

一种新的准随机 LDPC 卷积码及窗译码

彭万权¹, 伍小兵¹, 张承畅², 冯文江²

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

摘要: 以 Gallager 随机 LDPC 分组码为基础, 通过列置换、置数、剪切与合并, 构造了一种新的 $(l, 3, 6)$ 准随机 LDPC 卷积码. 针对该码类, 提出了一种窗扇尺寸固定、滑动步长可选的窗译码算法, 大幅降低了寄存器开销和译码延时. 仿真实验界定了滑动步长的取值范围, 验证了构造方案和窗译码的有效性, 测试了不同约束度的准随机 LDPC 卷积码的误码性能, 结果显示约束度 l 仅为 1535 即可获得距离香农限约 1dB 的纠错能力.

关键词: 卷积码; 低密度校验码; 时变周期卷积码; 流水线译码; 窗译码

中图分类号: TN911.22 **文献标识码:** A **文章编号:** 0372-2112 (2014)07-1379-08

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

DOI: 10.3969/j.issn.0372-2112.2014.07.021

A New Quasi-Random LDPC Convolutional Codes and Window Decoding

PENG Wan-quan¹, WU Xiao-bing¹, ZHANG Cheng-chang², FENG Wen-jiang²

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

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

Abstract: Using column permutation, number setting, cutting and merging, a new $(1, 3, 6)$ quasi-random LDPC convolutional codes is derived from Gallager LDPC block code. Aiming at this codes, the paper advances a windowed decoding, which the size of window sash is invariable and sliding step is optional, and it can considerably reduce the size of the register and decoding delay. The simulation defines the value range of sliding step and confirms the effectiveness of the construction method and windowed decoding, and also tests the BER performance of quasi-random LDPC convolutional codes with different constraint degrees. The results show that $l = 1535$ can obtain an error-correcting ability about 1 dB from the capacity limit.

Key words: convolutional codes; low-density parity-check codes; time-varying periodic convolutional codes; pipeline decoding; windowed decoding

1 引言

LDPC 卷积码是一种由半无限长稀疏校验矩阵定义的纠错码类, 兼具卷积码和 LDPC 分组码的优点, 是当前编码领域的一个新的研究热点. 目前, LDPC 卷积码基本上是以 LDPC 分组码为基础进行构造的, 据此可将其划分为三类: 第一类为随机 LDPC 卷积码, Felstrom 在其开创性文献中^[1], 通过对 Gallager 随机 LDPC 分组码的校验矩阵进行剪切、插入、合并等一系列处理, 获得了半无限长校验矩阵; 第二类为准循环 LDPC 卷积码^[2~5], 该类码是以准循环 (Quasi-Cyclic, QC) LDPC 分组码^[6]为基础, 采用一种分组码构造卷积码的方法^[7]构造而得; 第三类为原模图 LDPC 卷积码^[8,9], 借鉴原模图 LDPC 分组码^[10]的复制、置换等方法, 构造性能优异、多种码率的原模图 LDPC 卷积码. 以上三类构造均须对 LDPC 分

组的校验矩阵实施对角剪切、复制、合并等处理, 同时, 为了提高信息位的使用效率, 增强码的纠错性能, 也都采用了时变周期卷积码^[11]的编码策略. 与 LDPC 分组码相比, LDPC 卷积码具有诸多优点: 无需分块便能完成连续数据的编译码, 特别适合可变帧长度的通信系统^[12]; 编码复杂度仅随码长呈线性增长, 比 LDPC 分组码更具竞争力; 可借助置信传播算法实现大约约束度卷积码的译码, 充分挖掘了卷积码记忆特性所带来的高性价比. 然而, 进一步文献检索表明, 虽然目前已有不少 LDPC 卷积码的相关成果, 但其研究份额却仍不及 LDPC 分组码, 特别是第一类随机 LDPC 卷积码, 其研究力度明显不如后两者, 在探寻新的构造方法、提升码的纠错性能等方面尚存有一定的研究空间.

LDPC 卷积码的校验矩阵是半无限长的, 导致其迭代译码比 LDPC 分组码更为复杂. 文献[1]针对信息位 b

=1 的 LDPC 卷积码设计了一种由 l 个处理器级联而成的流水线译码器 (Pipeline Decoder, PD), 其复杂度随卷积码的约束度 l 和处理器个数 l 呈线性增长, 缺点是寄存单元多、译码延时长. 文献[13, 14]将流水线译码器的 l 个分离的处理器合为一体, 实现了 LDPC 卷积码的窗译码 (Windowed Decoding, WD), 简化了译码结构, 降低了译码开销, 文献[8]则以一种大信息块的原模图 LDPC 卷积码为基础, 进一步完善了窗译码. 除迭代产生的运算量外, 寄存器开销和译码延时也是衡量 LDPC 卷积码实际应用价值的两个重要指标, 在 Viterbi 译码中, 二者仅为卷积码约束度的 5~10 倍^[15], 然而流水线译码和窗译码与之相比却存有明显差距, 需更多的研究予以改进.

