

# 多阶幻方卷积码的构造及译码

彭万权<sup>1</sup>, 张承畅<sup>2</sup>, 冯文江<sup>2</sup>, 伍小兵<sup>1</sup>

(1. 重庆工程职业技术学院, 重庆 400037; 2. 重庆大学通信工程学院, 重庆 400044)

**摘要:** 本文结合线性分组码和卷积码, 构造出一类信息组长度和约束度都相对较大的多阶幻方卷积码. 通过定义一种多维矩阵, 进行了状态转移分析, 并借助矩阵运算构建出单一结构但却能并行处理的软判决维特比译码器. 仿真分析表明, 该码类存在大量距离特性优良、高效率的好码, 具有获取香农码的良好预期.

**关键词:** 卷积码; 状态转移; 多维矩阵; 幻方; 维特比算法

**中图分类号:** TN911.22      **文献标识码:** A      **文章编号:** 0372-2112 (2013)01-0123-08

**电子学报 URL:** <http://www.ejournal.org.cn>

**DOI:** 10.3969/j.issn.0372-2112.2013.01.022

## Construction and Decoding of High Order Magic Square Convolutional Codes

PENG Wan-quan<sup>1</sup>, ZHANG Cheng-chang<sup>2</sup>, FENG Wen-jiang<sup>2</sup>, WU Xiao-bing<sup>1</sup>

(1. Chongqing Vocational Institute of Engineering, Chongqing 400037, China;

2. College of Communication Engineering of Chongqing University, Chongqing 400044, China)

**Abstract:** A high order magic square convolution code with relative greater length of information bits and constraint degrees is built based on linear block codes and convolutional codes in the paper. The analysis to the state transition is given through defining a multi-dimensional matrix, and the soft decision viterbi decoder with single structure and the ability of parallel processing is built with the help of matrix computation. The simulation and analysis demonstrate that the code has a large number of good codes with fine distance performance and high efficiency, and it has good expectations to obtain shannon codes.

**Key words:** convolutional codes; state transition; multidimensional matrix; magic square; viterbi algorithm

## 1 引言

卷积码通过不同时刻信息分组的协同编码, 实现了码组之间的信息共享, 加长了码字的记忆时间跨度. 早期的卷积码有: 由鲁滨逊等提出的利用差集三角构造的自正交码和试凑法构造的可正交码<sup>[1]</sup>, 以及由梅西构造的一种可直接从接收序列中快速获取信息序列估值的内快检码<sup>[2]</sup>. 由于卷积码没有成熟的代数理论支撑, 七十年代有学者开始尝试借助计算机搜索好码, 如 Bahl 等创立的基于维特比译码的搜索算法<sup>[3]</sup>, 该算法适用于中小约束度卷积码, 而对于大约束度卷积码, 则往往采用基于序列译码的搜索算法<sup>[4]</sup>. 自上世纪八十年代, 凿孔卷积码、咬尾卷积码和网格编码调制广泛应用于各种数字通信系统<sup>[5-7]</sup>, 其中, 凿孔卷积码通过周期性删除码字, 实现了不同码率; 咬尾卷积码的收尾状态和起始状态相同, 其环状记忆链可避免传输额外符号; 而网格码则巧妙地将编码和调制相结合, 无需拓展带宽即可实现明显的编码增益. 九十年代, 一种反馈型卷积码-递归

系统卷积码伴随 Turbo 码的发现应运而生<sup>[8]</sup>, 这是一种系统码, 却具有非系统码一样好的距离特性. 目前, 卷积低密度校验码<sup>[9-11]</sup>已成为继分组低密度校验码<sup>[12,13]</sup>后的又一研究热点, 当采用置信传播迭代译码时, 可获得高于维特比译码的性价比. 此外, 针对量子通信的误码特性而提出的量子卷积码<sup>[14,15]</sup>也成为目前最好的量子纠错码之一.

信息组长度和约束度的增加都能够引起卷积码记忆时间跨度的增加, 但纵观卷积码的发展, 更多的研究还是着眼于如何增加约束度, 虽也有类似凿孔卷积码或正交卷积码成功增加信息位长度的例子, 但其增幅十分有限, 且往往导致码率变高, 距离特性变差. 此现状的主要原因是: 计算机搜索往往比借助代数理论更易找到好的卷积码, 而试探性搜索时, 单方面思考约束度的增加给编码结构带来的变化会更为容易. 但另一方面, 由于试探数量是随约束度的增长而呈指数增长的, 导致大约束度卷积码的搜索变得十分困难.

基于以上分析, 本文结合  $(2, 1, l)$  卷积码与  $(2k, k)$

线性分组码,构造了一类 $(2k, k, l)$ 新型卷积码,实现了记忆时间跨度的 $k$ 倍增长.之后,通过定义一种多维矩阵,引入了状态转移的矩阵描述,完成了相关证明.第三节通过借助一些矩阵处理模块,设计了一种软判决维特比<sup>[16]</sup>矩阵译码器,阐述了译码过程.仿真部分测试了幻方卷积码的误码特性,比较发现幻方卷积码在纠错能力和运算速度两方面均优于传统卷积码.

## 2 编码及状态转移

### 2.1 基于矩阵运算的编码结构

以下一般按伽罗华域 GF(2) 运算,设 $(2, 1, l)$ 卷积码的生成多项式矩阵为:

$$\mathbf{G}(\mathbf{D}) = \begin{bmatrix} \sum_{j=0}^l g_j \mathbf{D}^j \\ \sum_{j=0}^l h_j \mathbf{D}^j \end{bmatrix} \quad (1)$$

其中 $\mathbf{D}$ 为延迟算子, $g_j, h_j = 1$ 或 $0$ ,按该式构造可得到 $(2k, k, l)$ 卷积码编码器,如图1.设矢量寄存器 $\mathbf{D}^j$ 保存的信息组 $\mathbf{M}(t-j) = [m_0(t-j) \ m_1(t-j) \ m_2(t-j), \dots, m_{k-1}(t-j)]^T$ , $\mathbf{D}^j$ 对 $k$ 位元素同时进行延时.线性组合器1按多项式 $\sum_{j=0}^l g_j \mathbf{M}(t-j)$ 计算,并与 $(2k, k)$ 线性分组码的生成矩阵 $\mathbf{G} = \begin{bmatrix} \mathbf{I} \\ \mathbf{P} \end{bmatrix}$ 相乘,得到编码输出:

$$\mathbf{C}_1 = \begin{bmatrix} \mathbf{I} \\ \mathbf{P} \end{bmatrix} \times \sum_{j=0}^l g_j \mathbf{M}(t-j) = \begin{bmatrix} \sum_{j=0}^l g_j \mathbf{M}(t-j) \\ \mathbf{P} \sum_{j=0}^l g_j \mathbf{M}(t-j) \end{bmatrix} \quad (2)$$

其中 $\mathbf{I}$ 为单位阵, $\mathbf{I}, \mathbf{P}$ 均为 $k \times k$ 矩阵;线性组合器2按多项式 $\sum_{j=0}^l (g_j + h_j) \mathbf{M}(t-j)$ 计算,并送入嵌零部分嵌入 $k$ 个0,得到 $2k \times 1$ 的输出矢量:

$$\mathbf{C}_2 = \begin{bmatrix} \mathbf{0} \\ \sum_{j=0}^l (g_j + h_j) \mathbf{M}(t-j) \end{bmatrix} \quad (3)$$

该式与式(2)相加后可计算得到编码输出:

$$\mathbf{C}_c = \begin{bmatrix} \sum_{j=0}^l g_j \mathbf{M}(t-j) \\ (\mathbf{P} + \mathbf{I}) \sum_{j=0}^l g_j \mathbf{M}(t-j) + \sum_{j=0}^l h_j \mathbf{M}(t-j) \end{bmatrix} \quad (4)$$

图1将线性分组码(以下称内嵌码)嵌入到卷积码的编码结构,每次有 $k \times (l+1)$ 位信息参与到内嵌码的编码,符合短码构造长码的思想,且不会有级联码<sup>[17,18]</sup>

那样的码率损失.当内嵌码为 $(2, 1)$ 偶校验码时, $\mathbf{P} + \mathbf{I} = [0]$ ,式(4)与式(1)等效,可得到 $(2, 1, l)$ 卷积码,可见这种编码结构揭示了短码构造长码的本质.

