2　CSS 码

CSS 码结合了级联码和对偶码 2 种编码思想，不仅与经典线性分组具有良好的对应关系，从而可以利用经典纠错码来构造量子纠错码；而且具有非常直观的代数结构，易于分析。CSS 码可以由一个自对偶或者一对满足扭积行正交关系的经典二元线性码构造而成。但通过自对偶构造的 CSS 码一定会有长度为 4 的块码，这对译码具有消极影响。CSS 码的效率也许不如一般的量子编码，但很容易从已知的经典编码中获得，并且 CSS 型的简单形式往往使它们适合于其他用途[5]。如，七量子位编码适用于容错计算。

CSS 码一般具有如下形式：

$$\boldsymbol{A}=\left[\begin{array}{c|c}\boldsymbol{H} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{G}\end{array}\right]\boldsymbol{HG}^{\mathrm{T}}=\boldsymbol{0} \tag{3}$$

式中 $\boldsymbol{H}$ 和 $\boldsymbol{G}$ 都是 $M\times N$ 的矩阵，条件 $\boldsymbol{HG}^{\mathrm{T}}=\boldsymbol{0}$ 确保了 $\boldsymbol{A}$ 满足上文所说的扭积行正交条件。因此，就是 $2M$ 个稳定子条件应用在 N 个量子比特中，用 N 量子比特可以构建 $N-2M$ 个量子比特的编码器。

CSS 码中有一个特殊类别，结合经典理论中对偶码的概念，当 $\boldsymbol{H}$=$\boldsymbol{G}$ 时又被称之为 CSS 对偶码。其量子校验矩阵具有如下形式[6]：

$$\boldsymbol{A}=\left[\begin{array}{c|c}\boldsymbol{H} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{H}\end{array}\right]\boldsymbol{HH}^{\mathrm{T}}=\boldsymbol{0} \tag{4}$$

$\boldsymbol{HH}^{\mathrm{T}}=\boldsymbol{0}$ 等效于 $C^{\perp}(\boldsymbol{H})\subset C(\boldsymbol{H})$ 成立，$C(\boldsymbol{H})$ 是码字中 $\boldsymbol{H}$ 的奇偶校验矩阵；$C^{\perp}(\boldsymbol{H})$ 是它的对偶矩阵。

3　量子卷积编码

3.1 量子卷积码

量子卷积码被看成是量子稳定子编码的无限生成元的一种形式[7]。从稳定子分组码到量子卷积码的近似描述，在文献[8]中用多项式的形式已描述出来。

对于一个稳定子编码，可以用多项式矩阵的形式表示：

$$\boldsymbol{G}(D)=(A_X(D)|A_Z(D)) \tag{5}$$

需要满足扭积行正交条件，因此，对于 $\boldsymbol{G}$ 中的 2 个元素 $\boldsymbol{A}=(A_X|A_Z)$ 和 $\boldsymbol{B}=(B_X|B_Z)$，应该满足式(6)：

$$A_XB_Z+A_ZB_X=0\Leftrightarrow \boldsymbol{AB}=\boldsymbol{BA} \tag{6}$$

以此类比，多项式形式满足扭积行正交条件应满足方程(7)：

$$A_X(D)A_Z(1/D)^t-A_Z(D)A_X(1/D)^t=0 \tag{7}$$

根据 CSS 的特殊结构，若卷积码的生成矩阵为 $\boldsymbol{G}(D)=[g_1(D),g_2(D),g_3(D),\cdots,g_n(D)]$，其当且仅当 $\sum_{l=1}^{n}g_1(D)g_1(1/D)=0$ 等式成立，即 $\boldsymbol{G}(D)$与它的所有延时正交时满足正交条件，对于其对偶码 $\boldsymbol{H}(D)$同样如此。

