

用于实现并行行列联合译码的 QC-LDPC 码构造方法^①董明科^② 王 达 郑雅丹 项海格^③

(北京大学信息科学技术学院 区域光纤通信网与新型光通信系统国家重点实验室 北京 100871)

摘 要 针对普通低密度校验(LDPC)码制约行列联合(JRC)译码算法并行度提高的问题,基于块渐进边增长(BPEG)算法,提出了一种用于并行 JRC 译码的 LDPC 码构造方法。该方法构造的准循环 LDPC 码(QC-LDPC)基矩阵由含 r (r 为大于 1 的整数)行的行组构成,允许一个行组内的 r 行进行并行 JRC 运算。仿真结果表明,用上述构造方法构造的 LDPC 码与 BPEG 码的误码性能相当。硬件实现表明,用此构造码的并行译码器的速率能达到典型传统准循环译码器的 3 倍以上,为面向译码器的 LDPC 码构造提供了范例。

关键词 低密度校验(LDPC)码,行列联合(JRC)译码,渐进边增长(PEG)算法,块渐进边增长(BPEG)算法,准循环 LDPC(QC-LDPC)码,面向译码器的 LDPC 码构造方法

0 引言

常用的低密度校验(low-density parity-check, LDPC)码译码方法是置信度传播算法^[1],该类算法中,最小和(min-sum, MS)算法^[2]及其改进算法^[3,4]由于运算简便、性能良好而为译码器硬件实现所常用。LDPC 码分层译码的置信度传播规划比传统规划传播速度快,典型迭代次数能减少一半^[5]。行列联合(joint row-column, JRC)译码方法^[6]使用 MS 运算和行分层置信度传播方式,运算简单且迭代次数少,为近年的 LDPC 译码器方案所采纳,但是文献[7,8]一般都注重分层译码器的结构设计,缺乏针对 JRC 译码的 LDPC 码构造研究。LDPC 码实用码型多采用准循环(quasi-cyclic LDPC, QC-LDPC)码^[9]结构,很多通信标准^[10,11]都采用 QC-LDPC 码或其等效变换形式^[12]。QC-LDPC 码便于译码器中的存储器、运算器和连线的整体设计,也便于规划译码器的资源和速率关系^[12]。JRC 并行译码算法应采用 QC-LDPC 码以带来实现优势。

高速译码器需要并行译码结构^[9,13],所采用的码要充分支持并行译码。为进行并行 JRC 译码,译码器需对校验矩阵中 r (r 为大于 1 的整数)行同时实施置信度传播计算。但普通 LDPC 码的校验矩阵

结构对此或不支持,或不充分支持,这阻碍了高速译码器的实现,所以本研究旨在构造面向 JRC 并行译码所需的 QC-LDPC 码。设计 JRC 并行算法码需用合适的构造方法^[14,15],而渐进边增长(progressive edge-growth, PEG^[16])算法是重要的一种。块渐进边增长(Block-PEG, BPEG^[17])构造算法推广了 PEG 算法,可以构造 QC-LDPC 码,所构造码的码长也可灵活选取。BPEG 构造算法兼具 PEG 算法带来的大环长特性和准循环码的规则性,所构造的码兼顾性能和可实现性。近年文献[18,19]对 PEG 算法的改进大多都是改善二分图围长或者环长分布,可获得约 0.1 ~ 0.3 dB 的编码增益,但 PEG 类算法文献中缺乏优化 LDPC 码校验矩阵并行度方面的研究。根据以上情况,本文提出基于 BPEG 的 QC-LDPC 码构造算法,以构造并行度高而且性能优良的 QC-LDPC 码。为并行 JRC 译码构造的 LDPC 码基矩阵由含 r 行的行组构成,以保证行组内各行能同时参加译码运算。

1 适于并行译码的校验矩阵结构分析

JRC 译码方法^[6]是基于 MS 运算的分层高效译码算法,下面先简析 JRC 算法流程。

① 国家自然科学基金(61071083,60972008)和国防基金(9140A220310)资助项目。

② 男,1973 年生,博士生;研究方向:通信中的信号处理;E-mail: mingke.dong@pku.edu.cn

③ 通讯作者, E-mail: xianghg@pku.edu.cn

(收稿日期:2011-03-18)

设 $H_{m \times n}$ 是 LDPC 校验矩阵, 首先初始化: $U_i = u_{i0}, 1 \leq i \leq n; u_{ij} = 0, i \in N(j), 1 \leq j \leq m$ 。上式中 i, j, n, m 为正整数, U_i 表示变量节点 i 发出的外信息的和。 u_{ij} 是 $+1, -1$ 符号经过加性高斯白噪声信道后接收到的信道信息概率对数似然比, u_{ij} 表示校验节点 j 传向变量节点 i 的外信息。 $N(j)$ 为校验节点 j 相邻的变量节点集合。