$(2, 1, l)$ 卷积码的编码结构和内嵌码的距离特性将直接影响 $(2k, k, l)$ 卷积码的纠错能力,表1从文献[19]引入 $l = 1 \sim 5$ 的具有最佳距离特性的卷积码,对应式(1),在表中将多项式系数按三个一组划分后以八进制表示;表2从文献[20]引入 $k = 1 \sim 5$ 的具有最大最小距离的二进制 $(2k, k)$ 双环循环码作为内嵌码,其中 $\mathbf{P}$ 矩阵的每一列同样按三个一组划分后表示成八进制.

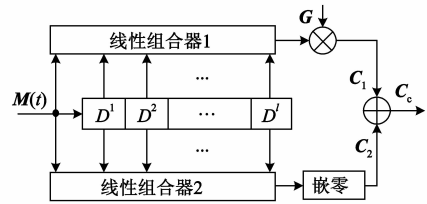


图1  $(2k, k, l)$ 卷积码编码器

表1  $(2, 1, l)$ 卷积码的生成多项式系数

约束度( $l$ )	$g_0 \sim g_l$	$h_0 \sim h_l$
1	3	1
2	7	5
3	17	13
4	31	27
5	75	53

表2  $(2k, k)$ 内嵌码的 $\mathbf{P}$ 矩阵和最小距离

信息组长( $k$ )	$\mathbf{P}$ 矩阵	最小距离
1	[1]	2
2	[1 2]	2
3	[3 5 6]	3
4	[13 15 16 7]	4
5	[7 23 31 34 16]	4

### 2.2 状态转移的矩阵描述

在引入状态转移矩阵描述之前,先按 $l+1$ 维空间的 $x_0, x_1, \dots, x_l$ 方向定义一种 $l+1$ 维矩阵.由图1,状态

$$S_{i_0 i_1 \dots i_l} = \begin{bmatrix} \mathbf{M}(t-1) \\ \mathbf{M}(t-2) \\ \dots \\ \mathbf{M}(t-l) \end{bmatrix} \text{ 与 } S_{i_0 i_1 \dots i_{l-1}} = \begin{bmatrix} \mathbf{M}(t) \\ \mathbf{M}(t-1) \\ \dots \\ \mathbf{M}(t-l+1) \end{bmatrix} \text{ 之间}$$

可发生转移,其中 $i_0, i_1, i_2, \dots, i_l$ 分别为 $\mathbf{M}(t), \mathbf{M}(t-1), \dots, \mathbf{M}(t-l)$ 的十进制,取值范围是 $0 \sim 2^k - 1$ .设 $K = 2^k - 1$ ,可按式(4)分别计算出 $S_{i_0 i_1 \dots i_{l-1}, 0} \sim S_{i_0 i_1 \dots i_{l-1}, K}$ 转移到 $S_{i_0 i_1 \dots i_{l-1}}$ 所对应的码字 $\mathbf{C}_{i_0 i_1 \dots i_{l-1}, 0}^0 \sim \mathbf{C}_{i_0 i_1 \dots i_{l-1}, K}^0$ ,并按 $x_l$ 方向合并,得到如下矩阵:

$$\mathbf{C}_{x_l}^{i_0 i_1 \dots i_{l-1}} = [\mathbf{C}_{i_0 i_1 \dots i_{l-1}, 0}^0 \ \mathbf{C}_{i_0 i_1 \dots i_{l-1}, 1}^0 \ \dots \ \mathbf{C}_{i_0 i_1 \dots i_{l-1}, K}^0]_{x_l} \quad (5)$$

若将每个码字视为一个元素,这是一个 $1 \times 2^k$ 一维矩阵,该式的含义是:当前某个状态可由 $2^k$ 种前一状态的其中之一转移而来.接下来,可将 $i_{l-1} = 0 \sim K$ 带入式

(5), 得到  $2^k$  个  $1 \times 2^k$  一维矩阵, 再按  $x_{l-1}$  方向合并成  $2^k \times 2^k$  的二维矩阵:

$$\mathbf{C}_{x_{l-1}x_l} = [\mathbf{C}_{x_l}^{i_0 i_1 \dots i_{l-2} 0} \mathbf{C}_{x_l}^{i_0 i_1 \dots i_{l-2} 1} \dots \mathbf{C}_{x_l}^{i_0 i_1 \dots i_{l-2} K}]_{x_{l-1}} \quad (6)$$

以此类推, 最后按  $x_0$  方向合并得到  $2^k \times 2^k \times \dots \times 2^k$  的  $l+1$  维状态转移矩阵:

$$\mathbf{C}_{x_0 x_1 x_2 \dots x_l} = [\mathbf{C}_{x_1 x_2 \dots x_l}^0 \mathbf{C}_{x_1 x_2 \dots x_l}^1 \dots \mathbf{C}_{x_1 x_2 \dots x_l}^K]_{x_0} \quad (7)$$

该式囊括了  $(2k, k, l)$  卷积码的所有状态转移, 笔者针对一些  $k, l$ , 通过计算机编程导出式(7), 发现从  $x_0, x_1, \dots, x_l$  不同方向获得的任意一个  $2^k \times 2^k$  二维子矩阵满足以下性质:

**性质 1** 二进制情况下, 任意一行(列)的  $2^k$  个码字的相同位, 0 和 1 等数量;

**性质 2** 二进制情况下, 任意一行(列)的  $2^k$  个码字的码重和等于  $2^k \times k$ ;

**性质 3** 十进制情况下, 矩阵的每行(列)元素之和相等, 符合幻方特性<sup>[21]</sup>, 当编码约束度  $l$  大于 2 时, 可称其为多阶幻方卷积码。

以上三点从不同角度体现了幻方卷积码具有高度均衡的码字重量分布. 不妨定义所有  $k$  位向量  $\mathbf{M}_i = [m_{i0} \ m_{i1} \ m_{i2} \ \dots \ m_{i(k-1)}]^T$  构成包含  $2^k$  个元素的集合  $\Phi(k)$ , 其中  $i$  是  $\mathbf{M}_i$  的十进制, 取值范围  $0 \sim 2^k - 1$ . 在证明这三个性质之前, 先给出相关引理:

**引理 1** 集合  $\Phi(k)$  中的所有元素的相同位, 0 和 1 等数量;

**引理 2** 集合  $\Phi(k)$  中的各元素加上同一个元素后, 映射为集合自身;

**引理 3** 设  $\mathbf{M} \in \Phi(k)$  的码重为  $w(\mathbf{M})$ , 则集合中  $w(\mathbf{M})$  为奇数的数量和偶数的数量各占  $2^{k-1}$ ;

**引理 4** 对集合  $\Phi(k)$  的所有元素进行相同位置的位删除, 定义不改变原有次序的、缩短到  $s$  位的新元素为  $\mathbf{M}'_i = [m'_{i0} \ m'_{i1} \ m'_{i2} \ \dots \ m'_{i(s-1)}]^T$ , 并映射为集合  $\Phi'(k)$ , 其中  $0 < s < k$ , 则  $\Phi'(k)$  包含有  $2^{k-s}$  个  $\Phi(s)$ .