基于以上分析, 本文着眼于第一类随机 LDPC 卷积码, 通过理论分析, 导出实现系统编码以及获得好码所应具备的条件, 并以此为基础, 对 Gallager LDPC 分组码的校验矩阵实施“列置换-置数-剪切-合并”, 提出一种新的准随机 LDPC 卷积码的构造方法. 之后, 基于系数修正的最小和算法 (Min-Sum Algorithm, MSA)^[16], 提出一种窗扇尺寸固定、滑动步长可选的窗译码算法, 并与流水线译码进行了对比. 第 2、3 节分别讨论了准随机 LDPC 卷积码的构造和窗译码的执行; 第 4 节通过仿真实验, 界定了窗扇滑动步长的取值范围, 验证了构造方案和窗译码的有效性, 完成了不同约束度的准随机 LDPC 卷积码的误码性能测试.

2 准随机 LDPC 卷积码的构造

2.1 LDPC 卷积码的定义

LDPC 卷积码由半无限长稀疏校验矩阵定义^[1~5, 8, 9], 本文拟构造一种信息位长度 $b=1$ 、码字长度 $c=2$ 、码率 $R=1/2$ 的 LDPC 卷积码, 设对应的半无限长校验矩阵为:

$$\mathbf{H} = \begin{bmatrix} h_1^0(0) & h_1^1(1) & h_1^2(2) & \cdots & h_1^l(1) \\ h_2^0(0) & h_2^1(1) & h_2^2(2) & \cdots & h_2^l(1) \\ h_1^0(1) & h_1^1(2) & \cdots & h_1^{l-1}(l) & h_1^l(l+1) \\ h_2^0(1) & h_2^1(2) & \cdots & h_2^{l-1}(l) & h_2^l(l+1) \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots \\ & & & h_1^0(l) & h_1^1(l+1) & h_1^2(l+2) & \cdots \\ & & & h_2^0(l) & h_2^1(l+1) & h_2^2(l+2) & \cdots \\ & & & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \quad (1)$$

其中, 元素 $h_n^m(t)$ 的上标 $m=0 \sim l$, l 指卷积码的编码约束度; 下标 $n=1, 2$, 分别对应 2 位码元; $t=0 \sim \infty$, 表示时间. 该矩阵从 l 时刻, 即第 $l+1$ 列开始, 每一列包含了 $2(l+1)$ 个元素, 可根据其中的 1 元素建立相应的校验方程, 校验方程是 LDPC 卷积码能够实施置信传播迭代译码的关键. 取式(1)中第 $2\tau+1 \sim 2\tau+2$ 行和 $\tau+1 \sim \tau+l+1$ 列, $\tau=0 \sim \infty$, 得到子矩阵:

$$\mathbf{H}(\tau) = \begin{bmatrix} h_1^0(\tau) & h_1^1(\tau+1) & h_1^2(\tau+2) & \cdots & h_1^l(\tau+l) \\ h_2^0(\tau) & h_2^1(\tau+1) & h_2^2(\tau+2) & \cdots & h_2^l(\tau+l) \end{bmatrix} \quad (2)$$

其规格为 $2 \times (l+1)$. 设 T 为满足“对于任意 τ 均有 $\mathbf{H}(\tau) = \mathbf{H}(\tau+T)$ ”这一条件的最小正整数. 可以看出, 若 $T=1$, 式(1)的各列相同, 校验矩阵定义的是常规 $(2, 1, l)$ 卷积码; 若 $T>1$, 式(1)的连续 T 列不相同, 则定义的是周期为 T 的时变卷积码, 进一步, 若校验矩阵是稀疏矩阵, 即可获得时变周期 LDPC 卷积码. 本文讨论的是规则 LDPC 卷积码, 若式(1)的行、列重分别为 γ, ρ , 以下称之为 (l, γ, ρ) LDPC 卷积码, 其中 γ, ρ 的位置与 LDPC 分组码相反, 以示区别.

2.2 LDPC 卷积码的构造基础分析

首先讨论 l 时刻的编码, 设式(1)定义的 (l, γ, ρ) LDPC 卷积码在 $0 \sim l$ 时刻的信息序列为:

$$\mathbf{M}^{[0, l]} = [m^0, m^1, m^2, \cdots, m^l] \quad (3)$$

对应的编码序列为:

$$\mathbf{C}^{[0, l]} = [c_1^0 c_2^0, c_1^1 c_2^1, \cdots, c_1^l c_2^l] \quad (4)$$

其中, $c_1^0 c_2^0, c_1^1 c_2^1, \cdots, c_1^{l-1} c_2^{l-1}$ 为之前的已知码字, $c_1^l c_2^l$ 为待编码字. 由校验矩阵的定义可知, 式(1)的第 $l+1$ 列与式(4)满足如下校验方程:

$$\begin{aligned} \mathbf{C}^{[0, l]} \mathbf{H}_l &= \sum_{i=0}^l (c_1^{l-i} h_1^i(l) + c_2^{l-i} h_2^i(l)) \\ &= c_1^l h_1^0(l) + c_2^l h_2^0(l) + \sum_{i=1}^l (c_1^{l-i} h_1^i(l) + c_2^{l-i} h_2^i(l)) \\ &= 0 \end{aligned} \quad (5)$$