然后逐行进行置信度传播运算, 对校验矩阵第 $j(1 \leq j \leq m)$ 行进行以下 3 式的运算(简称 JRC 运算):

$$v_{ij} = U_i - u_{ji}, i \in N(j) \quad (1)$$

$$u_{ji} = \left(\prod_{k \in N(j)/i} \text{sgn}(v_{kj}) \right) \min_{k \in N(j)/i} |v_{kj}|, i \in N(j) \quad (2)$$

$$U_i = v_{ij} + u_{ji}, i \in N(j) \quad (3)$$

其中 $N(j)/i$ 是 $N(j)$ 集合排除掉 i 所得的集合。 $\text{Sgn}(\cdot)$ 表示求符号运算, $\min(\cdot)$ 表示对集合求最小运算。式(2)也可使用改进运算方式^[3,4]。每进行完所有 m 行对应的 JRC 运算就完成一次迭代处理, 就可对所有 U_i 进行判决并计算校验式, 如果满足校验矩阵约束, 则停止迭代。否则继续迭代, 直到满足校验关系或者迭代次数达到预设值。

JRC 译码算法逐行进行 JRC 运算, 每行有关的运算包括式(2)表示的行方向置信度传播运算, 和式(1)、(3)表示的本行有关的列方向置信度传播运算。

然而, JRC 高速硬件译码都需要实施多行并行的 JRC 运算, 普通 LDPC 码却不支持。图 1 表示一个典型的 $1/2$ 码率 LDPC 系统码校验矩阵 $H_{14 \times 28}$ 。 H 的左半部分有双对角线结构, H 中的黑色的方块表示非 0 元素, 白色的表示 0。

对图 1 中的行组 $\{1, 2\}$, 参看表 1, 第 1 行 JRC 运算用的 U_i 变量和第 2 行 JRC 运算用的 U_i 变量有交集 $\{U_1, U_{27}\}$ 。行组中两个行的运算都要对 U_1, U_{27} 值更新, 所以 1、2 两行的并行 JRC 运算有写操作冲突, 不能并行进行。这是因为行组 $\{1, 2\}$ 中的两行在相同列有非 0 元素。但考虑 $\{1, 3\}$ 行组, 参见表 2, 行组内两行的 JRC 运算要刷新的 U_i 值都不相同, 即该行组形成的子矩阵的所有列的非 0 元素的个数至多为 1, 故能并行运算。普通矩阵不能有效支持并行 JRC 运算, 要支持 r 行的并行 JRC 运算, 只有构造多个像表 2 中的 r 行行组构成的校验矩阵, 且要求各个行组内每列最多只能有一个非 0 元素, 即要求校验矩阵各列的非 0 元素处于不同行组。

	1	2	3	...	24	25	26	27	28
1				...					
2				...					
3				...					
4				...					
...	...	...	...	...	...	...	...	...	...
13				...					
14				...					

图 1 LDPC 码校验矩阵 H 的示意图

表 1 行组 $\{1, 2\}$ 并行 JRC 运算用的 U_i, U_{27} 访问冲突

列号	1	2	3	...	24	25	26	27	28
矩阵 H 第 1 行	1	0	0	...	1	0	0	1	0
矩阵 H 第 2 行	1	1	0	...	0	0	0	1	0
第 1 行运算 U_i 值	U_1				U_{24}			U_{27}	
第 2 行运算 U_i 值	U_1	U_2						U_{27}	

表 2 行组 $\{1, 3\}$ 并行 JRC 运算用的 U_i 值无冲突

列号	1	2	3	...	24	25	26	27	28
矩阵 H 第 1 行	1	0	0	...	1	0	0	1	0
矩阵 H 第 3 行	0	1	1	...	0	1	0	0	1
第 1 行运算 U_i 值	U_1				U_{24}			U_{27}	
第 3 行运算 U_i 值		U_2	U_3			U_{25}			U_{28}

以图 1 为基矩阵, 扩充成 QC-LDPC 码的具体方式是: 在黑格处填充 $p \times p$ 的单位循环移位子矩阵, 在白格处填充 $p \times p$ 的 0 矩阵。每个非 0 的 $p \times p$ 子矩阵循环偏移量取从 0 到 $p-1$ 的整数值。每个单位循环阵内列重量为 1, 若图 1 都由表 2 所示的 r 行行组构成, 基矩阵每个行组对应的所有 $r \times p$ 行的非 0 元素的位置没有重叠, 那么所得的准循环 LDPC 校验阵的 JRC 译码并行度是普通 LDPC 校验阵的 $r \times p$ 倍, 也是普通准循环 LDPC 码的 r 倍。

综上所述, JRC 并行译码法需要 QC-LDPC 码的基矩阵为 r 个行的行组组成, 且基矩阵每个列的各非 0 元素处于不同行组。

2 适于并行译码的码构造方法——JRC-BPEG 算法

本文基于 BPEG 算法^[17] 提出适于 JRC 并行译码器的新码构造算法(此算法涉及 JRC 译码, 又用到 BPEG 方法, 故下文简记为 JRC-BPEG 算法)。BPEG 构造方法用 PEG 算法^[14] 构造 QC-LDPC 基矩