以上四个引理可从 8421 自然二进制码的码字结构获得, 这里不予以证明。

**引理 5** 给定向量  $\mathbf{M} = [m_0 \ m_1 \ m_2 \ \dots \ m_{k-1}]^T \in \Phi(k)$ ,  $\mathbf{P} = [p_{xy}]$  为域  $\text{GF}(2)$  的  $k \times k$  常数矩阵, 其中  $x, y = 0 \sim k-1$ ,  $\mathbf{P}$  的任何一行(列)不全为 0. 当  $\mathbf{M}_i$  遍历 8421 码时, 定义

$$\mathbf{H}_i = \mathbf{P} \times \mathbf{M}_i + \mathbf{M} = [h_{i0} \ h_{i1} \ h_{i2} \ \dots \ h_{i(k-1)}]^T \quad (8)$$

构成集合  $\psi(k)$ , 则  $\psi(k)$  中所有元素的相同位, 0 与 1 等数量。

**证明** 设  $\mathbf{H}'_i = \mathbf{P} \times \mathbf{M}_i = [h'_{i0} \ h'_{i1} \ h'_{i2} \ \dots \ h'_{i(k-1)}]^T$ , 则  $\mathbf{H}_i$  的首位

$$h'_{i0} = [p_{00} \ p_{01} \ \dots \ p_{0(k-1)}]$$

$$\times [m_{i0} \ m_{i1} \ m_{i2} \ \dots \ m_{i(k-1)}]^T = \sum_{y=0}^{k-1} p_{0y} m_{iy} \quad (9)$$

当系数  $p_{0y} = 0$  时, 相当于删除掉  $m_{iy}$ , 由引理 4 对  $\mathbf{M}'_i$  的定义得:

$$h'_{i0} = \sum_{y=0}^{k-1} p_{0y} m_{iy} = \sum_{y=0}^{s-1} m'_{iy} = w(\mathbf{M}'_i) \pmod{2} \quad (10)$$

再联合引理 3 和引理 4;  $i$  遍历  $0 \sim 2^k - 1$  后,  $h'_{i0}$  为 0 或 1 的数量是相等的. 接下来, 考虑到  $\mathbf{M}$  的首位  $m_0$  取值为 0 或 1, 由式(8)得:

$$h_{i0} = h'_{i0} + m_0 = h'_{i0} \text{ 或 } \overline{h'_{i0}} \quad (11)$$

该式表明, 当  $m_0 = 1$  时, 相当于对  $h'_{00} \sim h'_{(2^k-1)0}$  同时取反码, 即 0, 1 相互转换, 且互换的次数相同, 所以依然不会改变 0 和 1 等数量的事实.  $h_{i1} \sim h_{i(k-1)}$  同理可证。

**定理 1** 一阶幻方卷积码满足性质 1;

**证明** 根据式(7), 设一阶幻方卷积码  $(2k, k, 1)$  的二维状态转移矩阵为:

$$\mathbf{C}_{x_0 x_1} = \begin{bmatrix} \mathbf{C}^{00} & \mathbf{C}^{01} & \dots & \mathbf{C}^{0K} \\ \mathbf{C}^{10} & \mathbf{C}^{11} & \dots & \mathbf{C}^{1K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}^{K0} & \mathbf{C}^{K1} & \dots & \mathbf{C}^{KK} \end{bmatrix} \quad (12)$$

查表 1 得到  $(2, 1, 1)$  卷积码的生成多项式矩阵为  $\mathbf{G}(\mathbf{D}) = \begin{bmatrix} 1 + \mathbf{D} \\ 1 \end{bmatrix}$ , 可根据式(4)计算出式(12)的某个元素:

$$\begin{aligned} \mathbf{C}^{i_0 i_1} &= \mathbf{G} \times (\mathbf{M}(t) + \mathbf{M}(t-1)) + \begin{bmatrix} 0 \\ \mathbf{M}(t-1) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{M}(t) + \mathbf{M}(t-1) \\ \mathbf{P} \times (\mathbf{M}(t) + \mathbf{M}(t-1)) + \mathbf{M}(t-1) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{C}_{h'}^{i_0 i_1} \\ \mathbf{C}_{p'}^{i_0 i_1} \end{bmatrix} \end{aligned} \quad (13)$$

其中  $\mathbf{C}^{i_0 i_1}$  对应状态  $\mathbf{S}_{i_1}$  到状态  $\mathbf{S}_{i_0}$  的转移;  $i_0, i_1$  对应  $\mathbf{M}(t), \mathbf{M}(t-1)$  的十进制;  $\mathbf{C}_{h'}^{i_0 i_1}, \mathbf{C}_{p'}^{i_0 i_1}$  对应  $\mathbf{C}^{i_0 i_1}$  的高  $k$  位和低  $k$  位. 首先证明式(12)的某行, 从  $\mathbf{C}^{i_0 i_1}$  的上标可以看出, 其中的  $2^k$  个码字有相同的  $\mathbf{M}(t), \mathbf{M}(t-1)$  则构成了集合  $\Phi(k)$ . 根据引理 2, 这  $2^k$  个码字的高  $k$  位

$$\mathbf{C}_{h'}^{i_0 i_1} = \mathbf{M}(t) + \mathbf{M}(t-1) \quad (14)$$

可映射成集合  $\Phi(k)$ , 由引理 1 可知结论成立. 现在考察低  $k$  位

$$\begin{aligned} \mathbf{C}_{p'}^{i_0 i_1} &= \mathbf{P} \times (\mathbf{M}(t) + \mathbf{M}(t-1)) + \mathbf{M}(t-1) \\ &= \mathbf{P} \times \mathbf{M}(t) + (\mathbf{P} + \mathbf{I})\mathbf{M}(t-1) \end{aligned} \quad (15)$$

前面的分析已经指出, 对于给定的行,  $\mathbf{P}$  和  $\mathbf{M}(t)$  都是确定的, 所以式(15)中的  $(\mathbf{P} + \mathbf{I})$  为常数矩阵,  $\mathbf{P} \times \mathbf{M}(t)$  为常数向量, 又由于  $\mathbf{M}(t-1)$  构成集合  $\Phi(k)$ , 所以该式完全符合式(8)条件, 根据引理 5 可知结论也是成立的. 同理可证“列”的情况。