显然, 仅由该式并不能计算 c_1^l, c_2^l 两个未知码元, 但如果考虑采用系统码, 不妨设 $c_1^l = m^l$, 将其带入式(5)并移项, 可得

$$c_2^l h_2^0(l) = m^l h_1^0(l) + \sum_{i=1}^l (c_1^{l-i} h_1^i(l) + c_2^{l-i} h_2^i(l)) \quad (6)$$

该式右边两项为已知数, 容易看出, 若采用系统编码, 只需增加一个必要条件 $h_2^0(l) = 1$, 即可根据式(6)计算获得 c_1^l, c_2^l . 系统编码的好处还在于: 可直接依托稀疏的校验方程进行编码, 极大降低了计算复杂度; 译码器可直接对变量节点进行硬判决, 有利于执行置信传播迭代译码. 目前 $R=1/2$ 的 LDPC 卷积码多采用这种类似于系统反馈卷积码^[17]的编码策略.

LDPC 卷积码的编码约束度 l 越大, 纠错能力就越强. 然而, 其校验矩阵的稀疏性和时变性导致各个时刻的编码约束度参差不齐, 势必影响到纠错性能. 在式(6)中, $m^l = c_1^l$ 表征最新的输入信息, c_2^0 表征最旧的编码码元, 若令对应的系数 $h_2^l(l) = 1, h_1^0(l) = 1$, 那么二者就能够同时参与到 c_2^l 的编码中, 实现编码约束度的最大化, 这一点将有利于构造更好的码型.

以此类推, 在任意 τ 时刻, 实现系统编码的条件

(1) 基础矩阵. 按照 Gallager 随机构造规则^[18], 获取 $R = 1/2$ 的 $(24, 4, 2)$ LDPC 分组码的校验矩阵, 其规格为 12×24 . 该矩阵的一些行存在 4 个连 1, 不利于第(2)步, 可通过行、列随机交换予以消除, 设得到的矩阵为 H_g , 如图 1(a) 所示;

(2) 列置换. 对 H_g 实施列置换, 使每一行指定位置的连续 4 个元素为 0100 (最后一行的首尾视为连续), 如图 1(b) 粗体所示. 可以看出, 该置换目标实质上等价于“奇数列指定位置为 00, 偶数列指定位置为 01”, 可通过编程实现. 具体过程为: 首先从左到右逐次置换偶数列, 使得指定位置为 01, 例如第 4 列, 通过扫描, 发现 H_g 的第 3 列元素 $H_g(1, 3)$ 和 $H_g(2, 3)$ 分别为 0 和 1, 则将 3 列与 4 列相互交换位置; 奇数列的置换须扫描 00, 具体操作与偶数列类似. 很明显, 已置换的列不能参与到后面的扫描和置换, 因此可供扫描的数量会逐渐减少, 扫描失败的可能性会逐渐增大. 但是, 这里采用了先置换偶数列, 再置换奇数列的策略, 当稀疏矩阵 H_g 的尺寸足够大时, 0 的数量远多于 1, 即使最后扫描范围越来越窄, 也能够以很高的概率找到 00. 若扫描失败, 可返回第(1)步重新开始;

(3) 置数. 对图 1(b) 中 0100 的首尾两个元素取反, 置为 1101, 如图 1(c) 粗体所示. 置数后, 原矩阵的行、列重分别增加 2 和 1, 其结果实质为新的 $(24, 6, 3)$ LDPC 分组码. 本环节可能会导致短环, 须进行四环检测. 若检测发现四环, 可返回第(1)步重新开始;

(4) 剪切. 文献[1]按斜率 1 对校验矩阵剪切, 须先隔行、再两边插入固定序列, 才能得到 $1/2$ 码率的半无限长校验矩阵. 本方案无插入操作, 直接以 $1/2$ 的斜率进行剪切, 如图 1(d) 虚线所示;

(5) 合并. 横向合并, 将图 1(d) 中剪切得到的暗色部分的下三角阵向右平移, 与上三角阵并接, 得到如图 1(e) 所示的矩阵 $H_{[1, 12]}$, 该矩阵的第 1 行对应式(7)的第 l 列, 以此类推, 一共 12 行对应式(7)的 11 ~ 22 列; 纵向合并, 向右下方重复并接 $H_{[1, 12]}$, 即可得到周期性半无限长稀疏校验矩阵.

上述过程是按式(7)的转置矩阵处理的, 这是为了便于图 1 的绘制. 基础矩阵的生成和列置换两个环节都充分体现了随机编码, 但受限于式(7), 并未完全随机化, 因而称之为“准随机 LDPC 卷积码”. 一旦 Gallager LDPC 分组码的码长 N 确定, 准随机 LDPC 卷积码的编码约束度 l 和周期 T 也随之确定, 三者的关系为

$$l = T - 1 = N/2 - 1 \quad (8)$$

从中也可以看出 $T = N/2$. 例如图 1 中, $N = 24$, 那么 l 和 T 分别等于 11 和 12.

