文章编号:1001-506X(2011)10-2331-07

多进制准循环 LDPC 码满秩校验矩阵构造及系统编码

刘 冰[1,3],张用宇[1],吴东伟[1],陶 伟[2,3]
(1. 中国人民解放军 91469 部队,北京 100841;2. 海军装备研究院,北京 100161;
3. 海军工程大学电子工程学院,湖北 武汉 430033)

摘 要:提出了一种多进制准循环低密度奇偶校验(low-density parity-check,LDPC)码满秩校验矩阵的构造方法。该方法基于循环置换方阵,利用随机掩蔽的方法构造出满秩校验矩阵,从而得到具有循环阵形式的系统生成矩阵,并设计出具有线性复杂度的串行和并行多进制 LDPC 码编码器。仿真结果表明,由此构造出的规则和非规则多进制准循环 LDPC 码相比于掩蔽前的码字取得了更为优越的误码和收敛性能。

关键词:多进制低密度奇偶校验码;准循环;满秩校验矩阵;编码
中图分类号:TN 911.22 **文献标志码**:A **DOI**:10.3969/j.issn.1001-506X.2011.10.37

Construction of full rank parity check matrix and systematic encoding for nonbinary quasi-cyclic LDPC codes

LIU Bing[1,3], ZHANG Yong-yu[1], WU Dong-wei[1], TAO Wei[2,3]
(*1. Unit 91469 of PLA, Beijing 100841, China;*
2. Naval Academy of Armament, Beijing 100161, China;
3. College of Electronic Engineering, Naval University of Engineering, Wuhan 430033, China)

Abstract: The method for constructing a full rank parity-check matrix of nonbinary quasi-cyclic (QC) low-density parity-check (LDPC) codes is proposed. The method is based on the circulant permutation matrices, and the full rank parity-check matrices are obtained using the proposed random masking method. The systematic generator matrix of a QC LDPC code is given in a circulant form from its parity-check matrix. Correspondingly, the serial and parallel efficient encoders with linear complexity are devised for the nonbinary full rank parity-check matrices. Experimental results show that regular and irregular nonbinary QC LDPC codes constructed by the random masking method achieve a better error performance and convergence than those without masking.

Keywords: nonbinary low-density parity-check (LDPC) code; quasi-cyclic; full rank parity-check matrix; encoding

0 引 言

多进制低密度奇偶校验(low-density parity-check,LDPC)码早在 1963 年由文献[1]发明并研究,但由于当时计算机水平的限制和级联码的优良性能,并没有引起研究学者们的足够重视。直到 1998 年,文献[2]重新发现多进制 LDPC 码并提出了有效的 QSPA(*q*-ary sum-product algorithm)译码算法。制约多进制 LDPC 码走向实用化的瓶颈在于编码和译码复杂过高,寻求一种线性化编码器和低复杂度译码器是目前的迫切需求。基于快速傅里叶变换(fast Fourier transform,FFT)的 QSPA 译码算法 FFT-QSPA 的提出[3]在保证性能不变的情况下有效地解决了多进制 LDPC 码译码复杂度过高的弊端。而准循环(qusic-cyclic, QC)或循环结构的 LDPC 码可解决编码器线性复杂度问题[4],有利于工程硬件实现。

与二进制 LDPC 码校验矩阵的构造不同,多进制LDPC 码校验矩阵的构造需要确定非零值的位置和取值两个参数,只有这两个参数同时具有结构化的特性时,才认为矩阵是结构化的。据此,多进制 LDPC 码校验矩阵的构造方法大致可分为三大类:随机化构造法[5]、结构化构造法[6-7]、混合构造法。随机化校验矩阵缺少有规律的结构,致使多进制 LDPC 码编码过程变得复杂,且需要较大的存储空间来存储校验矩阵,这些缺陷都阻碍了随机化构造的应用。结构化构造方法在工程应用中发挥着重要的作用,尤其是 QC 结构,其仅需存储一个元素就可取代一个阵列,存储空间大大减小。与实现系统编码密切相关的参数是校验矩阵的

收稿日期:2010-12-20; **修回日期**:2011-05-23。
基金项目:国家高技术研究发展计划(863 计划)(2009AAJ128,2010AA7010422);中国博士后科学基金(200902671)资助课题
作者简介:刘冰(1984-),男,工程师,博士,主要研究方向为差错控制编码、数字通信信号处理。E-mail:liubing5275093@hotmail.com

秩，非满秩校验矩阵构造的 LDPC 码编出的码字为非系统码，非系统部分完成信息位的恢复需要额外的运算。对于基于非满秩校验矩阵的 LDPC 码编码可直接采用非系统编码方法，但若要达到系统编码的目的，可牺牲码率，对非系统部分对应的信息不进行编码。为实现有效编译码，一般需达到以下两个目标：① 线性化编码；② 系统编码。对于目标①，准循环校验矩阵使得生成矩阵也具有循环阵形式，可实现线性化编码，减小复杂度和存储容量。对于目标②，系统编码有利于译码时直接从码字读出信息位，不需要额外的解方程运算。

一般而言，相比于二进制 LDPC 码，精心设计的多进制 LDPC 码在中短帧上具有更好的性能。文献[4,8]提出了二进制 QC LDPC 码的有效编码方法，由具有循环阵形式的满秩和秩亏校验矩阵可得到系统和不完全系统循环阵形式的生成矩阵。可见校验矩阵的秩与编码器能否系统编码密切相关，但是都没有给出构造满秩校验矩阵的方法，同时也没有系统地给出多进制 LDPC 码的编码方案。文献[9-12]对一类具有循环阵形式的校验矩阵进行了秩分析，但也未给出构造满秩校验矩阵的方法。本文提出了构造多进制 QC LDPC 码满秩校验矩阵的随机掩蔽方法，进而设计出线性复杂度的系统结构编码器。对多进制 LDPC 码编码器的设计分两步进行：其一，研究多进制 LDPC 码 QC 满秩校验矩阵的构造方法；其二，根据该奇偶校验矩阵的 QC 结构来研究编码器的设计。提出方法的优势在于：① 构造方法简单灵活，基于阵列形式的校验矩阵采用随机掩蔽可以得到满足不同码率需求的码字；② 适用范围广，同时适用于多进制和二进制 LDPC 码阵列形式校验矩阵的构造，可以构造出规则和非规则满秩校验矩阵的 LDPC 码；③ 线性复杂度系统编码，存储容量小，译码时由码字到信息位不需要进行额外求解运算。

1 多进制 QC LDPC 码构造

1.1 矩阵的二维扩展

首先构造出 $r\times n$ 的 $GF(q)$ 域下基矩阵：

$$\boldsymbol{W}=\begin{bmatrix}\boldsymbol{w}_0\\ \boldsymbol{w}_1\\ \vdots\\ \boldsymbol{w}_{r-1}\end{bmatrix}=\begin{bmatrix}w_{0,0} & w_{0,1} & \cdots & w_{0,n-1}\\ w_{1,0} & w_{1,1} & \cdots & w_{1,n-1}\\ \vdots & \vdots & \ddots & \vdots\\ w_{r-1,0} & w_{r-1,1} & \cdots & w_{r-1,n-1}\end{bmatrix} \tag{1}$$