**定理 2** 一阶幻方卷积码满足性质 2;

**证明** 定理 1 可得出一个更弱的结论,即式(12)的其中一行(列)的  $2^k$  个码字中,0 和 1 必然等数量,所以这  $2^k$  个码字的码重和为:

$$w = \sum_{i_0=0}^{2^k-1} w(\mathbf{C}^{i_0 i_1}) = \sum_{i_1=0}^{2^k-1} w(\mathbf{C}^{i_0 i_1}) = \frac{2k \times 2^k}{2} = k \times 2^k \quad (16)$$

**定理 3** 一阶幻方卷积码满足性质 3;

**证明** 以下计算限于  $\text{GF}(\infty)$ , 设  $\mathbf{C}^{i_0 i_1} = [c_{2^k-2}^{i_0 i_1} \cdots c_{0^0}^{i_0 i_1}]^T$ , 则式(12)的某一列的  $2^k$  个码字的十进制之和为:

$$\mathbf{S} = \sum_{i_0=0}^{2^k-1} \mathbf{C}^{i_0 i_1} = \sum_{i_0=0}^{2^k-1} \sum_{n=0}^{2^k-1} 2^n c_n^{i_0 i_1} = \sum_{n=0}^{2^k-1} \sum_{i_0=0}^{2^k-1} 2^n c_n^{i_0 i_1} \quad (17)$$

由定理 1 的结论可知,当  $i_0$  遍历  $0 \sim 2^k - 1$  时,  $c_n^{i_0 i_1}$  中 0 和 1 的数量均为  $2^{k-1}$ , 则:

$$\mathbf{S} = \sum_{n=0}^{2^k-1} \sum_{i_0=0}^{2^k-1} 2^n c_n^{i_0 i_1} = \sum_{n=0}^{2^k-1} 2^n 2^{k-1} = 2^{k-1} (2^k - 1) \quad (18)$$

同理可证任意一行的  $2^k$  个码字的十进制之和也等于  $2^{k-1} (2^k - 1)$ . 从定理 1 和定理 2 的证明过程可以看出,性质 2、3 实则为性质 1 的两个推论.

**定理 4** 二阶、多阶幻方卷积码满足三个性质.

**证明** 根据式(7), 设  $(2k, k, 2)$  二阶幻方卷积码的三维状态转移矩阵为:

$$\mathbf{C}_{x_0 x_1 x_2} = \begin{bmatrix} \begin{bmatrix} \mathbf{C}^{000} & \mathbf{C}^{001} & \cdots & \mathbf{C}^{00K} \\ \mathbf{C}^{010} & \mathbf{C}^{011} & \cdots & \mathbf{C}^{01K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}^{0K0} & \mathbf{C}^{0K1} & \cdots & \mathbf{C}^{0KK} \end{bmatrix}_{x_1 x_2} & \begin{bmatrix} \mathbf{C}^{100} & \mathbf{C}^{101} & \cdots & \mathbf{C}^{10K} \\ \mathbf{C}^{110} & \mathbf{C}^{111} & \cdots & \mathbf{C}^{11K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}^{1K0} & \mathbf{C}^{1K1} & \cdots & \mathbf{C}^{1KK} \end{bmatrix}_{x_1 x_2} \\ \cdots & \begin{bmatrix} \mathbf{C}^{K00} & \mathbf{C}^{K01} & \cdots & \mathbf{C}^{K0K} \\ \mathbf{C}^{K10} & \mathbf{C}^{K11} & \cdots & \mathbf{C}^{K1K} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}^{KK0} & \mathbf{C}^{KK1} & \cdots & \mathbf{C}^{KKK} \end{bmatrix}_{x_1 x_2} \end{bmatrix}_{x_0} \quad (19)$$

其中,  $x_1, x_2$  两个方向确定了  $2^k$  个  $2^k \times 2^k$  二维子矩阵, 并按  $x_0$  方向合并成一个三维矩阵. 查表 1 可得  $(2, 1, 2)$

卷积码的生成多项式矩阵为  $\mathbf{G}(\mathbf{D}) = \begin{bmatrix} 1 + \mathbf{D} + \mathbf{D}^2 \\ 1 + \mathbf{D}^2 \end{bmatrix}$ , 根据式(4)可计算出式(19)的某个元素:

$$\mathbf{C}^{i_0 i_1 i_2} = \begin{bmatrix} \mathbf{M}(t) + \mathbf{M}(t-1) + \mathbf{M}(t-2) \\ \mathbf{P} \times (\mathbf{M}(t) + \mathbf{M}(t-1) + \mathbf{M}(t-2)) + \mathbf{M}(t-1) \end{bmatrix} \\ = \begin{bmatrix} \mathbf{C}_h^{i_0 i_1 i_2} \\ \mathbf{C}_p^{i_0 i_1 i_2} \end{bmatrix} \quad (20)$$

其中  $\mathbf{C}^{i_0 i_1 i_2}$  对应状态  $\mathbf{S}_{i_1 i_2}$  到状态  $\mathbf{S}_{i_0 i_1}$  的转移;  $i_0, i_1, i_2$  对应  $\mathbf{M}(t), \mathbf{M}(t-1), \mathbf{M}(t-2)$  的十进制;  $\mathbf{C}_h^{i_0 i_1 i_2}, \mathbf{C}_p^{i_0 i_1 i_2}$  对应  $\mathbf{C}^{i_0 i_1 i_2}$  的高  $k$  位和低  $k$  位. 首先考察式(19)中某给定子

矩阵的某一行, 其中的  $2^k$  个码字有相同的  $\mathbf{M}(t)$  和  $\mathbf{M}(t-1), \mathbf{M}(t-2)$  则构成了集合  $\Phi(k)$ , 所以由引理 2 可知高  $k$  位  $\mathbf{C}_h^{i_0 i_1 i_2} = \mathbf{M}(t) + \mathbf{M}(t-1) + \mathbf{M}(t-2)$  也会构成集合  $\Phi(k)$ , 性质 1 成立. 对于低  $k$  位, 可整理得到:

$$\mathbf{C}_p^{i_0 i_1 i_2} = \mathbf{P} \times \mathbf{M}(t) + (\mathbf{P} + \mathbf{I}) \times \mathbf{M}(t-1) + \mathbf{P} \times \mathbf{M}(t-2) \quad (21)$$

类似于定理 1 的分析, 该式中  $\mathbf{P}$  为常数矩阵,  $\mathbf{P} \times \mathbf{M}(t) + (\mathbf{P} + \mathbf{I}) \times \mathbf{M}(t-1)$  为固定矢量,  $\mathbf{M}(t-2)$  则构成集合  $\Phi(k)$ , 符合式(8)条件, 可见性质 1 同样成立. 事实上, 对于其他方向的子矩阵的某一行或列的  $2^k$  个码字,  $\mathbf{M}(t), \mathbf{M}(t-1)$  和  $\mathbf{M}(t-2)$  之中总是有一个且只有一个不同, 并构成了集合  $\Phi(k)$ , 所以同理可证. 性质 1 得证, 性质 2、3 的证明就非常容易了.

对于多阶幻方卷积码的状态转移子矩阵的某一行或列,  $\mathbf{M}(t), \mathbf{M}(t-1), \cdots, \mathbf{M}(t-l)$  之中仍然仅仅只有一个不同, 而且正好构成集合  $\Phi(k)$ , 所以最终都可以归结为类似于一、二阶幻方卷积码的证明.