量子卷积码的稳定子卷积码表示具有如下形式[9-10]：

$$\boldsymbol{M}=\left(\begin{array}{cc}\begin{bmatrix}M_{0,1}\\ \vdots\\ M_{0,n-k}\end{bmatrix} & \\ \begin{matrix}\\ n-k\end{matrix}\begin{bmatrix}M_{1,1}\\ \vdots\\ M_{1n-k}\end{bmatrix} & \\ \begin{matrix}m & n\end{matrix} & \ddots\end{array}\right) \tag{8}$$

矩阵的每一行代表一个 $M_{j,i}$，每一列代表不同的量子比特，重叠部分称为编码存储。

3.2 编译码方法

3.2.1 理论分析

根据量子稳定子编码理论[11]，给定一个量子校验矩阵，基于线性代数，将稳定子编码的矢量偶矩阵进行高斯–约旦消元，得到标准型：

$$\boldsymbol{A}_{\mathrm{q}}=\left(\begin{array}{ccc|ccc}\boldsymbol{I} & \boldsymbol{A}_1 & \boldsymbol{A}_1 & \boldsymbol{B} & \boldsymbol{C}_1 & \boldsymbol{C}_2\\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{D} & \boldsymbol{I} & \boldsymbol{E}\end{array}\right) \tag{9}$$

针对 CSS 码，由于其特殊结构，可以将其写成标准形式(10)，且满足稳定子编码理论。

$$\boldsymbol{M}_{\mathrm{q}}=\left(\begin{array}{ccc|ccc}\boldsymbol{A}(D) & \boldsymbol{B}(D) & \boldsymbol{C}(D) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{D}(D) & \boldsymbol{E}(D) & \boldsymbol{F}(D)\end{array}\right) \tag{10}$$

根据上述的标准型计算其比特翻转和相位翻转，用 $\overline{\boldsymbol{X}}$ 和 $\overline{\boldsymbol{Z}}$ 来表示。设

$$\overline{\boldsymbol{X}(D)}=(u_1(D),u_2(D),u_3(D)\,|\,v_1(D),v_2(D),v_3(D))$$

由于 $\overline{\boldsymbol{X}}$ 乘以稳定子编码中的任意元素，得到的结果仍然为 $\overline{\boldsymbol{X}}$，由此条件可以得到 $u_1(D)=0$，$v_2(D)=0$。因此 $\overline{\boldsymbol{X}(D)}$ 可以化简如下：

$$\overline{\boldsymbol{X}(D)}=\big(0,u_2(D),u_3(D)\,|\,v_1(D),0,v_3(D)\big) \tag{11}$$

根据稳定子码理论，由于 $\overline{\boldsymbol{X}(D)}\in\boldsymbol{C}(D)$，因此 $\overline{\boldsymbol{X}(D)}$ 必须和 S 的所有生成元都互易。因此可以推出式(12)，将式(10)和式(11)代入，可以得到式(13)。

$$\overline{\boldsymbol{X}(1/D)}\boldsymbol{M}_{\mathrm{q}}^{\mathrm{T}}=\left(\begin{array}{c}\boldsymbol{0}\\ \boldsymbol{0}\end{array}\right) \tag{12}$$

$$\left(\begin{array}{c}\boldsymbol{0}\\ \boldsymbol{0}\end{array}\right)=\big(0,u_2(D),u_3(D)\,|\,v_1(D),0,v_3(D)\big)\left(\begin{array}{ccc|ccc}\boldsymbol{A}(D) & \boldsymbol{B}(D) & \boldsymbol{C}(D) & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0}\\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{D}(D) & \boldsymbol{E}(D) & \boldsymbol{F}(D)\end{array}\right)^{\mathrm{T}} \tag{13}$$

另一方面，由于泡利矩阵必须满足互易，因此式(14)成立：

$$\boldsymbol{0}=\overline{\boldsymbol{X}(D)}\cdot\overline{\boldsymbol{X}(D)}^{\mathrm{T}}=(0,u_2(D),u_3(D)\,|\,v_1(D),0,v_3(D))\bullet(0,u_2(D),u_3(D)\,|\,v_1(D),0,v_3(D))^{\mathrm{T}}=u_3(D)v_3(D)^{\mathrm{T}}+u_3(D)^{\mathrm{T}}v_3(D) \tag{14}$$