式中，$w_{i,j}\in GF(q)$；$\boldsymbol{W}$ 满足行距离(row-distance, RD)约束[6]。令 $\boldsymbol{Z}$ 为矩阵 $\boldsymbol{W}$ 的指数矩阵，其中的元素 $z_{i,j}=\log w_{i,j}$。

将 $\boldsymbol{W}$ 的每一个元素进行二维扩展，可得到 $GF(q)$ 域上具有 $r\times n$ 阵列 $(q-1)\times(q-1)$ 大小的循环置换矩阵：

$$\boldsymbol{H}=\begin{bmatrix}\boldsymbol{A}_0\\ \boldsymbol{A}_1\\ \vdots\\ \boldsymbol{A}_{r-1}\end{bmatrix}=\begin{bmatrix}\boldsymbol{H}_{0,0} & \boldsymbol{H}_{0,1} & \cdots & \boldsymbol{H}_{0,n-1}\\ \boldsymbol{H}_{1,0} & \boldsymbol{H}_{1,1} & \cdots & \boldsymbol{H}_{1,n-1}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{H}_{r-1,0} & \boldsymbol{H}_{r-1,1} & \cdots & \boldsymbol{H}_{r-1,n-1}\end{bmatrix} \tag{2}$$

式中，$\boldsymbol{H}$ 的每一个阵列矩阵对应着基矩阵 $\boldsymbol{W}$ 相应位置上的一个元素，$\boldsymbol{W}$ 中的非零元素通过二维扩展生成一个大小为 $(q-1)\times(q-1)$ 的多进制循环置换矩阵，或称为乘 α 循环置换矩阵，零元素通过二维扩展生成一个大小为 $(q-1)\times(q-1)$ 的全零矩阵。

多进制二维扩展包括水平扩展和垂直扩展，具体方法如下：令 α 是 $GF(q)$ 域的本原元素，δ 是 $GF(q)$ 域中的非零元素。对于 $GF(q)$ 域中的非零元素 α^i，可以得到一个多进制向量 $\boldsymbol{z}(i)=[z_0,z_1,\cdots,z_{q-2}]$，其中第 i 个分量 z_i 为 α^i，其他所有分量为 0；对于 $GF(q)$ 域中的零元素，$\boldsymbol{z}(0)=[0,0,\cdots,0]$。由此得到的向量就称为域元素 α^i 的多进制位置向量。对非零元素 δ，水平扩展是多进制位置向量的生成过程，即 $\boldsymbol{disp}_{\mathrm{h}}(\delta)=\boldsymbol{z}(\delta)$；垂直扩展是垂直乘 α 的一个过程，即 $\boldsymbol{disp}_{\mathrm{v}}(\delta)=[\delta,\alpha\delta,\cdots,\alpha^{q-2}\delta]^{\mathrm{T}}$。而对零元素，$\boldsymbol{disp}_{\mathrm{h}}(\alpha^{-\infty})=\boldsymbol{disp}_{\mathrm{v}}(\alpha^{-\infty})^{\mathrm{T}}=[0,0,\cdots,0]$。因此，根据基矩阵中的任意 $w_{i,j}$ 可得到 $\boldsymbol{H}$ 中的一个阵列方阵：

$$\boldsymbol{H}_{i,j}=\boldsymbol{disp}_{\mathrm{h}}(\boldsymbol{disp}_{\mathrm{v}}(w_{i,j}))=\begin{bmatrix}\boldsymbol{z}(w_{i,j})\\ \boldsymbol{z}(\alpha w_{i,j})\\ \vdots\\ \boldsymbol{z}(\alpha^{q-2}w_{i,j})\end{bmatrix} \tag{3}$$

令 $\boldsymbol{P}$ 是根据 $GF(q)$ 域元素 α 进行二维扩展得到的 $(q-1)\times(q-1)$ 大小的多进制循环置换矩阵，则 $\boldsymbol{P}^i$ 是根据 $GF(q)$ 域元素 α^i 进行二维扩展得到的多进制循环置换矩阵，其中 $\boldsymbol{P}^{-\infty}=\boldsymbol{O}$。$\boldsymbol{H}$ 的每一个子阵列 $\boldsymbol{H}_{i,j}$ $(0\leqslant i<r, 0\leqslant j<n)$ 具有 $\boldsymbol{P}^i$ $(i\in\{-\infty,0,1,\cdots,q-2\})$ 的形式。α^i(或 $w_{i,j}$)决定多进制循环置换矩阵 $\boldsymbol{P}^i$(或 $\boldsymbol{H}_{i,j}$)。矩阵 $\boldsymbol{P}^i$ 可以表示成两个 $(q-1)\times(q-1)$ 矩阵 $\boldsymbol{Y}$ 和 $\boldsymbol{P}^{*i}$ 的乘积，即 $\boldsymbol{P}^i=\boldsymbol{Y}\cdot\boldsymbol{P}^{*i}$，其中 $\boldsymbol{P}^{*i}$ 是 $GF(q)$ 域上 $(q-1)\times(q-1)$ 的循环阵，$\boldsymbol{Y}$ 是一个以 $\alpha^0,\alpha,\cdots,\alpha^{q-2}$ 为主对角线，其余为 0 的 $(q-1)\times(q-1)$ 的方阵，记为 $\boldsymbol{Y}=\mathrm{diag}(\alpha^0,\alpha,\cdots,\alpha^{q-2})$。循环置换矩阵集合 $\mathcal{P}=(\boldsymbol{P}^{*0},\boldsymbol{P}^{*},\boldsymbol{P}^{*2},\cdots,\boldsymbol{P}^{*q-2})$ 在 $GF(q)$ 域矩阵乘法运算下形成 $q-1$ 阶的循环群，其中 $\boldsymbol{P}^{*0}$ 是单位元，$\boldsymbol{P}^{*i}$ 的逆为 $\boldsymbol{P}^{*q-1-i}$。

对任意一对 (d_v,d_c)，$\boldsymbol{H}(d_v,d_c)$ 是 $\boldsymbol{H}$ 的一个 $d_v\times d_c$ 的子阵列，其中 $1\leqslant d_v\leqslant r$，$1\leqslant d_c\leqslant n$，则 $\boldsymbol{H}(d_v,d_c)$ 是一个 $d_v(q-1)\times d_c(q-1)$ 的矩阵。d_v 和 d_c 是否为列重和行重与 $\boldsymbol{H}(d_v,d_c)$ 是否含有零阵有关。如果 $\boldsymbol{H}(d_v,d_c)$ 中不含零阵，则多进制稀疏校验矩阵 $\boldsymbol{H}(d_v,d_c)$ 的零空间将给出码长为 $d_c(q-1)$ 的多进制 LDPC 码 $\mathcal{C}$，其列重为 d_v，行重为 d_c，码率至少为 $(d_c-d_v)/d_c$。

1.2 满秩校验矩阵构造

引理 1[13] 对于如式(2)所示的 $\boldsymbol{H}$，设 $\boldsymbol{A}_i$ 为矩阵 $\boldsymbol{H}$ 的一个行阵列，其大小为 $(q-1)\times(q-1)n$，其中，$0\leqslant i<r$，$r<n$。则 $\mathrm{rank}(\boldsymbol{A}_i)=q-1$，$\mathrm{rank}(\boldsymbol{H})\leqslant r(q-1)-r+1$。