### 3 viterbi 矩阵译码器

各状态节点以 8421 自然二进制码排序, 依照式(7)将相邻时间的节点连通, 并按式(4)计算对应码字, 若干状态节点首尾相连便可构成  $(2k, k, l)$  卷积码的栅格图. 虽然幻方卷积码与常规卷积码的栅格图有所区别, 但同样可采用 Viterbi 译码. 由状态转移描述可知, 栅格图有  $2^{lk}$  种状态, 前一时刻状态的高  $k(l-1)$  位与当前时刻状态的低  $k(l-1)$  位相同时对应的节点连通, 因此有  $2^k$  条分支路径汇接于同一个节点, 一共可形成  $2^{(l+1)k}$  条分支. 称单个分支的似然度为分支度量, 一条连通路径的分支度量累加值为路径度量. 在 Viterbi 译码器的工程设计中一般采用并行结构, 以牺牲结构复杂度获得快速译码<sup>[22,23]</sup>, 当前时刻, 幻方卷积码的并行译码需对汇接于某个状态节点的  $2^k$  条分支分别计算度量值, 并与前一时刻的路径度量相加, 得到  $2^k$  个新的路径度量, 然后从中挑选出具有最大度量值的路径作为幸存路径. 由于一共有  $2^{lk}$  个这样的“乘-加比选”并行操作, 所以需要  $2^{lk}$  个寄存单元保存幸存路径度量, 以提供给下一次累加计算使用. 为避免溢出, 应定期对路径度量实施衰减. 另外, 并行译码还需提供  $2^{lk}$  个寄存器组, 用以保存幸存路径, 寄存器组包含的寄存单元越多, 保存的幸存路径越长, 译码结果越接近最优.

#### 3.1 乘-加比选

借助矩阵运算可设计一种运算效率更高、分析和设计更方便的单一结构 Viterbi 并行译码器, 如图 2 所示. 为方便分支度量的计算, 可按  $x_0, x_1, \cdots, x_l$  方向逐一提取式(7)中的元素, 并整形为  $2^{(l+1)k} \times 1$  一维矩阵:

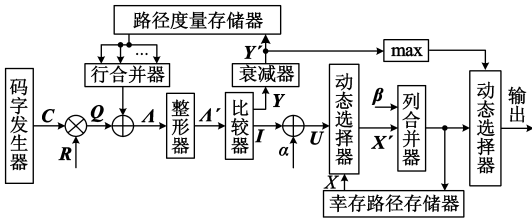


图2 译码器

$$C = [C^{00\cdots 00} \ C^{00\cdots 01} \ \cdots \ C^{00\cdots 0K} \ C^{00\cdots 10} \ \cdots \ C^{KK\cdots KK}]^T \quad (22)$$

其中,上标  $i_0 i_1 \cdots i_{l-1} i_l$  可视为  $l+1$  位  $2^k$  进制排序;  $C^{00\cdots 00} \sim C^{00\cdots 0K}$  对应汇接于状态节点  $S_{00\cdots 0}$  的  $2^k$  条分支,后面以此类推。如果视码字为  $1 \times 2k$  矢量而不是一个元素,可将式(22)变换为  $2^{(l+1)k} \times 2k$  矩阵:

$$C = \begin{bmatrix} c_0^{00\cdots 00} & c_0^{00\cdots 01} & \cdots & c_0^{00\cdots 0K} & c_0^{00\cdots 10} & \cdots & c_0^{KK\cdots KK} \\ c_1^{00\cdots 00} & c_1^{00\cdots 01} & \cdots & c_1^{00\cdots 0K} & c_1^{00\cdots 10} & \cdots & c_1^{KK\cdots KK} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ c_{2k-1}^{00\cdots 00} & c_{2k-1}^{00\cdots 01} & \cdots & c_{2k-1}^{00\cdots 0K} & c_{2k-1}^{00\cdots 10} & \cdots & c_{2k-1}^{KK\cdots KK} \end{bmatrix} \quad (23)$$

该式为常数矩阵,也即是图2中码字发生器的输出,可借助矩阵运算进行编码,再转换成双极性码后存入工作空间,仿真时直接取用。

当前时刻,设接收到带有噪声的软信息矢量  $R(t) = [r_0(t) \ r_1(t) \ \cdots \ r_{2k-1}(t)]^T$ ,矩阵乘法器完成“乘”操作,输出:

$$Q(t) = C \times R(t) = [\rho^{00\cdots 00}(t) \ \rho^{00\cdots 01}(t) \ \cdots \ \rho^{00\cdots 0K}(t) \ \rho^{00\cdots 10}(t) \ \cdots \ \rho^{KK\cdots KK}(t)]^T \quad (24)$$

$$\text{其中} \quad \rho^{i_0 i_1 \cdots i_{l-1} i_l}(t) = \sum_{n=0}^{2^k-1} c_n^{i_0 i_1 \cdots i_{l-1} i_l} r_n(t) \quad (25)$$

为  $C^{i_0 i_1 \cdots i_{l-1} i_l}$  和  $R(t)$  的相关运算,由文献[17]可知,相关运算与欧氏距离运算等价,进而与最大后验概率等价。可见  $C$  和  $R(t)$  的矩阵乘法可一次性完成所有  $2^{(l+1)k}$  条分支的似然度计算,计算结果符合最大后验概率准则。设矩阵加法器的当前输出为:

$$\Lambda(t) = [\lambda^{00\cdots 00}(t) \ \lambda^{00\cdots 01}(t) \ \cdots \ \lambda^{00\cdots 0K}(t) \ \lambda^{00\cdots 10}(t) \ \cdots \ \lambda^{KK\cdots KK}(t)]^T \quad (26)$$

和  $Q(t)$  一样,这是一个  $2^{(l+1)k} \times 1$  矩阵,为了便于执行“比较”操作,整形模块按顺序依次从该式中提取  $2^k$  个元素作为新矩阵的一行,将其整形为  $2^k \times 2^k$  矩阵:

$$\Lambda'(t) = \begin{bmatrix} \lambda^{00\cdots 00}(t) & \lambda^{00\cdots 01}(t) & \cdots & \lambda^{00\cdots 0K}(t) \\ \lambda^{00\cdots 10}(t) & \lambda^{00\cdots 11}(t) & \cdots & \lambda^{00\cdots 1K}(t) \\ \vdots & \vdots & \ddots & \vdots \\ \lambda^{00\cdots K0}(t) & \lambda^{00\cdots K1}(t) & \cdots & \lambda^{00\cdots KK}(t) \\ \vdots & \vdots & \ddots & \vdots \\ \lambda^{KK\cdots K0}(t) & \lambda^{KK\cdots K1}(t) & \cdots & \lambda^{KK\cdots KK}(t) \end{bmatrix} \quad (27)$$

其中每一列元素上标的高  $l$  位按  $2^k$  进制排序,比较器模块对每一行的  $2^k$  个元素进行比较后输出最大值,得到:

$$Y(t) = \begin{bmatrix} \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 0j}(t) \} \\ \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 1j}(t) \} \\ \vdots \\ \max_{0 \leq j \leq K} \{ \lambda^{00\cdots Kj}(t) \} \\ \vdots \\ \max_{0 \leq j \leq K} \{ \lambda^{KK\cdots Kj}(t) \} \end{bmatrix} \quad (28)$$