综合式(13)和式(14)联合求解，可以得到最终 $\overline{\boldsymbol{X}(D)}$ 的值：

$$\overline{\boldsymbol{X}(D)}=(0,\boldsymbol{F}^{\mathrm{T}}(1/D)\boldsymbol{E}^{-1}(1/D)\Lambda(D),\Lambda(D)\bullet\boldsymbol{I}\,|\,\boldsymbol{0},\boldsymbol{0},\boldsymbol{0}) \tag{15}$$

式中 $\Lambda(D)$ 是 $GF_2(D)$ 上的非零多项式。必须注意的是，$\Lambda(D)$ 必须是非零多项式的最小维度。要使得标准多项式形式衍生编码器是非灾难性的，当且仅当 $\Lambda(D)$ 是一个单项。因此，在具体操作取 $\Lambda(D)$ 值的过程中要注意这一点。

类比求 $\overline{\boldsymbol{X}(D)}$ 的值，同理通过计算可以得到 $\overline{\boldsymbol{Z}(D)}$ 的值，不同的地方在于泡利矩阵必须满足互易关系，将目标算子 $\overline{\boldsymbol{X}(D)}$ 改为 $\overline{\boldsymbol{Z}(D)}$ 后，$\overline{\boldsymbol{Z}_i(D)}$ 和 $\overline{\boldsymbol{X}_j(D)}$ 在 $i=j$ 时互易，但是在 $i\neq j$ 时两者是反互易的。因此，在计算时尤其要注意这一点。最终得出 $\overline{\boldsymbol{Z}(D)}$ 算子如下：

$$\overline{\boldsymbol{Z}(D)}=(\boldsymbol{0},\boldsymbol{0},\boldsymbol{0}\,|\,\boldsymbol{C}^{\mathrm{T}}(1/D)\boldsymbol{A}^{-1}(1/D)/\Lambda(1/D),\boldsymbol{0},\boldsymbol{I}/\Lambda(1/D)) \tag{16}$$

3.2.2 编码电路

对量子卷积码的编码操作过程，可以依据文献[5]将其初步表示为式(17)的形式：

$$\left|c_{0,1},c_{1,1}\cdots,c_{q-1,k}\right\rangle\rightarrow\left(\prod_{i,j}\frac{1+M_{j,i}}{\sqrt{2}}\right)\prod_{r,s}\overline{\boldsymbol{X}}_{s,r}^{C_{s,r}}\left|0,0,\cdots,0\right\rangle \tag{17}$$

但依据上述定义中稳定子编码标准型可知，有一半的算子右边矩阵全为 0，另一半算子左边矩阵全为 0，这意味着编码时有一半的算子只涉及比特翻转操作，另一半的算子只涉及相位翻转操作。若将仅包含相位翻转操作的算子先作用在全 $|0\rangle$ 态上，那么在 $|0\rangle$ 态上进行相位翻转操作等于不操作。因此可以先不用考虑涉及比特翻转操作的算子，从而大大简化编码电路。

依据式(17)，注意到稳定子编码标准型 $\boldsymbol{M}$ 的前 r 个量子位只需要简单的比特翻转操作，这意味着进行 $(1+M_{j,i})$ 操作等于将第 i 位变成 $|0\rangle$ 与 $|1\rangle$ 的和，因此对前 r 个量子位进行 Hadamard 操作，使前 r 位变成 $|0\rangle+|1\rangle$，然后以第 i 位作为控制位，根据 $M_{i,j}$ 进行相应比特翻转操作，编码完成。