2.4 准随机 LDPC 卷积码的构造复杂度分析

本文提供的构造方案, 生成矩阵借用了 Gallager

LDPC 分组码, 有较为成熟的程序可借鉴^[18]; 置数、剪切、合并环节操作简单, 只需很小的开销; 列置换需要逐列反复多次扫描、置换, 且存有扫描失败、重头再来的风险, 构造的复杂度主要集中在该环节. 事实上, 尚若使用 $(24, 6, 3)$ 替换 $(24, 4, 2)$ LDPC 分组码, 并设置置换目标为“1101”, 则无需“置数”也可得到相同结果, 但低密度校验矩阵中 1 的数量很少, 这将会使扫描失败的可能性大增, 陷入多次循环. 反置置换目标“0010”, 辅以先置换偶数列、再置换奇数列的策略, 实现难度则要小得多. 实验也充分表明, 本方案采用的变通措施的确是合理、可行、高效的. 另外, 本文使用计算机编程, 提前构造获得准随机 LDPC 卷积码的校验矩阵, 仿真中直接调用, 其本身并未给编、译码器的复杂度带来影响.

3 准随机 LDPC 卷积码的窗译码

3.1 置信传播迭代译码算法

早期的置信传播多采用和积算法 (Sum-Product Algorithm, SPA)^[19], 之后有学者提出运算量低得多的最小和算法^[20], 文献[1]在流水线译码器中使用了这种算法, 以

$$r_j^i(\tau) = r(\tau) + \sum_{j'=1, j' \neq j}^{\gamma} \text{sign}\left(\prod_{\tau' \in C_j(\tau)} r_{j'}^{i-1}(\tau')\right) \min_{\tau' \in C_j(\tau)} |r_{j'}^{i-1}(\tau')| \quad (9)$$

完成变量节点 $r_j(\tau)$ 的更新. 其中, 第一项是译码器的接收软信息, 即内信息, 第二项为计算量很低的“符号-最小”运算, 即外信息; $j = 1, 2, \dots, \gamma$, γ 指 (l, γ, ρ) LDPC 卷积码校验矩阵的行重; $r_j^i(\tau)$ 和 $r_{j'}^{i-1}(\tau')$ 分别表示新、旧变量节点; $\tau' \in C_j(\tau)$ 表示第 j' 个校验方程中, 除 $r_j^i(\tau)$ 自身外其余变量节点的位置. 考虑到本文构造的 $(l, 3, 6)$ LDPC 卷积码, 其 $\gamma = 3, \rho = 6$, 那么每次迭代需要分别计算 $r_1^i(\tau)$ 、 $r_2^i(\tau)$ 、 $r_3^i(\tau)$ 三个变量节点, 每个变量节点的计算需对另外 5 个变量节点寻址.

相较于和积算法, 最小和算法存有较大的性能损失, 文献[21]提出一种改进的系数修正算法 (也看文献[16, 18]), 以微弱的代价获得了明显的性能改善, 文献[22]在流水线译码中采用了这种修正算法, 本文则将其移植到窗译码, 以

$$r_j^i(\tau) = r(\tau) + \alpha \times \sum_{j'=1, j' \neq j}^3 \text{sign}\left(\prod_{\tau' \in C_j(\tau)} r_{j'}^{i-1}(\tau')\right) \min_{\tau' \in C_j(\tau)} |r_{j'}^{i-1}(\tau')| \quad (10)$$

完成变量节点 $r_j(\tau)$ 的更新, 其中 α 为修正系数, $\alpha = 0.8$.

3.2 窗译码过程描述

设两位码元 c'_1, c'_2 经由高斯信道后, 接收端获得的软信息为:

$$r(2t) = c_1^t + n(2t), r(2t + 1) = c_2^t + n(2t + 1) \quad (11)$$

其中时间 $t = 0 \sim \infty$, c_1^t 、 c_2^t 是双极性码, $n(2t)$ 、 $n(2t + 1)$ 为高斯噪声. 为了实现窗译码, 须对半无限长校验矩阵进行截取, 设截取时间为 $t = 0 \sim w_f - 1$, 相应的码字序列为:

$$\mathbf{C}^{[0, w_f - 1]} = [c_1^0 c_2^0, c_1^1 c_2^1, \dots, c_1^{w_f - 1} c_2^{w_f - 1}] \quad (12)$$

其中 w_f 被定义为窗框尺寸, 是大于周期 T 的正整数.