采用二维扩展构造出具有循环阵形式的校验矩阵 $\boldsymbol{H}(d_v,d_c)$ 的秩至多为 $d_v(q-1)-d_v+1$。通过分析可知，采用直接结构化方法构造出具有循环置换方阵的校验矩阵是秩亏的，但是各行阵列自身却是满秩的。只有通过适当的方法打破行阵列之间的相关性才能得到行满秩校验矩阵。

令 $\boldsymbol{W}(d_v,d_c)=[w_{i,j}]$ 是一个 $GF(2)$ 域上 $d_v\times d_c$ 的矩阵。定义如下特殊乘法：

$$\boldsymbol{M}(d'_v,d'_c)=\boldsymbol{W}(d_v,d_c)\otimes\boldsymbol{H}(d_v,d_c)=$$

$$\begin{cases}\boldsymbol{H}_{i,j},\ w_{i,j}=0(z_{i,j}=-\infty)\\ \boldsymbol{O},\ w_{i,j}=1(z_{i,j}\neq-\infty)\end{cases}\tag{4}$$

式中,$\boldsymbol{W}(d_v,d_c)=[w_{i,j}]$是一个 $GF(2)$域上 $d_v\times d_c$ 的矩阵。采用式(4)定义的乘法运算,根据 $\boldsymbol{W}(d_v,d_c)$中为 1 元素或 $\boldsymbol{Z}(d_v,d_c)$中不为$-\infty$元素的位置,将 $\boldsymbol{H}(d_v,d_c)$中对应位置的多进制循环置换矩阵掩蔽,实际上就是用一个零矩阵去取代 $\boldsymbol{H}(d_v,d_c)$中的一个阵列。$\boldsymbol{W}(d_v,d_c)$称为掩蔽基矩阵,$\boldsymbol{H}(d_v,d_c)$称为被掩蔽阵列矩阵,$\boldsymbol{M}(d'_v,d'_c)$称为已掩蔽阵列矩阵。本文定义掩蔽矩阵与文献[14]中掩蔽矩阵 0 和 1 的取值互反,$z_{i,j}$是一个二进制判决值,式(4)中 $z_{i,j}$是否为$-\infty$分别对应 $w_{i,j}$取 0 和 1。在 $\boldsymbol{M}(d'_v,d'_c)$中多进制循环置换矩阵的分布与 $\boldsymbol{W}(d_v,d_c)$中 0 的分布是一致的。如果 $\boldsymbol{H}(d_v,d_c)$满足 RC 约束,则无论掩蔽矩阵 $\boldsymbol{W}(d_v,d_c)$如何选取,$\boldsymbol{M}(d_v,d_c)$也将满足 RC 约束,所以 $\boldsymbol{M}(d_v,d_c)$Tanner 图的 girth 至少为 6。如果 $\bar{\boldsymbol{W}}(d_v,d_c)$和 $\boldsymbol{H}(d_v,d_c)$相应 Tanner 图的 girth 分别为 λ_1 和 λ_2,则 $\boldsymbol{M}(d'_v,d'_c)$对应 Tanner 图的 girth 至少为 $\max\{\lambda_1,\lambda_2\}$,其中 $\bar{\boldsymbol{W}}$ 中元素与 $\boldsymbol{W}$ 互反。因此,采用掩蔽方法可以使得 girth 增大或者减少 Tanner 图中短环数量。

为了得到满秩校验矩阵,本文将采用随机掩蔽方法来设计掩蔽矩阵,设计思路如下:

(1) 通过 $\boldsymbol{W}(d_v,d_c)$消除 $\boldsymbol{H}(d_v,d_c)$每个行阵列之间的相关性;

(2) $d_c>d_v$,即 $\boldsymbol{H}(d_v,d_c)$为一长方阵;

(3) 具有设计规则和不规则 LDPC 码的能力。

算法要求 $\boldsymbol{W}(d_v,d_c)$每行非零值的列位置互异,每行中非零值的个数分别为 $w_1,w_2,\cdots,w_{d_v}$,$0\leqslant w_1,w_2,\cdots,w_{d_v}\leqslant d_v$,且 $w_1+w_2+\cdots+w_{d_v}\leqslant d_c$。其中,$w_1,w_2,\cdots,w_{d_v}$是由用户确定的参数,称为行图样参数。

随机掩蔽算法具体表述如下:

步骤 1　在整数 0~d_c-1(列号)范围内产生长度为 d_c 不重复的随机整数序列 $\mathcal{S}(d_c)$;

步骤 2　根据行图样参数 $w_1,w_2,\cdots,w_i,\cdots,w_{d_v}$,将随机序列 $\mathcal{S}(d_c)$依据 w_i 的大小按顺序分成 d_v 个部分 $\mathcal{S}(w_1)$,$\mathcal{S}(w_2),\cdots,\mathcal{S}(w_{d_v})$,即 $\mathcal{S}(d_c)=[\mathcal{S}(w_1)\ \ \mathcal{S}(w_2)\ \cdots\ \mathcal{S}(w_{d_v})]$;

步骤 3　$\boldsymbol{W}(d_v,d_c)$中的第 i 行第 $\mathcal{S}(w_i)$列位置置 1,其余位置置 0;

步骤 4　通过式(4)采用随机掩蔽基矩阵 $\boldsymbol{W}(d_v,d_c)$对 $\boldsymbol{H}(d_v,d_c)$进行掩蔽得到 $\boldsymbol{M}(d'_v,d'_c)$;

步骤 5　计算 $\boldsymbol{M}(d'_v,d'_c)$的秩 rank ($\boldsymbol{M}(d'_v,d'_c)$),如果 rank ($\boldsymbol{M}(d'_v,d'_c)$)$=d_v(q-1)$,则搜寻到满秩矩阵 $\boldsymbol{M}(d'_v,d'_c)$,否则,跳至步骤 1。

根据需求确定合适的行图样参数后,结构化校验矩阵 $\boldsymbol{H}(d_v,d_c)$采用随机掩蔽矩阵 $\boldsymbol{Z}(d_v,d_c)$进行掩蔽后,一次搜索得到的 $\boldsymbol{M}(d_v,d_c)$在很大概率上是行满秩矩阵,而不必进行多次搜索,可以说通过随机掩蔽方法得到满秩校验矩阵是简单有效的。随机掩蔽的方法保留了原有校验矩阵的结构,并且使得矩阵更为稀疏,减少了短环或减小了短环存在的数量,对性能具有一定的改善作用。无论是多进制,还是二进制,提出的随机掩蔽方法对于任何具有阵列形式校验矩阵构造的码字均可适用。

2　快速编码设计

多进制 QC LDPC 码的校验矩阵是由循环置换矩阵构成的阵列,由校验矩阵不但可以得到系统形式的生成矩阵,而且生成矩阵也具有循环阵列形式。虽然生成矩阵中循环方阵通常不是稀疏的,但只需保存循环阵中的第一行(列)即可,编码时采用反馈移位寄存器进行循环移位编码。串行编码的复杂度与校验位数成正比,并行编码的复杂度与码长成正比。