3.3 举例说明

为更直观地证明正确性，举个简单的例子来进行验证，假设经典自正交码[4,1,2]的生成矩阵为 $\boldsymbol{G}=[1,D,D,1]$，即 $g_1=X_1X_4;g_2=X_2X_3;g_3=Z_1Z_4;g_4=Z_2Z_3$，其满足自正交的条件：

$$\sum_{l=1}^{n} g_1(D)g_1(1/D)=1+D\times D^{-1}+D\times D^{-1}+1=0 \tag{18}$$

由此码字可以构成[4,2,8]量子卷积码及其稳定子生成元、稳定子偶矩阵和响应的翻转算子。基于 Matlab 平台编写相应的模拟程序并运行，得：

$$\boldsymbol{G}_std=\begin{bmatrix}
\boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} \\
\boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} \\
\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} \\
\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} \\
\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} \\
\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} \\
\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{I} & \boldsymbol{X} & \boldsymbol{X} & \boldsymbol{I} \\
\boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{I} & \boldsymbol{Z} & \boldsymbol{Z} & \boldsymbol{I}
\end{bmatrix} \tag{19}$$

$$\boldsymbol{M}\_std=\begin{bmatrix}1 & D & D & 1 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 1 & D & D & 1\end{bmatrix} \tag{20}$$

$$\overline{\boldsymbol{X}}=\begin{bmatrix}0 & 1 & D & 0 & 0 & 0 & 0 & 0\\ 0 & D & 0 & D & 0 & 0 & 0 & 0\end{bmatrix},\ \overline{\boldsymbol{Z}}=\begin{bmatrix}0 & 0 & 0 & 0 & 1 & 0 & D & 0\\ 0 & 0 & 0 & 0 & D & 0 & 0 & D\end{bmatrix} \tag{21}$$

其中，式(19)表示的是根据生成矩阵得到的稳定子生成元；式(20)表示的是稳定子矢量偶矩阵；式(21)表示计算得出的 $\overline{\boldsymbol{X}(D)}$ 和 $\overline{\boldsymbol{Z}(D)}$。然后根据其编码网格的步骤，可以得到[4,2,8]量子卷积码的编码电路图，如图 1 所示。

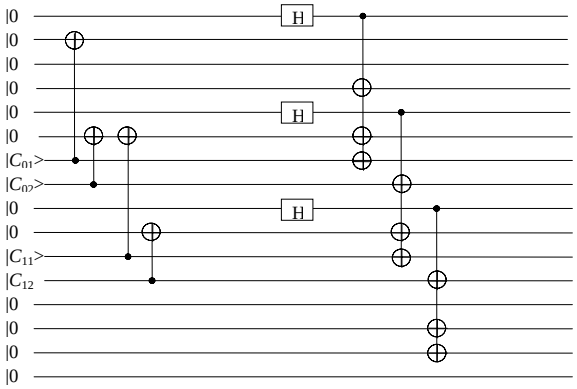


Fig.1 [4,2,8]quantum convolutional coding circuit
图 1 [4,2,8]量子卷积码编码网络

4 具有珍珠项链结构的量子卷积码

2010 年 Houshmand 和 Hosseini-Khayat 设计了一个算法，以珍珠项链结构进行量子编码[12]，它阐述了在珍珠项链编码器中，算法的构建和图的搜索所需的最少时间和量子比特流的数量呈二次方的关系。而这个算法用于量子卷积码使得珍珠项链量子卷积编码器成为可能。在量子卷积编码器和后续的纠缠辅助编码器中，珍珠项链编码器的这一方法显得尤为重要。本文利用原码及其对偶码的级联，分别纠正比特翻转和相位翻转错误的 CSS 量子卷积码，转换成具有珍珠项链结构的编码电路，并对其进行简化。

构建珍珠项链结构的量子卷积码有一个重要的因素，由于在量子情况下，单个位翻转可能传播到无限数量量子比特，因而非灾难性标准编码器 $\Lambda(D)$ 必须为一个单项式，它等效于仅使用 X 算子来对多项式形式进行有效说明。而相位翻转电路可以通过从目标到控制量子比特的控制非门电路来实现，并且对于 CSS 量子卷积码，它的每一个基本操作在珍珠项链编码器中，都对应于 CNOT 比特流。因此，CNOT 门在研究珍珠项链编码结构中发挥着重大的作用，很多相关的文献都是基于 CNOT 门来进行研究。