由式(11)可知, 式(12)所对应的接收软信息序列为:

$$\begin{aligned} \mathbf{R}^{[0, w_f - 1]} \\ = [r(0)r(1), r(2)r(3), \dots, r(2w_f - 2)r(2w_f - 1)] \end{aligned} \quad (13)$$

现以(11,3,6)准随机 LDPC 卷积码为例, 按时间 $t = 0 \sim w_f - 1$ 截取即可获得图 2 所示的窗译码示意图. 其中, 实线框所示的有限长校验矩阵被定义为“窗框”, 其规格为 $w_f \times 2w_f$; 虚线框所示的部分(临近部分有重叠)被定义为“窗扇”, 其规格为 $w_c \times 2w_c$; 相邻窗扇的垂直间距被定义为窗扇的“滑动步长” Δ , $1 \leq \Delta \leq w_c$; 窗扇数量 n 也被称为更新次数.

文献[8, 14]针对大信息长度、小周期的原模图 LDPC 卷积码, 提出一种窗扇尺寸 w_c 可选、滑动步长 Δ 固定的窗译码算法. 本文则针对信息长度 $b = 1$ 、周期 $T \gg 1$ 的准随机 LDPC 卷积码, 提出窗扇尺寸 w_c 固定、滑动步长 Δ 可选的窗译码算法. 该算法的窗扇尺寸 w_c 被设定为一个周期, 此时图 2 暗色部分对应的两个码元

正好可以在窗扇界内各获取 3 个校验方程, 为对应变量节点的更新提供了基础条件. 该算法的具体实施过程如下:

首先看窗扇 1, 译码器根据式(10)完成与码元 c_1^t 与 c_2^t 相对应的变量节点 $r_j(2l)$ 与 $r_j(2l + 1)$ 的更新, 图中, 粗体“1”标明了各校验方程另外五个变量节点的位置, 左下角对变量节点 $r_j(2l)$ 与 $r_j(2l + 1)$ ($j = 1, 2, 3$) 作了进一步标示; 接下来, 译码器滑动 Δ 个时间点, 完成窗扇 2 中 $c_1^{t+\Delta}$ 与 $c_2^{t+\Delta}$ 所对应的变量节点 $r_j(2l + 2\Delta)$ 与 $r_j(2l + 2\Delta + 1)$ 的更新; 以此类推, 直至滑动到窗扇 n , 完成 $c_1^{t+\Delta(n-1)}$ 与 $c_2^{t+\Delta(n-1)}$ 所对应的变量节点 $r_j(2l + 2n\Delta - 2\Delta)$ 与 $r_j(2l + 2n\Delta - 2\Delta + 1)$ 的更新, 至此完成一轮完整的迭代, 并计算

$$c = \frac{1}{2} (1 + \text{sign}(r(2w_f - 2) + \sum_{j=1}^3 r_j(2w_f - 2))) \quad (14)$$

作为硬判决译码输出.

下一时刻, 译码器接收到新的软信息, 校验矩阵逐行向下轮动, 译码器重复以上过程, 实施新一轮的迭代. 事实上, 每一轮迭代仅仅对图 2 暗色部分的变量节点进行更新, 换言之, $r_j(2l)$ 与 $r_j(2l + 1)$ 在窗扇 1 中获得第 1 次更新后, 须间隔 Δ 个时刻才能在窗扇 2 中获得第 2 次更新, 以此类推, 间隔 $\Delta(n - 1)$ 个时刻后在窗扇 n 中获得第 n 次、也是最后一次更新, 变量节点的可信度从左到右逐次得到增强.