以上过程的作用是:通过比较进入状态节点  $S_{i_0 i_1 \cdots i_{l-1}}$  的  $2^k$  条路径度量,找到该节点所对应的最大似然路径。为了避免  $Y(t)$  的元素逐渐变大,衰减器从该式中找出最小值  $\lambda_{\min}$  并相减,输出:

$$Y'(t) = \begin{bmatrix} \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 0j}(t) \} - \lambda_{\min} \\ \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 1j}(t) \} - \lambda_{\min} \\ \vdots \\ \max_{0 \leq j \leq K} \{ \lambda^{00\cdots Kj}(t) \} - \lambda_{\min} \\ \vdots \\ \max_{0 \leq j \leq K} \{ \lambda^{KK\cdots Kj}(t) \} - \lambda_{\min} \end{bmatrix} \quad (29)$$

这里对  $Y(t)$  中所有元素都做了同等衰减,并不会破坏最大似然准则,且可使路径度量始终稳定在较小值。

$Y'(t)$  被送到路径度量存储器保存,由于每一个旧状态会指向  $2^k$  个新状态,对应路径度量在下次累加中将被调用  $2^k$  次,因此需要对  $Y'(t)$  进行  $2^k$  重矩阵的行合并,则下一时刻累加器的输出为:

$$\Lambda(t+1) = Q(t+1) + \begin{bmatrix} Y'(t) \\ Y'(t) \\ \vdots \\ Y'(t) \end{bmatrix} = \begin{bmatrix} \rho^{00\cdots 0}(t+1) + \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 0j}(t) \} - \lambda_{\min} \\ \rho^{00\cdots 1}(t+1) + \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 1j}(t) \} - \lambda_{\min} \\ \vdots \\ \rho^{KK\cdots K}(t+1) + \max_{0 \leq j \leq K} \{ \lambda^{KK\cdots Kj}(t) \} - \lambda_{\min} \\ \rho^{00\cdots 0}(t+1) + \max_{0 \leq j \leq K} \{ \lambda^{00\cdots 0j}(t) \} - \lambda_{\min} \\ \vdots \\ \rho^{KK\cdots K}(t+1) + \max_{0 \leq j \leq K} \{ \lambda^{KK\cdots Kj}(t) \} - \lambda_{\min} \\ \vdots \\ \rho^{KK\cdots K}(t+1) + \max_{0 \leq j \leq K} \{ \lambda^{KK\cdots Kj}(t) \} - \lambda_{\min} \end{bmatrix} \quad (30)$$

累加器、整形器、比较器、衰减器、路径度量存储器以及合并器构成的环路完成了加比选以及路径度量的衰减和更新等操作。

### 3.2 幸存路径的保存和更新

每一个当前状态节点都将与式(28)选中的上一时

刻状态节点所保留的那条幸存路径连通,并删除该路径的最旧分支,完成幸存路径的更新,此过程由图 2 中的动态选择器和列合并器完成.译码器需要提供与当前时刻状态节点  $S_{i_0 i_1 \dots i_{l-1}}$  顺序一致的原始信息组,设  $2^{kl}$  组信息矢量构成如下矩阵:

$$\beta = [\beta_{00\dots 00} \beta_{00\dots 01} \dots \beta_{00\dots 0K} \beta_{00\dots 10} \dots \beta_{KK\dots KK}]^T \quad (31)$$

其中  $\beta_{i_0 i_1 \dots i_{l-1}} = (i_0)_B$  为  $1 \times k$  二进制信息,设定当前时刻幸存路径存储器的内容为:

$$X(t) = \begin{bmatrix} \beta_{00\dots 00} X_{00\dots 00}(t-1) & X_{00\dots 00}(t-2) & \dots & X_{00\dots 00}(t-\tau) \\ \beta_{00\dots 01} X_{00\dots 01}(t-1) & X_{00\dots 01}(t-2) & \dots & X_{00\dots 01}(t-\tau) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{00\dots 0K} X_{00\dots 0K}(t-1) & X_{00\dots 0K}(t-2) & \dots & X_{00\dots 0K}(t-\tau) \\ \beta_{00\dots 10} X_{00\dots 10}(t-1) & X_{00\dots 10}(t-2) & \dots & X_{00\dots 10}(t-\tau) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{KK\dots KK} X_{KK\dots KK}(t-1) & X_{KK\dots KK}(t-2) & \dots & X_{KK\dots KK}(t-\tau) \end{bmatrix} \quad (32)$$

这是  $2^{lk} \times k(\tau+1)$  矩阵,  $\tau$  为存储深度.该矩阵的最后  $k$  列为最旧信息组.

动态选择器的作用是找到每一个新分支所对应的上一时刻幸存路径,设式(28)中最大值的索引矩阵为:

$$I = [i_{00\dots 00} i_{00\dots 01} \dots i_{00\dots 0K} i_{00\dots 10} \dots i_{KK\dots KK}]^T \quad (33)$$

其中各元素取值范围是  $0 \sim 2^k - 1$ .事实上,该式是式(28)的每一行的列索引,需转换成式(32)的行索引才能为其所用,令:

$$\alpha_0 = \alpha_1 = \dots = \alpha_K = 2^k [0 \ 1 \ 2 \ \dots \ 2^{(l-1)k} - 1] \quad (34)$$

可合并得到修正矢量:

$$\alpha = [\alpha_0 \ \alpha_1 \ \alpha_2 \ \dots \ \alpha_K]^T \quad (35)$$

该式与式(33)相加后修正为新的行索引矩阵:

$$U = [u_{00\dots 00} u_{00\dots 01} \dots u_{00\dots 0K} u_{00\dots 10} \dots u_{KK\dots KK}]^T \quad (36)$$

当下一时刻到来时,动态选择器从幸存路径存储器获取  $X(t)$ ,并根据  $U$  对其逐行选择,完成行的重新排序:

$$X'(t) = \begin{bmatrix} \beta_{u_{00\dots 00}} X_{u_{00\dots 00}}(t-1) & X_{u_{00\dots 00}}(t-2) & \dots & X_{u_{00\dots 00}}(t-\tau) \\ \beta_{u_{00\dots 01}} X_{u_{00\dots 01}}(t-1) & X_{u_{00\dots 01}}(t-2) & \dots & X_{u_{00\dots 01}}(t-\tau) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{u_{00\dots 0K}} X_{u_{00\dots 0K}}(t-1) & X_{u_{00\dots 0K}}(t-2) & \dots & X_{u_{00\dots 0K}}(t-\tau) \\ \beta_{u_{00\dots 10}} X_{u_{00\dots 10}}(t-1) & X_{u_{00\dots 10}}(t-2) & \dots & X_{u_{00\dots 10}}(t-\tau) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{u_{KK\dots KK}} X_{u_{KK\dots KK}}(t-1) & X_{u_{KK\dots KK}}(t-2) & \dots & X_{u_{KK\dots KK}}(t-\tau) \end{bmatrix} \quad (37)$$