4.1 珍珠项链结构卷积码的构造

珍珠项链编码结构交替表现为几个相同的 U 门作用到量子比特流中，每一个 U 操作类比于珍珠项链的一颗珠子，因此这一编码表现被称为珍珠项链编码器。针对上文例子，可以将最后得到的编码网格通过变换，简单地说，就是通过交换编码网络中的门电路来构造出具有珍珠项链结构的量子卷积码。

交换编码电路中的门电路涉及到电路的优化规则问题。在量子领域，电路移动规则其实就是互易性的一种实际体现。假设有 3 个门电路 U_1,U_2,U_3 按顺序作用到一个电路中，那么电路可以简写成 $U_1U_2U_3$。如果 U_1,U_2 满足互

易性，即 U_1,U_2 满足 $[U_1,U_2]=0$，那么在电路中可以交换这两者的顺序，电路的功能并不会发生变化，因此电路可以简写成 $U_1U_2U_3$。这就是量子电路的优化规则，根据这个规则，可以简化电路或者将电路化成别的形式，以达到优化的目的。为直观显示，将例子中输出往后循环，交换得到所需要的珍珠项链结构，如图 2 所示。

对比图 1 和图 2 的电路实现，电路都是通过量子比特流 $\overline{CNOT}(3,2)(D^{-1})$ $\overline{CNOT}(4,2)(1)$ $\overline{H}(1)$ $\overline{CNOT}(1,4)(1)$ $\overline{CNOT}(1,2)(D)$ $\overline{CNOT}(1,3)(D)$ 来实现。因此，在实现珍珠项链结构的具体过程中，并不需要增加或者减少量子门操作，而是根据其量子比特流表达式，将原来整个的量子比特操作门(可以看成是一个巨大的 U 操作)分解成一小块一小块的量子比特操作门电路的合成，简单地说，就是分解 U 操作，交替表现为几个分解的 U 门作用到量子比特流。这样做的好处是由原来复杂的 U 操作变成一系列简单的分解操作，使得编码网格既具有高度结构化，又易于实现。