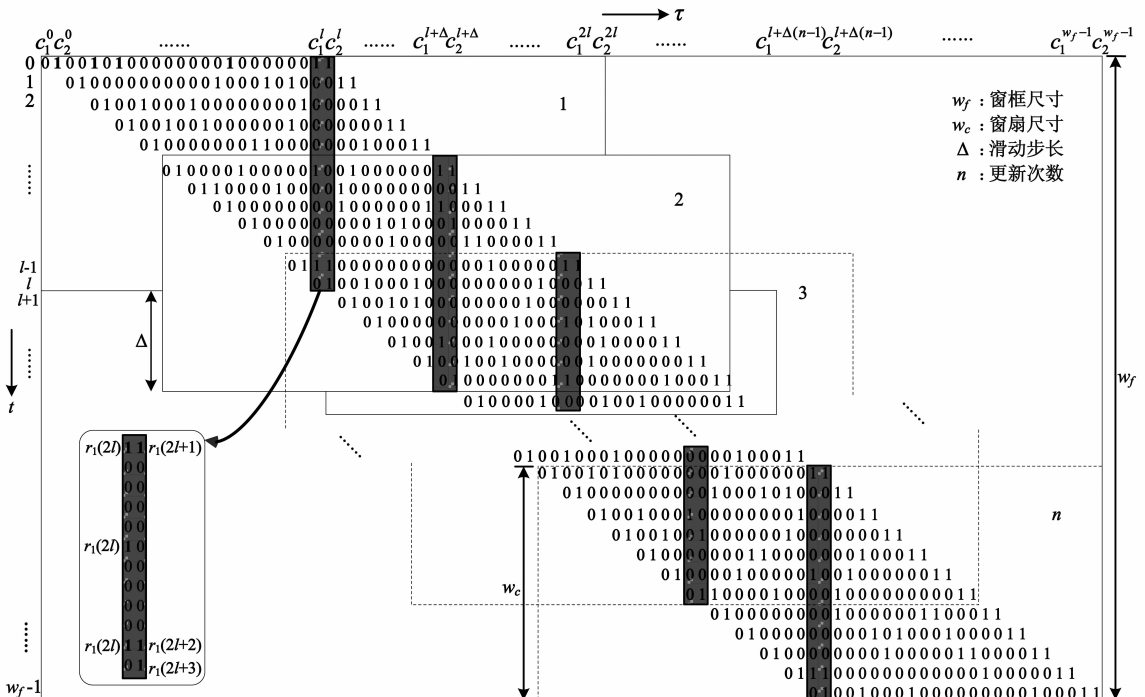


图 2 (11,3,6)准随机LDPC卷积码的窗译码过程

3.3 窗译码复杂度分析

当更新次数 n 相同时,窗译码和流水线译码针对式(10)所产生的迭代运算量并无太大区别,窗译码的主要优势体现在结构更为单一、简洁,具有更少的寄存器开销和译码延时.和流水线译码一样,窗译码也需要保存变量节点的更新结果,由于每个时间点含有 6 个变量节点和 2 个接收软信息,那么所需的寄存器单元数应为 $8w_f$. 同时也不难看出,窗译码的译码延时即为窗扇尺寸 w_f . 由图 2,窗框与滑动步长满足如下关系:

$$w_f = w_c + \Delta(n-1) = T + \Delta(n-1) \quad (15)$$

这意味着滑动步长 Δ 和更新次数 n 越小, w_f 就越小,进而寄存器开销和译码延也越小.在一轮迭代中,当 Δ 为最小值 1 时,除首尾三角阵外,其余部分均可获得更新,与 LDPC 分组码相似;当 Δ 为最大值 T 时,窗译码与流水线译码器本质上是等价的,设此时的窗框尺寸为 w_{\max} , 则有

$$w_{\max} = T + T(n-1) = Tn \quad (16)$$

鉴于 $n \gg 1$, 结合式(15)与(16)可知:

$$\begin{aligned} w_f &= \frac{T + \Delta(n-1)}{Tn} w_{\max} \\ &= \left(\frac{\Delta}{T} + \frac{1 - \Delta/T}{n} \right) w_{\max} \approx \frac{\Delta}{T} w_{\max} \quad (17) \end{aligned}$$

该式表明,窗译码的窗框尺寸仅为流水线译码的 Δ/T , 若选择足够小的 Δ ,窗译码器的寄存器开销和译码延时将会远低于流水线译码器.

4 仿真分析

准随机 LDPC 卷积码编、译码器的建模分别基于式(6)、式(10),以及前面所述规则,仿真置于 BPSK 调制及高斯信道环境.准随机 LDPC 卷积码的校验方程具有周期时变性,式(6)、式(10)所需的位置信息也同样如此,为了降低运算量,可借助计算机编程事先获得 T 组位置信息,仿真时按对应的时间节点循环调用即可.迭代运算采用双精度数据类型,视具体情况在出现 5000 ~ 20000 个错误比特时停止仿真.

本文提出的窗译码的优势是可以通过选择小的滑动步长 Δ 来降低寄存器开销和译码延时,但同时也会导致一定的性能损失.为了探寻性能与开销之间的平衡点,笔者针对本文构造得到的(47, 3, 6)、(95, 3, 6)、(191, 3, 6)和(383, 3, 6)等 4 种不同约束度的准随机 LDPC 卷积码进行了仿真.鉴于这些码型的周期 T 均为 24 的倍数,每种码型可按比例 $\Delta/T = 1/24, 2/24, 3/24, 4/24, 6/24, 8/24, 12/24, 24/24$ 分别选取 8 个 Δ 测试点,其中 $\Delta/T = 24/24 = 1$ 与流水线译码器等效.例如(47, 3, 6)选择 $\Delta = 2, 4, 6, 8, 12, 16, 24, 48$, 其他码型依次类推.同时,仿真也针对 4 种码型的差异,各自选取了恰当

的信噪比,使各 BER 曲线收敛于大体相同的位置,测试结果如图 3 所示.可以看出,当 Δ 较小时,LDPC 卷积码的误码性能明显较差,随着 Δ 的增大,BER 曲线逐渐收敛于流水线译码器.所幸的是,全部 BER 曲线均在 $\Delta/T \approx 0.2$ 附近出现了明显的拐点,显然,在拐点稍靠右的位置选择 Δ 是合理的,例如 $\Delta/T = 1/4$, 此时其误码性能与流水线译码器相差甚微,而根据式(17)可知其矢量寄存器开销和译码延时仅仅为流水线译码器的 1/4,可见付出的性能损失是非常值得的.图 3 为选取滑动步长 Δ 提供了重要依据.