再与  $\beta$  执行列合并,并自动删除最后  $k$  列,得到新的幸存路径矩阵:

$$X'(t) = \begin{bmatrix} \beta_{00\dots 00} & \beta_{u_{00\dots 00}} & X_{u_{00\dots 00}}(t-1) & \dots & X_{u_{00\dots 00}}(t-\tau+1) \\ \beta_{00\dots 01} & \beta_{u_{00\dots 01}} & X_{u_{00\dots 01}}(t-1) & \dots & X_{u_{00\dots 01}}(t-\tau+1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \beta_{00\dots 0K} & \beta_{u_{00\dots 0K}} & X_{u_{00\dots 0K}}(t-1) & \dots & X_{u_{00\dots 0K}}(t-\tau+1) \\ \beta_{00\dots 10} & \beta_{u_{00\dots 10}} & X_{u_{00\dots 10}}(t-1) & \dots & X_{u_{00\dots 10}}(t-\tau+1) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \beta_{KK\dots KK} & \beta_{u_{KK\dots KK}} & X_{u_{KK\dots KK}}(t-1) & \dots & X_{u_{KK\dots KK}}(t-\tau+1) \end{bmatrix} \quad (38)$$

随着存储深度的增加,某些幸存路径将汇接于同一节点,体现在式(38)靠右的列元素逐渐趋于相同,图 3 是(6,3,3)幻方卷积码对该式的一个仿真截图,从中可以清楚的看到这一点.

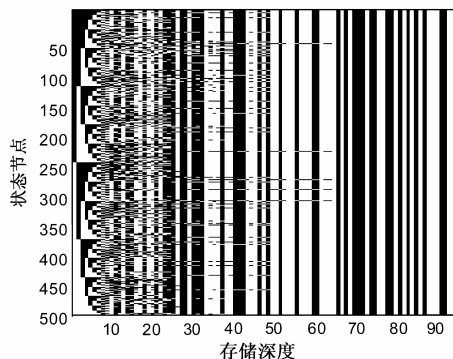


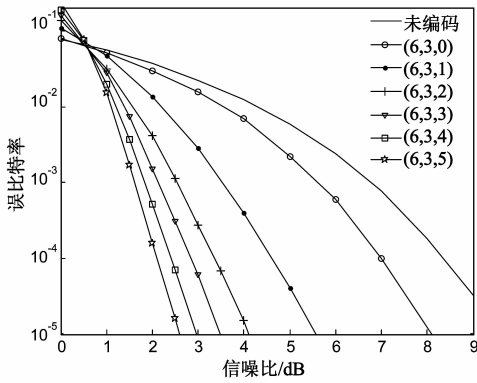
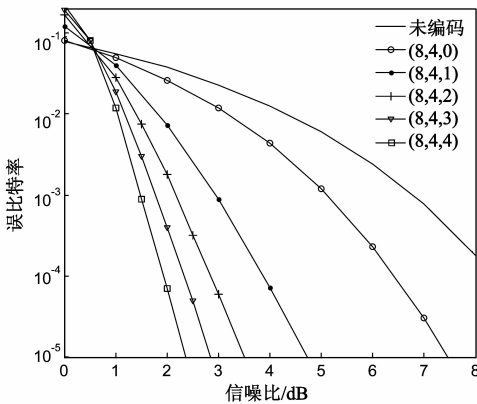
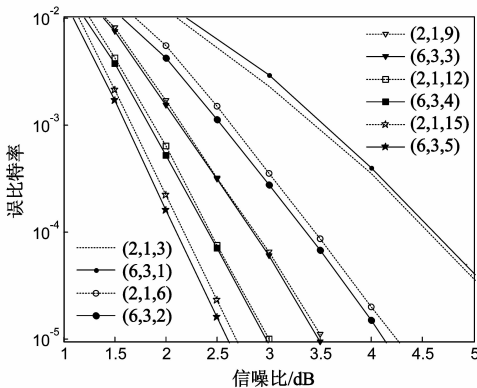
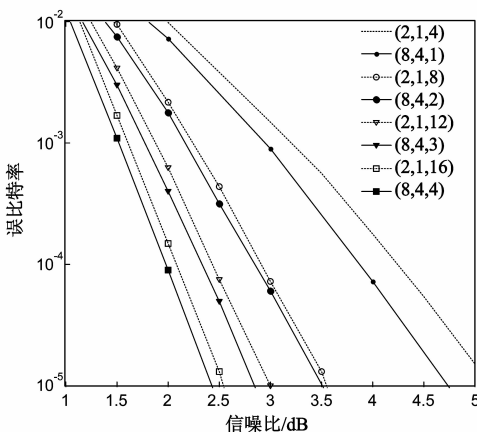
图3 (6,3,3)卷积码路径存储矩阵图

$\max$  模块的作用是输出  $Y'(t)$  中最大元素的索引,动态选择器根据该索引在式(38)的最后  $k$  列中选择对应信息码元作为译码输出,此操作可有效降低存储深度  $\tau$ ,因为即使最后一个节点幸存路径没有完全汇合,译码器仍可选择最好的一条路径作为译码输出.

## 4 仿真分析

仿真置于 BPSK 调制及高斯信道环境.以表 1 和表 2 所列的码型构造  $(2k, k, l)$  幻方卷积码,乘-加比选环路选用双精度数据类型,幸存路径的存储深度  $\tau = 10kl$ ,视具体情况在 2000 ~ 10000 个错误时终止仿真.鉴于(6,3)和(8,4)两款内嵌码的距离特性具有较高性价比,重点对(6,3,  $l$ )和(8,4,  $l$ )的误码特性进行了仿真,结果如图 4 和图 5.由该图可看出,随着  $l$  的增长,误码曲线有一个稳定的增益增量,当  $l = 3 \sim 5$  时便能在  $10^{-5}$  处进入到 2 ~ 3dB 信噪比区域,这一点充分表明,当  $l$  足够大时,幻方卷积码具有存在香农码的良好预期.

为了进一步验证幻方卷积码的优势,可将其与目前为止搜索到的具有最优距离特性的  $(2, 1, l)$  常规卷积码<sup>[19]</sup>进行纠错能力和译码速度的比较,设二者的约束度分别为  $l_1$  和  $l_2$ ,当  $k \times l_1 = l_2$  时状态数相等,译码复杂度相当,具有较强的可比性,图 6 和图 7 分别给出了

图4  $(6,3,l)$  误比特率与信噪比关系曲线图5  $(8,4,l)$  误比特率与信噪比关系曲线图6  $(6,3,l)$  与常规卷积码的比较图7  $(8,4,l)$  与常规卷积码的比较

$(6,3,l)$ 、 $(8,4,l)$ 与常规卷积码的误码特性对比,从这两个图可以看出,除 $(6,3,1)$ 外,其余幻方卷积码的纠错能力都不同程度优于常规卷积码.计算速度方面,表3给出了图6和图7所对应码型的测试结果,其中第二列和第三列分别为幻方卷积码和常规卷积码在程序运行10分钟后计算完成的信息比特数,从该表可以看出,幻方卷积码的译码速度具有明显的优势.