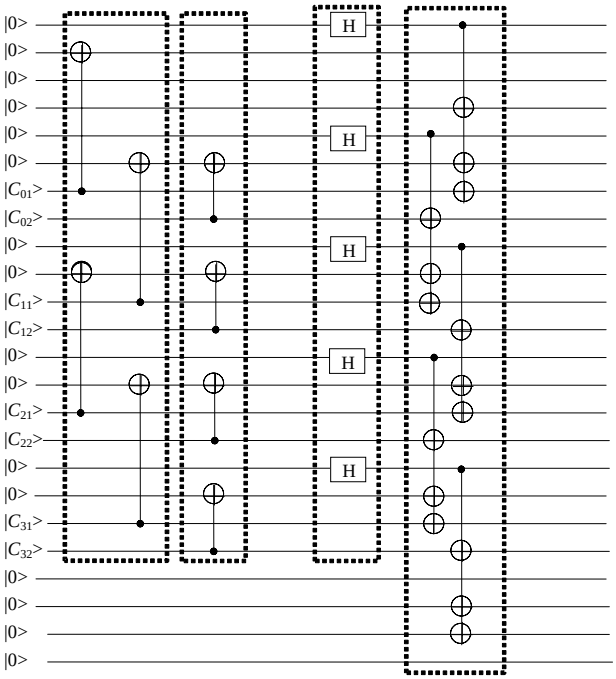


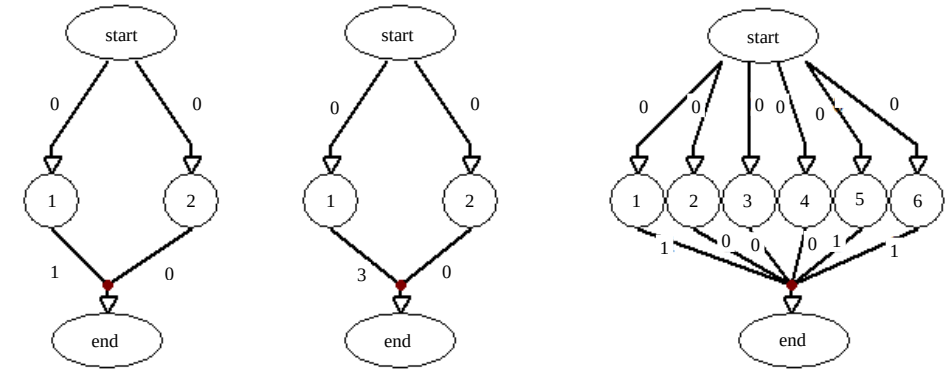
Fig.2 Circuit of the pearl-necklace quantum convolutional code
图 2 珍珠项链结构量子卷积码电路图

4.2 珍珠项链编码结构的存储需求

对珍珠项链编码器而言，计算存储需求的第 1 步是通过交换 CNOT 门的顺序来将其重新排列成一个卷积码编码器。为构造卷积编码器，首先必须找到珍珠项链编码器所有的每个量子比特流中单一量子门的集合。因此，把珍珠项链编码器中的所有门往右移动，并且在剩余的量子比特流中无限循环这一操作。

参照文献[11]可以发现，在珍珠项链编码结构的前半段，可以通过纯化的负性 CNOT 门进行转换，从而求得最小存储的卷积编码器。采用 2 种方式来求解珍珠项链编码器所需的最小存储，一种是 4 个比特流为 1 帧；另外一种是 2 个比特流为 1 帧。

首先考虑 4 个比特流为 1 帧的情况，其满足 CNOT 连续的比特流为：$\overline{CNOT}(3,2)(D^{-1})$ $\overline{CNOT}(4,2)(1)$，其状态图如图 3(a)所示。图 3(a)显示了在珍珠项链编码器中，门串流中源–目的和目的–源的不可交换性。图中最长路径是：开始->①->结束，权重等于 1。因此，这个卷积编码器的最小存储是 1 帧存储量子比特，也就是 4 量子比特的存储。



(a) four qubits for a frame (b) two qubits for a frame (c) pearl-necklace

Fig.3 Directed graph of encoding circuit
图 3 编码电路的有向图

在 2 个比特流为 1 帧的情况下，其 CNOT 连续比特流为：$\overline{CNOT}(1,2)(D^{-3})$ $\overline{CNOT}(2,2)(D^{-1})$，其状态图如图 3(b)所示。图 3(b)中最长路径是：开始->①->结束，权重等于 3。因此，这个卷积编码器的最小存储是 3 帧存储量子比特，也就是 6 量子比特的存储。由此结果可以看出，并不是帧越短越好，因此，选择合适的帧长也是相当重要

的。在本文例子中，量子传输信道为两输入四输出，因此，取帧长 4 量子比特为最佳。一般情况下，1 帧的长度应该等于输出的量子比特数。

对于珍珠项链编码电路的后半段，由于稳定子编码必有 H 门的出现，参考文献[10]，从而得出所需要的最小储存。综合文献[12–13]，可以得到满足图 2 的量子比特流信息：$\overline{CNOT}(3,2)(D^{-1})$ $\overline{CNOT}(4,2)(1)$ $\overline{H}(1)$ $\overline{CNOT}(1,4)(1)$ $\overline{CNOT}(1,2)(D)$ $\overline{CNOT}(1,3)(D)$，从而画出所得到的珍珠项链编码电路的最终存储状态图如图 3(c)所示。

图 3(c)中最长路径是：开始->①->结束(或者①替换成⑤,⑥)，权重都等于 1。因此，这个卷积编码器的最小存储是 1 帧存储量子比特，也就是 4 量子比特的存储。