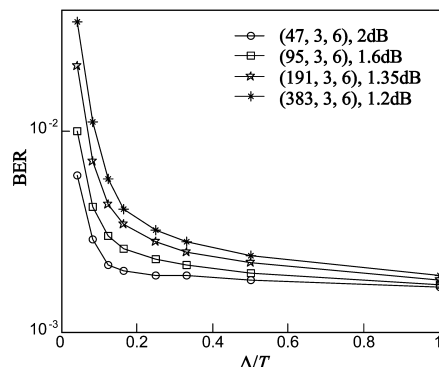


图3 窗扇的滑动步长对误码性能的影响

本文的(1, 3, 6)准随机 LDPC 卷积码与文献[1]的(1, 3(2), 5)LDPC 卷积码同属于第一类随机构造方法,且信息位长度 b 和码率 R 相同,具有较强的可比性.鉴于此,笔者就二者各自选取了(515, 3, 6)和(513, 3(2), 5)两款约束度大体相当的码型,并分别按最小和算法与系数修正最小和算法进行仿真和对比.窗译码的更新次数设置为 $n = 70$, 滑动步长设置为 $\Delta = T/3$, 仿真结果如图 4 所示.可以看出,相较于(1, 3(2), 5)LDPC 卷积码,(1, 3, 6)准随机 LDPC 卷积码存有大约 0.1dB 的优势;而相较于最小和算法,系数修正最小和算法可获得大约 0.35dB 的增益.进一步实验表明,其他的约束度也有类似结果.

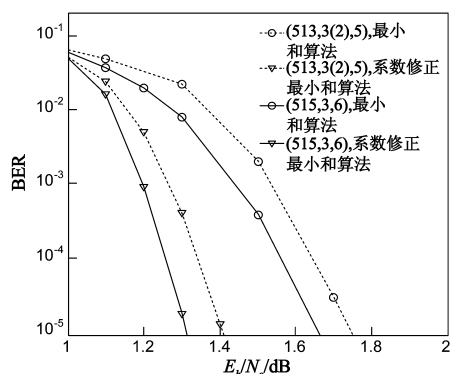


图4 两种码型及两种译码算法的对比

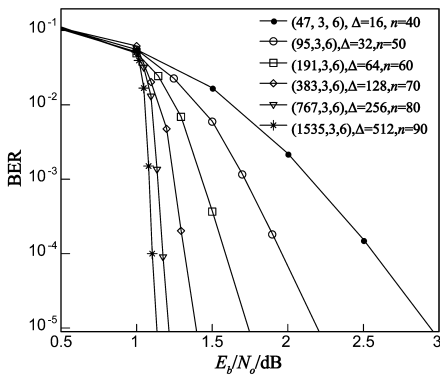


图5 部分准随机LDPC卷积码误比特率与信噪比关系曲线

基础矩阵 H_g 的生成与列的置换都具有随机性,这使得同样的编码约束度 l 也可以构造数量繁多的准随机 LDPC 卷积码,鉴于此,笔者针对 $l = 47, 95, 191, 383, 767, 1535$, 通过大量的误码测试择优选取了 6 款码型,以全面考察本文所构造的准随机 LDPC 卷积码的纠错能力.为确保窗译码的纠错性能,滑动步长 Δ 被设置为 $T/3$,迭代次数 n 则随 l 作适度增加,仿真结果如图 5 所示.可以看出,准随机 LDPC 卷积码的误码性能随着 l 的增加而改善,在 l 不到 10^3 量级时即获得了很好的瀑布效应,在 $l = 1535$ 时 BER 曲线距离香农限仅约 1dB.

5 总结

LDPC 卷积码既适合于置信传播迭代译码,又具有记忆特性所带来的高性价比.本文通过对 LDPC 卷积码的校验矩阵进行代数解析,给出了实现系统编码以及获得好码的基础条件,提出了一种新的构造方案.该方案具有较强的通用性,除 Gallager LDPC 分组码以外,包括非规则 LDPC 等在内的其他码型原则上也是适用的,这为构造新型的 LDPC 卷积码提供了更为广泛的途径,笔者也将就此问题展开新的研究.窗译码是一种比流水线译码更为简洁高效的迭代译码,本文提出一种窗扇尺寸固定、滑动步长可选的窗译码,通过选择小的滑动步长,在确保性能的前提下大幅降低了寄存器开销和译码延时;将系数修正最小和算法植入这种窗译码,可进一步改善其纠错性能.

参考文献

[1] Jimenez Felstrom A, Zigangirov K S. Time-varying periodic convolutional codes with low-density parity-check matrix[J]. IEEE Transactions on Information Theory, 1999, 45(6): 2181 - 2191.

[2] Tanner R M, Sridhara D, et al. LDPC block and convolutional codes based on circulant matrices[J]. IEEE Transactions on Information Theory, 2004, 50(12): 2966 - 2984.

[3] Pusane A E, Smarandache R, et al. Deriving good LDPC convolutional codes from LDPC block codes[J]. IEEE Transactions on Information Theory, 2011, 57(2): 835 - 857.

[4] Liwei Mu, Xingcheng Liu, Chulong Liang. Construction of binary LDPC convolutional codes based on finite fields[J]. IEEE Communications Letters, 2012, 16(6): 897 - 900.