将构造的多进制满秩校验矩阵记为 $\boldsymbol{H}$,其中每个阵列记为 $\boldsymbol{H}_{i,j}$,则 $\boldsymbol{H}^*_{i,j}$是满足式 $\boldsymbol{H}_{i,j}=\boldsymbol{Y}\cdot\boldsymbol{H}^*_{i,j}$的循环阵。将 $\boldsymbol{H}^*$ 进行列阵列重排得到 $\boldsymbol{H}^*_p$,其最右边的满秩 $r\times r$ 子阵列为

$$\boldsymbol{D}^*=\begin{bmatrix}\boldsymbol{H}^*_{1,(n-r+1)} & \boldsymbol{H}^*_{1,(n-r+2)} & \cdots & \boldsymbol{H}^*_{1,n}\\ \boldsymbol{H}^*_{2,(n-r+1)} & \boldsymbol{H}^*_{2,(n-r+2)} & \cdots & \boldsymbol{H}^*_{2,n}\\ \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{H}^*_{r,(n-r+1)} & \boldsymbol{H}^*_{r,(n-r+2)} & \cdots & \boldsymbol{H}^*_{r,n}\end{bmatrix}\tag{5}$$

对应重排阵列 $\boldsymbol{H}^*_p$,大小为$(n-r)(q-1)\times n(q-1)$生成矩阵具有如下形式:

$$\boldsymbol{G}=\begin{bmatrix}\boldsymbol{G}_1\\ \boldsymbol{G}_2\\ \vdots\\ \boldsymbol{G}_{n-r}\end{bmatrix}=\begin{bmatrix}\boldsymbol{I} & \boldsymbol{0} & \cdots & \boldsymbol{0} & \boldsymbol{G}_{1,1} & \boldsymbol{G}_{1,2} & \cdots & \boldsymbol{G}_{1,r}\\ \boldsymbol{0} & \boldsymbol{I} & \cdots & \boldsymbol{0} & \boldsymbol{G}_{2,1} & \boldsymbol{G}_{2,2} & \cdots & \boldsymbol{G}_{2,r}\\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{I} & \boldsymbol{G}_{n-r,1} & \boldsymbol{G}_{n-r,2} & \cdots & \boldsymbol{G}_{n-r,r}\end{bmatrix}\tag{6}$$

式中,$\boldsymbol{I}$ 为$(q-1)\times(q-1)$的单位阵;$\boldsymbol{0}$ 为$(q-1)\times(q-1)$的全零阵;$\boldsymbol{G}_{i,j}$为$(q-1)\times(q-1)$的循环阵,$1\leqslant i\leqslant n-r$,$1\leqslant j\leqslant r$。

令 $\boldsymbol{z}_i=[\boldsymbol{g}_{i,1},\boldsymbol{g}_{i,2},\cdots,\boldsymbol{g}_{i,c}]$为 $\boldsymbol{G}_i$ 最右 r 部分$[\boldsymbol{G}_{i,1}\ \ \boldsymbol{G}_{i,2}\ \cdots\ \boldsymbol{G}_{i,r}]$第一行的取值,$\boldsymbol{g}_{i,j}$为 $\boldsymbol{G}_{i,j}$的生成元,$\boldsymbol{u}=[1,0,\cdots,0]$,其向量长度为 $q-1$,$\boldsymbol{M}_i=[\boldsymbol{H}^{\mathrm{T}}_{1,i}\ \ \boldsymbol{H}^{\mathrm{T}}_{2,i}\ \ \cdots\ \ \boldsymbol{H}^{\mathrm{T}}_{c,i}]^{\mathrm{T}}$ 为 $\boldsymbol{H}^*_p$ 的第 i 个列阵列,由 $\boldsymbol{H}^*_p\boldsymbol{G}^{\mathrm{T}}=\boldsymbol{0}$ 可得

$$\boldsymbol{M}_i\boldsymbol{u}^{\mathrm{T}}+\boldsymbol{D}^*\boldsymbol{z}_i^{\mathrm{T}}=\boldsymbol{0}\tag{7}$$

由于 $\boldsymbol{D}^*$ 的秩 rank ($\boldsymbol{D}^*$)$=r(q-1)$且为满秩矩阵,因而 $\boldsymbol{D}^*$ 非奇异且存在 $GF(q)$域上的逆矩阵 $\boldsymbol{D}^{*-1}$。由式(7)可得

$$\boldsymbol{z}_i^{\mathrm{T}}=-\boldsymbol{D}^{*-1}\boldsymbol{M}_i\boldsymbol{u}^{\mathrm{T}}\tag{8}$$

对于 $1\leqslant i\leqslant n-r$,求解式(8)可得 $\boldsymbol{z}_1,\boldsymbol{z}_2,\cdots,\boldsymbol{z}_{n-r}$,根据 $\boldsymbol{z}_1,\boldsymbol{z}_2,\cdots,\boldsymbol{z}_{n-r}$可得 $\boldsymbol{G}$ 中除单位阵和全零阵外所有$(n-r)\times r$ 阵列的生成元 $\boldsymbol{g}_{i,j}$。

由多进制校验矩阵 $\boldsymbol{H}(d_v,d_c)$得到生成矩阵 $\boldsymbol{G}$ 的步骤如下:

步骤 1　预处理:将 $\boldsymbol{H}$ 中每个阵列 $\boldsymbol{H}_{i,j}$用 $\boldsymbol{H}^*_{i,j}$取代;

步骤 2　列阵列重排:将 $\boldsymbol{H}^*$ 的列阵列重排得到 $\boldsymbol{H}^*_p$,使得 $\boldsymbol{H}$ 最右边 $r\times r$ 子阵列 $\boldsymbol{D}^*$ 的秩为 rank ($\boldsymbol{D}^*$)$=r(q-1)$;

步骤 3　计算生成矩阵阵列生成元:根据式(8)得到所有生成元 $\boldsymbol{g}_{i,j}$;

步骤 4　产生生成矩阵:向量 $\boldsymbol{g}_{i,j}$作为 $\boldsymbol{G}_{i,j}$的第一行,$\boldsymbol{G}_{i,j}$中的每一行都是上一行的向右循环移位,最终生成 $\boldsymbol{G}$。

生成矩阵 $\boldsymbol{G}$ 形成之后,可采用不同的方法来实现线性化编码。从不同的角度理解,上述数学表述可以通过串行和并行两种方式进行工程实现。

2.1 串行编码器

令 $\boldsymbol{a}=[\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_{n-r}]$是具有$(n-r)(q-1)$个符号的信息序列，$\boldsymbol{a}$ 的第 i 段为 $\boldsymbol{a}_i=[a_{(i-1)(q-1)+1},a_{(i-1)(q-1)+2},\cdots,a_{i(q-1)}]$。采用满秩校验矩阵经编码器编码后得系统码字$\boldsymbol{c}=\boldsymbol{aG}=[\boldsymbol{a},\boldsymbol{p}_1,\boldsymbol{p}_2,\cdots,\boldsymbol{p}_r]$，其中，$\boldsymbol{p}_j=[p_{j,1},p_{j,2},\cdots,p_{j,(q-1)}]$。由 $\boldsymbol{c}=\boldsymbol{aG}$ 得

$$\boldsymbol{p}_j=\boldsymbol{a}_1\boldsymbol{G}_{1,j}+\boldsymbol{a}_2\boldsymbol{G}_{2,j}+\cdots+\boldsymbol{a}_{n-r}\boldsymbol{G}_{n-r,j} \tag{9}$$