表3  $(2k, k, l_1)$ 和 $(2, 1, l_2)$ 卷积码的计算速度比较

$k, l_1, l_2$	$(2k, k, l_1)$	$(2, 1, l_2)$
3, 1, 3	$1.7 \times 10^7$	$4.9 \times 10^6$
4, 1, 4	$1.6 \times 10^7$	$4.2 \times 10^6$
3, 2, 6	$8.4 \times 10^6$	$2.9 \times 10^6$
4, 2, 8	$4.4 \times 10^6$	$1.2 \times 10^6$
3, 3, 9	$2.1 \times 10^6$	$3.9 \times 10^5$
4, 3, 12	$2.4 \times 10^5$	$3.8 \times 10^4$
3, 4, 12	$2.2 \times 10^5$	$3.8 \times 10^4$
3, 5, 15	$3.3 \times 10^4$	$6.2 \times 10^3$
4, 4, 16	$9.5 \times 10^3$	$2.7 \times 10^3$

## 5 总结

用短码构造长码一直是纠错编码的努力方向,其中包含了分而治之的思想,级联码和常规卷积码就是其中的范例,如果把前者视为面状记忆链、后者视为半无限长的线状记忆链,那么幻方卷积码则实现了半无限长的带状记忆链.图1充分展现了如何借助常规卷积码向线性分组码的编码过程注入记忆,为长码的构造提供了一种新的思路.在分析状态转移时,引入矩阵描述的优势主要体现在三方面:(1)卷积码的 $k$ 和 $l$ 较大时,图解法难以完成状态转移描述,但矩阵法却能很好的解决;(2)可以据此分析幻方卷积码的距离特性;(3)Viterbi矩阵译码器的设计实际上是以状态转移矩阵为基础才得以完成的.对于不同的 $k$ 和 $l$ ,除了某些模块的内部参数外,Viterbi矩阵译码器的外观基本一样,非常有利于分析和设计.笔者认为研究幻方卷积码的主要意义在于,这种构造策略可充分调用分组码和卷积码已有的研究成果,通过信息组长度和约束度的双重增长,有效实现记忆时间跨度的成倍增长,从编码角度揭示了幻方卷积码具有存在香农码的可能性.

## 参考文献

- [1] Robinson J, Bernstein A. A class of binary recurrent codes with limited error propagation [J]. IEEE Transactions on Information Theory, 1967, 13(1): 106 - 113.
- [2] Massey J, Costello D. Nonsystematic convolutional codes for sequential decoding in space applications [J]. IEEE Transactions on Communication Technology, 1971, 19(5): 806 - 813.
- [3] Bahl L, et al. An efficient algorithm for computing free distance

- [J]. IEEE Transactions on Informational Theory, 1972, 18(3): 437 - 439.
- [4] Johannesson R, Stahl P. New rate  $1/2$ ,  $1/3$ , and  $1/4$  binary convolutional encoders with an optimum distance profile[J]. IEEE Transactions on Informational Theory, 1999, 45(5): 1653 - 1658.
- [5] Cain J, Clark G, Geist J. Punctured convolutional codes of rate  $(n-1)/n$  and simplified maximum likelihood decoding[J]. IEEE Transactions on Informational Theory, 1979, 25(1): 97 - 100.
- [6] Anderson J B, Hladik S M. Tail-biting MAP decoders[J]. IEEE Journal on Selected Areas Communications, 1998, 16(2): 297 - 302.
- [7] Ungerboeck G. Channel coding with multilevel/phase signals [J]. IEEE Transactions on Informational Theory, 1982, 28(1): 55 - 67.
- [8] Berrou C, Glavieux A. Near optimum error correcting coding and decoding: turbo-codes[J]. IEEE Transactions on Communication Theory, 1996, 44(10): 1261 - 1271.
- [9] Tanner R M, et al. LDPC Block and convolutional codes based on circulant matrices[J]. IEEE Transactions on Informational Theory, 2004, 50(12): 2966 - 2984.
- [10] Katsiotis A, et al. New constructions of high-performance low-complexity convolutional codes [J]. IEEE Transactions on Communication Theory, 2010, 58(7): 1950 - 1961.
- [11] Pusane A E, et al. Deriving good LDPC convolutional codes from LDPC block codes[J]. IEEE Transactions on Informational Theory, 2011, 57(2): 835 - 857.
- [12] 彭立, 朱光喜. QC-LDPC 码的置换矩阵循环移位次数设计[J]. 电子学报, 2010, 38(4): 786 - 790.  
PENG Li, ZHU Guang-xi. Shift value design of permutation matrices for QC-LDPC codes [J]. Acta Electronica Sinica, 2010, 38(4): 786 - 790. (in Chinese)
- [13] Abu-Surra S, Divsalar D, Ryan W E. Enumerators for photograph-based ensembles of LDPC and generalized LDPC codes [J]. IEEE Transactions on Informational Theory, 2011, 57(2): 858 - 886.
- [14] G David Forney, Markus Grassl, Saikat Guha. Convolutional and tail-biting quantum error-correcting codes [J]. IEEE Transactions on Informational Theory, 2007, 53(3): 865 - 880.
- [15] Peiyu Tan, Jing Li. Efficient quantum stabilizer codes: LDPC and LDPC-convolutional constructions [J]. IEEE Transactions on Informational Theory, 2010, 56(1): 476 - 491.
- [16] Jie Luo. On low-complexity maximum-likelihood decoding of convolutional codes [J]. IEEE Transactions on Informational Theory, 2008, 54(12): 5756 - 5760.
- [17] 彭万权, 冯文江. 乘积码基于相关运算的迭代译码 [J]. 电路与系统学报, 2006, 11(4): 26 - 30.  
Peng Wanquan, Feng Wenjiang. A new iterative decoding for product codes with correlation operation [J]. Journal of Circuits and Systems, 2006, 11(4): 26 - 30. (in Chinese)
- [18] Huebner A, Kliever J, Costello D J. Double serially concatenated convolutional codes with jointly designed S-type permutores [J]. IEEE Transactions on Informational Theory, 2009, 55(12): 5811 - 5821.
- [19] Shu Lin, Daniel J, Costello. Error Control Coding: Fundamentals and Applications [M]. Pearson Education, 2004. 515 - 558.
- [20] 王新梅, 肖国镇. 纠错码 - 原理与方法 [M]. 西安: 西安电子科技大学出版社, 2001. 52 - 161.  
Wang Xinmei, Xiao Guozhen. Error Correcting Codes-Principle and Method [M]. Xi'an: Xidian University Press, 2001. 52 - 161. (in Chinese)
- [21] 金丕龄. 幻方的智慧 [M]. 上海: 上海交通大学出版社, 2010.  
Jin Piling. Wisdom of Magic Square [M]. Shanghai: Shanghai Jiao Tong University Press, 2010. (in Chinese)
- [22] Akopian D, et al. Multistage interconnection networks for parallel Viterbi decoders [J]. IEEE Transactions on Communication Theory, 2003, 51(9): 1536 - 1545.
- [23] Jie Jin, Chi-ying Tsui. Low-power limited-search parallel state Viterbi decoder implementation based on scarce state transition [J]. IEEE Transactions on VLSI Systems, 2007, 15(10): 1172 - 1176.

#### 作者简介



彭万权 男, 1974 年生于重庆垫江. 2005 年在重庆大学通信工程学院获得工学硕士学位, 现为重庆工程职业技术学院副教授. 主要从事纠错码的算法研究工作.

E-mail: 408502@163.com



张承畅 男, 1975 年生于湖北恩施, 分别于 2005 年、2011 年在重庆大学获得工学硕士学位、工学博士学位, 现为重庆大学副教授. 主要从事纠错码及信号处理方面的研究工作.