[5] Chiu-Wing Sham, Xu Chen, Lau F C M, Yue Zhao, Tam W M. A 2.0 Gb/s throughput decoder for QC-LDPC convolutional codes[J]. IEEE Transactions on Circuits and Systems I, 2013, 60(7): 1857 - 1869.

[6] 张国华,王新梅.围长至少为 8 的 QC-LDPC 码的新构造:一种显式框架[J].电子学报,2012,40(2):331 - 337.

Zang Guohua, Wang Xinmei. Novel constructions of QC-LDPC codes with girth at least eight: an explicit framework[J]. Acta Electronica Sinica, 2012, 40(2): 331 - 337. (in Chinese)

[7] Massey J, Costello D, Justesen J. Polynomial weights and code constructions[J]. IEEE Transactions on Information Theory, 1973, 19(1): 101 - 110.

[8] Iyengar A R, Papaleo M, et al. Windowed decoding of protograph-based LDPC convolutional codes over erasure channels[J]. IEEE Transactions on Information Theory, 2012, 58(4): 2303 - 2320.

[9] Mitchell D G M, Pusane A E, Costello D J. Minimum distance and trapping set analysis of protograph-based LDPC convolutional codes[J]. IEEE Transactions on Information Theory, 2013, 59(1): 254 - 281.

[10] Hosung Park, Seokbeom Hong, Jong-Seon No, Dong-Joon Shin. Design of multiple-edge protographs for QC LDPC codes avoiding short inevitable cycles[J]. IEEE Transactions on Information Theory, 2013, 59(7): 4598 - 4614.

[11] Lee P J. There are many good periodically time-varying convolutional codes[J]. IEEE Transactions on Information Theory, 1989, 35(2): 460 - 463.

[12] Bates S, Zhengang Chen, Xiaodai Dong. Low-density parity check convolutional codes for Ethernet networks[A]. Proceedings of IEEE Pacific Rim Conference on Communications, Computers and signal Processing[C]. Victoria, BC, Canada: IEEE, 2005. 85 - 88.

[13] Pusane A E, Feltstrom A J, et al. Aspects of LDPC convolutional codes[J]. IEEE Transactions on Communications, 2008, 56(7): 1060 - 1069.

[14] Papaleo M, Iyengar A R, et al. Windowed erasure decoding of LDPC convolutional codes[A]. IEEE Information Theory Workshop[C]. USA: IEEE, 2010. 1 - 5.

[15] 彭万权,张承畅.多阶幻方卷积码的构造及译码[J].电子学报,2013,41(1):123 - 130.

Peng Wanquan, Zhang Chengchang. Construction and decoding of high order magic square convolutional codes[J]. Acta Electronica Sinica, 2013, 41(1): 123 - 130. (in Chinese)

- [16] 吴湛击, 王文博. 现代纠错编码与调制理论及应用[M]. 北京: 人民邮电出版社, 2008. 232 – 238.
Wu Zhanji, Wang Wenbo. Modern Channel Coding and Modulation: Theory and Application[M]. Beijing: Posts & Telecom Press, 2008. 232 – 238. (in Chinese)
- [17] Berrou C, Glavieux A. Near optimum error correcting coding and decoding: turbo-codes[J]. IEEE Transactions on Communications, 1996, 44(10): 1261 – 1271.
- [18] 肖扬. Turbo 与 LDPC 编解码及其应用[M]. 北京: 人民邮电出版社, 2010. 19 – 21.
Xiao Yang. Turbo and LDPC: Coding, Decoding and Application[M]. Beijing: Post & Telecom Press, 2010. 19 – 21. (in Chinese)
- [19] MacKay D J C. Good error-correcting codes based on very sparse matrices[J]. IEEE Transactions on Information Theory, 1999, 45(2): 399 – 431.
- [20] Fossorier M P C, Mihaljevic M, Imai H. Reduced complexity iterative decoding of low density parity check codes based on belief propagation[J]. IEEE Transactions on Communications, 1999, 47(5): 673 – 680.
- [21] Jinghu Chen, A Dholakia, E Eleftheriou, M P C Fossorier, Xiaoyu Hu. Reduced-complexity decoding of LDPC codes[J]. IEEE Transactions on Communications, 2005, 53(8): 1288 – 1299.
- [22] 刘原华, 王新梅, 等. 一种改进的卷积 LDPC 码置信传播译码算法[J]. 西安电子科技大学学报, 2009, 36(3): 424 – 428.
Liu Yuanhua, Wang Xinmei, et al. Improved belief propagation algorithm for decoding of convolutional LDPC codes[J]. Journal of Xidian University, 2009, 36(3): 424 – 428. (in Chinese)

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



伍小兵 男, 1968 年生于重庆开县, 2008 年在重庆大学电气工程学院获硕士学位, 现为重庆工程职业技术学院教授, 主要从事电工理论与新技术研究工作.

E-mail: wxb@cqvie.edu.cn