由式(9)可知，$\boldsymbol{a}_i\boldsymbol{G}_{i,j}$ 实际上是 $\boldsymbol{G}_{i,j}$ 的生成元 $\boldsymbol{g}_{i,j}$ 按照 $\boldsymbol{a}_i$ 中 $q-1$ 个连续信息符号进行循环移位累加的结果。$\boldsymbol{g}_{i,j}^{(l)}$ 是生成元 $\boldsymbol{g}_{i,j}$ 向右循环 l 位，或者说是 $\boldsymbol{G}_{i,j}$ 的第 l 行，$0\leqslant l<q-1$。串行编码的基本思想是：对于第 $j(1\leqslant j\leqslant r)$段校验位，采用的是对 $\boldsymbol{G}$ 中 r 个列阵列进行加载循环移位累加的运算，对于一个列阵列中的 $n-r$ 个子阵列需要加载 $n-r$ 次生成元。图 1 为串行移位寄存累加器(serial shift-register-adder-accumulator，SSRAA)单元。

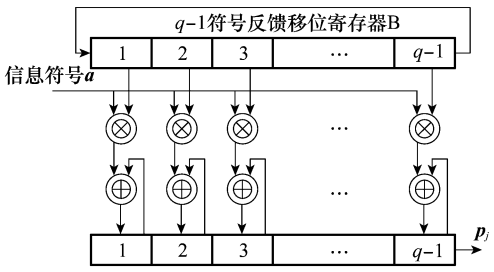

图 1 串行移位寄存累加器单元

串行编码器得到校验位 $\boldsymbol{p}_j$ 的具体步骤如下：

步骤 1 初始化：寄存器 A 所有单元置 0，移位寄存器 B 中载入生成元 $\boldsymbol{g}_{1,j}^{(0)}$；

步骤 2 将信息符号 a_1 移入到编码器，同时也送入调制器，经过 $GF(q)$域乘法器得 $a_1\boldsymbol{g}_{1,j}^{(0)}$，与寄存器 A 中已存储元素经 $GF(q)$域加法器相加，再存入寄存器 A 中；$\boldsymbol{g}_{1,j}^{(0)}$ 向右移位得到 $\boldsymbol{g}_{1,j}^{(1)}$，下一个信息符号 a_2，经乘法器得到 $a_2\boldsymbol{g}_{1,j}^{(1)}$，再经累加器得到 $a_1\boldsymbol{g}_{1,j}^{(0)}+a_2\boldsymbol{g}_{1,j}^{(1)}$；重复上述步骤，经过 $q-1$ 个符号后，寄存器 A 得到 $\boldsymbol{a}_1\boldsymbol{G}_{1,j}$；

步骤 3 重载生成元 $\boldsymbol{g}_{2,j}^{(0)}$，从信息符号 a_q 开始重复步骤 2，最终在寄存器 A 中得到 $\boldsymbol{a}_1\boldsymbol{G}_{1,j}+\boldsymbol{a}_2\boldsymbol{G}_{2,j}$，接着依次重载生成元 $\boldsymbol{g}_{3,j}^{(0)},\boldsymbol{g}_{4,j}^{(0)},\cdots,\boldsymbol{g}_{n-r,j}^{(0)}$，得到第 j 段的校验位 $\boldsymbol{p}_j=\boldsymbol{a}_1\boldsymbol{G}_{1,j}+\boldsymbol{a}_2\boldsymbol{G}_{2,j}+\cdots+\boldsymbol{a}_{n-r}\boldsymbol{G}_{n-r,j}$。

为了得到 r 段校验位，可以采用 r 个 SSRAA 单元同时进行计算，得到所有 $\boldsymbol{p}=[\boldsymbol{p}_1,\boldsymbol{p}_2,\cdots,\boldsymbol{p}_r]$。图 2 为基于 SSRAA 的编码结构，需要$(n-r)(q-1)$个时钟才可计算出所有 r 个校验位，其中的每一个时钟得到的都只是校验位的中间值。

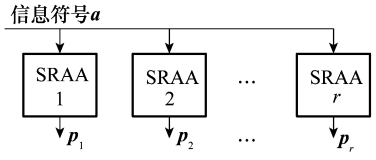

图 2 基于 SRAA 的 QC LDPC 码编码结构

2.2 并行编码器

对于 $1\leqslant i\leqslant n-r$，$1\leqslant j\leqslant r$，令 $\boldsymbol{v}_{i,j}$ 是阵列 $\boldsymbol{G}_{i,j}$ 的第一列。对于 $0\leqslant l<q-1$，$\boldsymbol{v}_{i,j}^{(l)}$ 是生成元 $\boldsymbol{v}_{i,j}$ 向下循环 l 位，或者说是 $\boldsymbol{G}_{i,j}$ 的第 l 列，$0\leqslant l<q-1$。并行编码的基本思想是：对于第 $j(1\leqslant j\leqslant r)$段校验位，采用对 $\boldsymbol{G}$ 中 r 个列阵列进行的运算，对于一个列阵列一次性地载入 $n-r$ 个生成元进行循环移位累加运算。图 3 为并行 SRAA(parallel shift-register-adder-accumulator，PSRAA)单元。

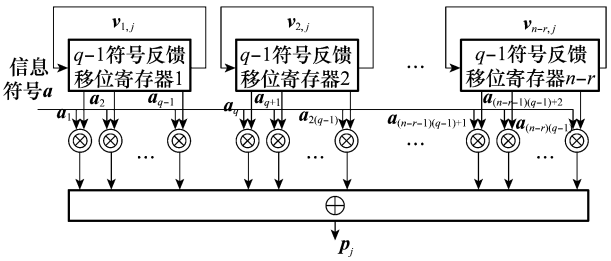

图 3 并行移位寄存累加器单元

并行编码器得到校验位 $\boldsymbol{p}_j$ 的具体步骤如下：

步骤 1 初始化：$n-r$ 个反馈移位寄存器分别载入 $\boldsymbol{v}_{1,j}^{(0)},\boldsymbol{v}_{2,j}^{(0)},\cdots,\boldsymbol{v}_{n-r,j}^{(0)}$；

步骤 2 并行输入整个信息位 $\boldsymbol{a}=[\boldsymbol{a}_1,\boldsymbol{a}_2,\cdots,\boldsymbol{a}_{n-r}]$，计算第 j 段校验的第 0 个校验符号为 $p_{j,0}=\boldsymbol{a}_1\boldsymbol{v}_{1,j}^{(0)}+\boldsymbol{a}_2\boldsymbol{v}_{2,j}^{(0)}+\cdots+\boldsymbol{a}_{n-r}\boldsymbol{v}_{n-r,j}^{(0)}$，$\boldsymbol{v}_{i,j}^{(0)}$ 向下移位得到 $\boldsymbol{v}_{i,j}^{(1)}$，$1\leqslant i\leqslant n-r$，进而得到 $p_{j,1}$，重复以上步骤，可得到其余 $p_{j,l}$，$0\leqslant l<q-1$。