上文所述说明了如何实现一个最小存储的卷积编码器，来执行和使用任意 CNOT 门的珍珠项链编码器进行一样的转换。本文方法是构建一个从属图，其定向性代表珍珠项链编码器中的量子比特流的不可交换性。这与图中寻找最小存储和最长路径采用了相同的原理。对所有可能的珍珠项链，可以首先执行优化量子卷积码编码器，因为有很多针对特定量子卷积编码器的珍珠项链编码器;还可以直接从多项式描述代码本身寻找方法来构造一个重复 $\boldsymbol{U}$ 矩阵。在移不变 Clifford 中还包含了对所有门串流的算法延伸，Clifford 群包括 H 门、P 门、控制 P 门和无限深度的 CNOT 门。

4.3 珍珠项链结构的优越性

具有珍珠项链结构的量子卷积码编码器，编译码复杂度低，网络结构非常简单，且由于其通过量子比特流的变换关系，编码网格又具有高度结构化，电路易于实现。

在通信信道上，一个量子比特的错误将会传播到其所有相关的量子位，这些量子位通过门电路被进一步错误传播，直到没有更多的门电路被应用，这就是灾难性传播问题。只有在通信过程中确保有限量子比特错误在传播过程中仅对有限数量的门操作起作用，才能保证所有的错误都是非灾难性的。具有珍珠项链结构的编码电路在满足 $\Lambda(D)$ 是单项式时满足这一条件，因此，其编码电路是非灾难性的。这一点对于量子通信理论来说尤其重要，其可以确保在通信过程中不会陷入无限错误的死循环中，是通信过程中必不可少的一个要求。

5　结论

量子信息是一门综合性非常强的交叉学科，涉及量子物理、经典信息科学、通信科学与技术、计算机与电子等多个学科领域。量子信息处理理论和技术在近年来飞速发展，在诸多方面已取得了非常惊人的成就，尤其是量子计算机的出现。卷积码无论在传统通信领域还是量子通信领域都发挥着重要而不可替代的作用。本文就 CSS 型特殊的量子卷积码为例，阐述了量子和经典编码之间的关系，并且基于稳定子编码将传统的量子分组码的编码方法扩展到量子卷积码领域，然后转换成珍珠项链结构；同时基于 Matlab 平台使得量子意义上的仿真成为可能，是后期进一步研究量子卷积码的基础。

参考文献：

[1] OLIVIER H,TILLICH J P. Quantum convolutional codes: fundamentals[J]. HAL-INRIA, 2004,54(9):4053–4068.

[2] CHAU H F. Quantum convolutional error-correcting codes[J]. Physical Review A, 1997,58(2):905–909.

[3] CHAU H F. Good quantum convolution error-correction codes and their decoding algorithm exist[J]. Physical Review A, 1999,60(3):1966–1974.

[4] CALDERBANK A R,RAINS E M,SHOR P W,et al. Quantum error correction and orthogonal geometry[J]. Physical Review Letters, 1997,78(3):405–408.

[5] GOTTESMAN D. Stabilizer codes and quantum error correction[D]. Pasadena:California Institute of Technology, 1997.

[6] GRASSL M,RTTELER M. Quantum block and convolutional codes from self-orthogonal product codes[C]// Proceedings 2005 IEEE International Symposium on Information Theory(ISIT 2005). Adelaide,Australia:IEEE, 2005:1018–1022.

[7] GRASSL M,RTTELER M. Constructions of quantum convolutional codes[J]. IEEE International Symposium on Information Theory, 2007,28(31):816–820.

[8] BENNET C. Quantum cryptography using any two non-orthogonal states[J]. Physical Review Letters, 1992,68(21):3121–3124.

[9] TAN Peiyu,LI Jing. Quantum convolutional codes:practical syndrome decoder[C]// 2012 46th Annual Conference on Information Sciences and Systems(CISS). [S.l.]:IEEE, 2012:1–6.

[10] XING Lijuan,LI Zhuo,WANG Xinmei. A class of quantum stabilizer codes based on classical convolutional codes[J]. Journal of Xidian University, 2008,35(2):277–281.

(下转第 871 页)