为了得到每段校验位的所有符号，可以采用 c 个 PSRAA 单元同时进行计算，得到所有 $\boldsymbol{p}=[\boldsymbol{p}_1,\boldsymbol{p}_2,\cdots,\boldsymbol{p}_r]$，因此，图 2 也可作为基于 PSRAA 的编码结构。一个时钟内可计算出所有 r 个校验位中的一个符号。

2.3 复杂度分析

串行编码器共有 r 个 SSRAA 单元，对于 $1\leqslant i\leqslant n-r$，$1\leqslant j\leqslant r$，第 j 个单元载入和移位表示分别为$[\boldsymbol{g}_{1,j}^{(0)}\rightarrow\boldsymbol{g}_{1,j}^{(1)}\rightarrow\cdots\rightarrow\boldsymbol{g}_{1,j}^{(q-2)}]\rightarrow[\boldsymbol{g}_{2,j}^{(0)}\rightarrow\boldsymbol{g}_{2,j}^{(1)}\rightarrow\cdots\rightarrow\boldsymbol{g}_{2,j}^{(q-2)}]\rightarrow\cdots\rightarrow[\boldsymbol{g}_{n-r,j}^{(0)}\rightarrow\boldsymbol{g}_{n-r,j}^{(1)}\rightarrow\cdots\rightarrow\boldsymbol{g}_{n-r,j}^{(q-2)}]$，方括号内第一个向量 $\boldsymbol{g}_{i,j}^{(0)}$ 表示载入值，方括号内其余 $q-2$ 个值表示反馈移位寄存器的 $q-2$ 次移位，可见，编码的时钟周期为$(n-r)(q-1)$。并行编码器共有 r 个 PSRAA 单元，第 j 个单元载入和移位分别表示为$[\boldsymbol{v}_{1,j}^{(0)},\boldsymbol{v}_{2,j}^{(0)},\cdots,\boldsymbol{v}_{n-r,j}^{(0)}\rightarrow\boldsymbol{v}_{1,j}^{(1)},\boldsymbol{v}_{2,j}^{(1)},\cdots,\boldsymbol{v}_{n-r,j}^{(1)}\rightarrow\cdots\rightarrow\boldsymbol{v}_{1,j}^{(q-2)},\boldsymbol{v}_{2,j}^{(q-2)},\cdots,\boldsymbol{v}_{n-r,j}^{(q-2)}]$，方括号内第一个向量 $\boldsymbol{v}_{1,j}^{(0)},\boldsymbol{v}_{2,j}^{(0)},\cdots,\boldsymbol{v}_{n-r,j}^{(0)}$ 表示所有的载入值，方括号内“→”之后表示 $q-2$ 次移位。表 1 具体给出了两种编码器编码速率和复杂度的比较。

表 1 两种编码器的速度和复杂度比较

编码方案	编码速率(时钟周期)	触发器	两输入 $GF(q)$ 域加法器	两输入 $GF(q)$ 域乘法器
串行编码	$(n-r)(q-1)$	$2r(q-1)$	$r(q-1)$	$r(q-1)$
并行编码	$q-1$	$r(n-r)\cdot(q-1)$	$r(n-r)\cdot(q-1)$	$r((n-r)\cdot(q-1)-1)$

3　仿真结果及性能分析

本节对随机掩蔽方法构造的 LDPC 码进行了性能比较和分析。所有仿真中的信号均采用二位移相键控(binary phase shift keying, BPSK)调制,且在双边功率谱密度为 $N_0/2$ 的加性高斯白噪声(additive white Gaussian noise, AWGN)信道中传输。译码采用 FFT-QSPA 算法,且最大迭代次数设置为 50。

对高码率 0.842 9 的(961,810) 32 进制有限域第Ⅱ种方法[15]构造的 LDPC 码进行研究。该码的校验矩阵 $\boldsymbol{H}$ 是一个由 31×31 循环置换矩阵的 5×31 阵列构成的大小为 155×961 矩阵,其列重行重分别为 5 和 31,秩为 151。采用随机掩蔽算法,行图样参数 $w_1, w_2, \cdots, w_5$ 均取 1,得到满秩校验矩阵 $\boldsymbol{M}$,用计算机搜索得到的掩蔽基矩阵 $\boldsymbol{W}$ 为

$$\boldsymbol{W}=\begin{bmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix} \tag{10}$$

通过随机掩蔽方法得到码率为 0.838 7 的 32 进制(961,806)不规则 LDPC 码,其校验矩阵的列重为 4 和 5,行重为 30,秩为 155。采用 FFT-QSPA 译码算法,掩蔽前后码字的误比特率(bit error rate, BER)、误符号率(symbol error rate, SER)和误帧率(frame error rate, FER)性能如图 4 所示,同时也给出了同比特长度 RS 码的 BM 算法的误码性能。在 FER 为 10^{-4} 处,随机掩蔽后的 LDPC 码相对于 GF(2^{10})域上采用 BM(Berlekamp-Massey)算法的(481,403) RS 码而言,取得了大约 1.92 dB 的编码增益;相对于基于未掩蔽校验矩阵 $\boldsymbol{H}$ 的 LDPC 码而言,取得了大约 0.03 dB 的编码增益。从性能上看,通过随机掩蔽方法,在码长码率基本变化不大的情况下,码字的误码性能相对改善了。图 5 给出了掩蔽前后 LDPC 码的误帧性能和迭代性能曲线图,掩蔽后的 LDPC 码在性能和收敛速度上都得到了提高。

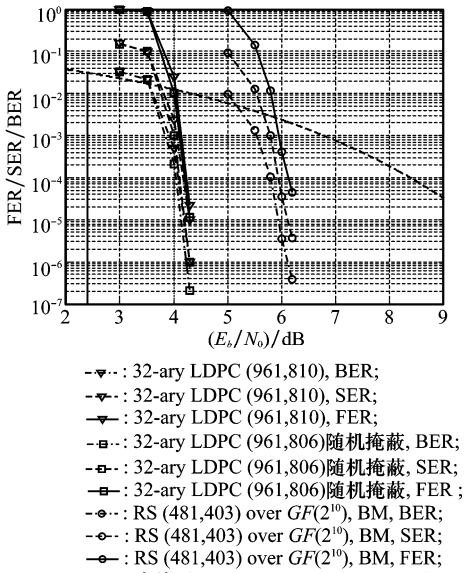

图 4　采用 FFT-QSPA 译码的 32-ary (961,810) LDPC 码、32-ary (961,806)掩蔽 LDPC 码和采用 BM 算法 $GF(2^{10})$域的(481,403)RS 码的误码性能

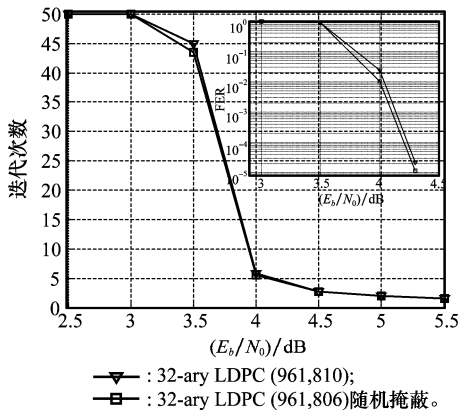

图 5　采用 FFT-QSPA 译码的 32-ary (961,810) LDPC 码和 32-ary (961,806)掩蔽 LDPC 码的误码性能和迭代性能

采用上述方法,得到的多进制的 LDPC 码是不规则的,为了得到规则多进制 LDPC 码,考查由文献[11]构造的校验矩阵 $\boldsymbol{H}$,具有 15×15 循环置换矩阵的 4×8 阵列构成大小为 60×120 的矩阵,其秩为 49,列重和行重分别为 4 和 8。$\boldsymbol{H}$ 的零空间给出了一个码率为 0.591 7 的 16 进制(120,71)规则 LDPC 码。通过随机掩蔽方法要得到规则的满秩校验矩阵,就要求掩蔽矩阵每行的行重和每列的列重分别相等,这样,行图样参数 w_1, w_2, w_3, w_4 取值应为 $d_c/d_v=2$。在 10 000 次搜索中,对 60×120 矩阵掩蔽后各秩出现次数的分布情况进行分析可知,得到满秩的概率基本趋近于 1,只有极少数分布在其他秩上。通过随机掩蔽方法,得到掩蔽基矩阵:

$$\boldsymbol{W}=\begin{bmatrix}
0&1&0&0&0&1&0&0\\
0&0&1&1&0&0&0&0\\
1&0&0&0&0&0&0&1\\
0&0&0&0&1&0&1&0
\end{bmatrix} \tag{11}$$

掩蔽之后得到的是 GF(2^4)域上码率为 0.5 的(120,60)规则 LDPC 码。掩蔽前后的多进制 LDPC 码的误码性

能曲线如图 6 所示。在 FER 为 10^{-5} 处，掩蔽后的(120,60) LDPC 码相对于掩蔽前(120,71) LDPC 码来说，取得了大约0.08 dB的编码增益。图 7 给出了掩蔽前后 LDPC 码的平均迭代次数曲线图，掩蔽后的 LDPC 码相比于原 LDPC 码来说，收敛特性也得到了进一步的提升。可以说采用随机掩蔽构造的 LDPC 码的误码性能和收敛特性都优于直接结构化构造的 LDPC 码。

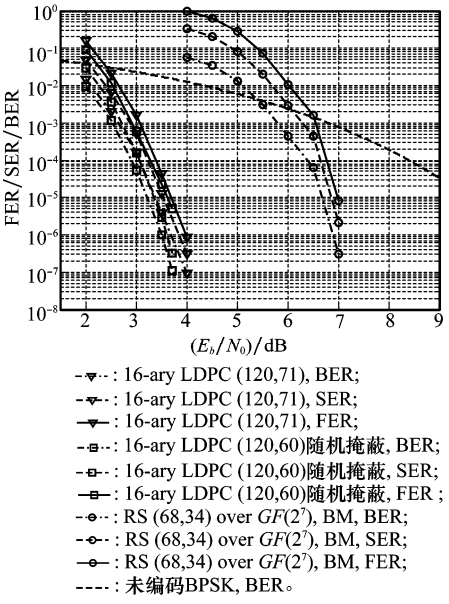

图 6 采用 FFT-QSPA 译码的 16-ary (120,71) LDPC 码、16-ary (120,60)掩蔽 LDPC 码和采用 BM 算法 $GF(2^7)$域的(68,34)RS 码的误码性能

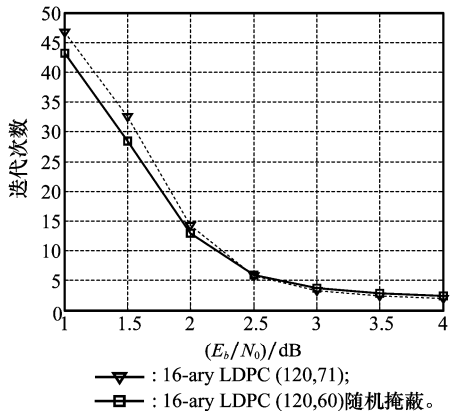

图 7 采用 FFT-QSPA 译码的 16-ary (120,71) LDPC 码和 16-ary (120,60)掩蔽 LDPC 码的平均迭代次数

为了更加清晰地表明编码器和译码器的工作过程和随机掩蔽方法带来的优势，采用上述随机掩蔽构造的 16 进制(120,60)规则 LDPC 码在 3 dB 信噪比下传输一幅如图 8 所示的熊猫图像。首先，将图像分成 1 050 帧，采用上述串行或并行编码器对图像进行编码，得到图像编码后的码字为系统码，直接包括信息位和校验位。通过 BPSK 调制进入到信道，将接收到的图像送入 LDPC 码的迭代译码器。译码器的最大迭代次数设为 50。随着每次迭代的进行，码字的错误不断地得到纠正，第 25 次迭代后的码字满足校验方程，停止迭代，最后译码的码字和信息符号都是无误的。

通过以上仿真，可以得出以下结论：① 随机掩蔽后码字的误码和迭代性能一般优于随机掩蔽前的码字；② 随机掩蔽能产生满秩校验矩阵，使编码复杂度降低，且编出的码字都为系统码。

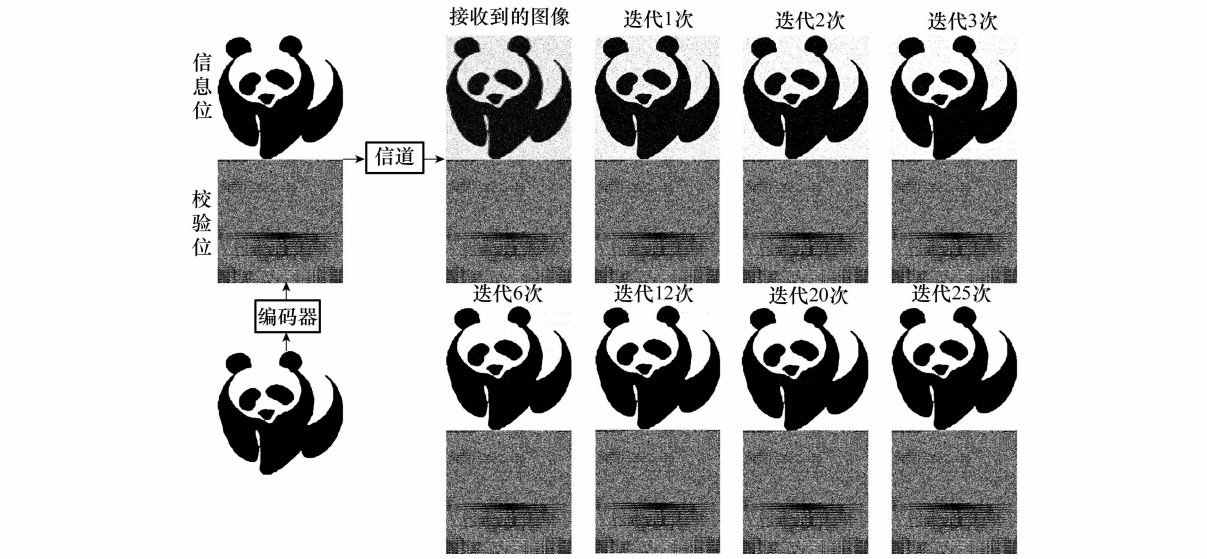

图 8 3 dB 时采用 16-ary (120,60) 掩蔽 LDPC 码进行数据传输的编译码过程和效果

4 结束语

多进制准循环 LDPC 码由于校验矩阵结构的规则性使得编译码具有较低的复杂度。本文提出了一种基于循环阵列校验矩阵的满秩构造方法——随机掩蔽。具有准循环结构的校验矩阵可以产生循环阵形式的生成矩阵，利用得到

的生成矩阵进行编码,会使编码的复杂度和存储容量大大降低。为了实现系统编码,需要利用构造的满秩校验矩阵,满秩校验矩阵可以解决两个问题:一是编码器可实现系统编码;二是译码端由码字得到信息位不需要额外的解方程运算。提出的构造满秩校验矩阵方法具有通用性,适用于所有的循环阵形式的校验矩阵,成功解决了 LDPC 码系统和线性化编码的问题,不足之处在于掩蔽前后码率会发生一定的变化。仿真结果表明,采用随机掩蔽方法构造出码字的误码性能和编码复杂度都优于随机掩蔽前的码字。因此,该方法在无线数据传输中具有很强的竞争实力,为无线数据传输的纠错码提供了一种实用工程方案。

参考文献:

[1] Gallager R. Low-density parity-check codes[D]. Cambridge, MA: Massachusetts Institute of Technology, 1963.

[2] Davey M C, MacKay D. Low-density parity check codes over GF(q)[J]. *IEEE Communications Letters*, 1998, 2(6): 165 - 167.

[3] Barnault L, Declercq D. Fast decoding algorithm for LDPC over GF(2^q)[C]// *Proc. of the IEEE Information Theory Workshop*, 2003: 70 - 73.

[4] Li Z, Chen L, Zeng L, et al. Efficient encoding of quasi-cyclic low-density parity-check codes[J]. *IEEE Trans. on Communications*, 2006, 54(1): 71 - 81.

[5] Chen X, Men A, Yang B, et al. Construction of LDPC codes over GF(q) with modified progressive edge growth[J]. *The Journal of China Universities of Posts and Telecommunications*, 2009, 16(5): 103 - 106.

[6] Kang J, Huang Q, Zhang L, et al. Quasi-cyclic LDPC codes: an algebraic construction[J]. *IEEE Trans. on Communications*, 2010, 58(5): 1383 - 1396.

[7] Chen C, Bai B, Wang X, et al. Nonbinary LDPC codes constructed based on a cyclic MDS code and a low-complexity nonbinary message-passing decoding algorithm[J]. *IEEE Communications Letters*, 2010, 14(3): 239 - 241.

[8] Zeng L. Algebraic constructions of nonbinary quasi-cyclic LDPC codes and efficient encoding[D]. Davis: University of California, 2006.

[9] Kamiya N, Sasaki E. Efficient encoding of QC-LDPC codes related to cyclic MDS codes[J]. *IEEE Journal on Selected Areas in Communications*, 2009, 27(6): 846 - 854.

[10] Kamiya N. Efficiently encodable irregular QC-LDPC codes constructed from self-reciprocal generator polynomials of MDS codes[J]. *IEEE Communications Letters*, 2010, 14(9): 860 - 862.

[11] Zhang L, Huang Q, Lin S, et al. Quasi-cyclic LDPC codes: an algebraic construction, rank analysis, and codes on Latin squares[J]. *IEEE Trans. on Communications*, 2010, 58(11): 3126 - 3139.

[12] Zhang L, Lin S, Abdel-Ghaffar K, et al. Circulant arrays: rank analysis and construction of quasi-cyclic LDPC codes[C]// *Proc. of the IEEE International Symposium Information Theory*, 2010: 814 - 818.

[13] Fossorier M. Quasi-cyclic low-density parity-check codes from circulant permutation matrices[J]. *IEEE Trans. on Information Theory*, 2004, 50(8): 1788 - 1793.

[14] Zhou B, Kang J, Tai Y Y, et al. High performance non-binary quasi-cyclic LDPC codes on euclidean geometries[J]. *IEEE Trans. on Communications*, 2009, 57(5): 1298 - 1311.

[15] Zeng L, Lan L, Tai Y Y, et al. Constructions of nonbinary quasi-cyclic LDPC codes: a finite field approach[J]. *IEEE Trans. on Communications*, 2008, 56(4): 545 - 554.

(上接第 2310 页)

[2] Eustice R M, Pizarro O, Singh H. Visually augmented navigation for autonomous underwater vehicles[J]. *IEEE Journal of Oceanic Engineering*, 2008, 33(2): 103 - 122.

[3] Rice H, Kelmenson S, Mendelsonhn L. Geophysical navigation technologies and applications[C]// *Proc. of the IEEE Position Location and Navigation Symposium*, 2004: 618 - 624.

[4] Gleason D M. Passive airborne navigation and terrain avoidance using gravity gradiometry[J]. *Journal of Guidance, Control and Dynamics*, 1995, 18(6): 1450 - 1458.

[5] Yuh J. Design and control of autonomous underwater robots: a survey[J]. *Autonomous Robots*, 2000, 8: 7 - 24.

[6] 李俊,沈安文,宋保维,等. 基于多普勒速度声纳的水下航行器导航方法[J]. 华中科技大学学报(自然科学版),2004,32(1):73 - 75. (Li J, Shen A W, Song B W, et al. The methods and shipping trial of the navigation of autonomous underwater vehicles based on Doppler dead reckon[J]. *Journal of Huazhong University of Science and Technology (Nature Science Edition)*, 2004, 32(1): 73 - 75.)

[7] Leader D E. Kalman filter estimation of underwater vehicle position and attitude using velocity aided by inertial motion unit[D]. Massachusetts: Massachusetts Institute of Technology, 1994.

[8] 李俊,徐德民,宋保维,等. 自主式水下潜器导航技术发展现状与展望[J]. 中国造船,2004,45(3):70 - 77. (Li J, Xu D M, Song B W, et al. Development and prospect of AUV navigation technology[J]. *Ship Building of China*, 2004, 45(3): 70 - 77.)

[9] 刘为任,庄良杰. 潜用惯导系统误差估计技术研究[J]. 应用科学学报,2005,23(3):324 - 326. (Liu W R, Zhuang L J. A study on the error estimation of INS installed on submarines[J]. *Journal of Applied Sciences*, 2005, 23(3): 324 - 326.)

[10] Yoshii T. Method for detecting a magnetic source by measuring the magnetic field thereabout [P]. Unite States Patent: 4309659, 1982 - 01 - 05.

[11] 黄玉,孙枫,郝燕玲. 载体姿态变化对水下磁场定位精度的影响及监测方法[J]. 系统工程与电子技术,2010,32(9):1982 - 1986. (Huang Y, Sun F, Hao Y L. Effect of vehicle attitude variation on underwater magnetic field localization precision and its detection method[J]. *Systems Engineering and Electronics*, 2010, 32(9): 1982 - 1986.)

[12] 黄玉,郝燕玲. 偶极子磁场中磁梯度测量装置误差对载体定位精度影响分析[J]. 华中科技大学学报(自然科学版),2010,38(9):76 - 81. (Huang Y, Hao Y L. Influence of measurement errors from magnetic dipole upon determination of underwater vehicle precision[J]. *Journal of Huazhong University of Science and Technology (Nature Science Edition)*, 2010, 38(9): 76 - 81.)