阵,但构造过程结合 QC-LDPC 码基矩阵中的循环偏移量计算。上一节已分析出 JRC 并行译码对 QC-LDPC 码基矩阵的要求,从图论的角度看,此要求就是要构造的二分图的校验节点由 r 个一组的校验节点集合(记为 C_{r_j}) 组成,而和每个变量节点连接的校验节点需处于不同的 C_{r_i} 集合。

下面给出新算法(JRC-BPEG)的完整流程。

(1) 初始化:

a) 初始化参数:基矩阵 H 的大小、码长(变量节点个数) n 、校验节点个数 m 。循环偏移单位矩阵(Block)的大小为 $p \times p$ 。

b) 利用密度推演^[20]等方法确定变量节点的优化的维度分布。

c) 将整个 LDPC 码二分图的校验节点按照并行运算要求, 规划成 m/r 个含 r 个校验节点的 C_{rj} 集合。

(2) 初始化基于 Block 的二分图, 也即基矩阵的二分图: 向基于 Block 的二分图中添加 m 个校验节点。

(3) 构造:向基于 Block 的二分图中逐个添加变量节点,挑选校验节点建立连接,并确定边线权重:

a) 以当前变量节点为根节点, 将当前基于 Block 的二分图展为树状图^[14], 参见图 2 (图中圆圈表示变量节点, 方框表示校验节点)。

b) 挑选出距根节点最远的校验节点集合。

c) 从所得集合中选出和当前变量节点相邻的校验节点不处于同一个 C_{r_j} 集合的校验节点, 组成集合。在所组成集合中再选择维度最小的校验节点加到当前变量节点的相邻节点集合中。

d) 遍历根节点到目的节点的所有路径,计算路径权重累加值^[17];

$$s = \left(\sum_{k=0}^{2L-2} (-1)^k p(i_k, j_k) \right) \bmod p \quad (4)$$

式中 $2L$ 是路径环长, $p(i_k, j_k)$ 是路径上已确定的边线权重值。然后按照环长最大化原则或者随机原则选取边线权重 (单位循环矩阵偏移量) ^[17]。

e) 判断根节点的维度是否已经满足初始维度分布,如果已经满足,则进入步骤 f), 否则返回步骤 a)。

f) 判断是否所有变量节点已经被添加进二分图,如果满足则进入步骤(4),否则返回步骤(3)。

(4) 根据构造出的带有偏移量的二分图, 可得到循环偏移量基础矩阵, 填充相应偏移量的单位循

环移位矩阵和全 0 矩阵,就得到最终的 QC-LDPC 码。

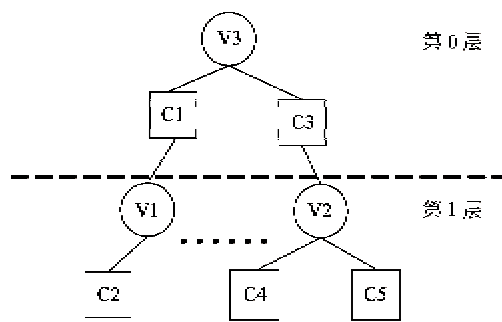


图2 树状展开图

图3展示了JRC-BPEG方法构造并行度 $r=2$ 的QC-LDPC码的基矩阵 H_1 ,该矩阵行数为 $m=14$,列数为 $n=28$ 。 H_1 中的子矩阵行列数为 $p=72$,对应码长为 $N=2016$,码率为 $1/2$ 。构造时规划的 C_{r_j} 集合如下: $C_{r_1}=\{1,3\}, C_{r_2}=\{5,7\}, C_{r_3}=\{9,11\}, C_{r_4}=\{13,2\}, C_{r_5}=\{4,6\}, C_{r_6}=\{8,10\}, C_{r_7}=\{12,14\}$ 。

10000000000000000072	0 0 32 0 0 0 0 61 0 15 0 14 5 6
1100000000000000000	0 0 0 0 1 0 1 0 20 0 0 19 6
611000000000000000	0 0 0 20 0 0 0 63 0 29 0 0 26
601100000000000000	0 1 0 0 0 0 0 1 0 0 1 0 29
600110000000000000	0 0 1 0 0 32 0 0 0 68 0 0 20 6
600011000000000000	43 0 0 0 0 0 0 0 57 0 0 1 1 6
600001100000000000	0 0 0 0 53 0 0 51 0 10 0 0 6
600000110000000000	0 0 0 1 0 1 0 0 0 2 0 0 0 37
600000011000000000	0 61 0 0 0 0 0 0 0 1 0 53 0 1
600000001100000000	36 0 0 0 0 0 53 0 0 0 31 27 0 6
600000000110000000	0 0 58 0 0 0 0 1 0 26 0 72 6
600000000011000000	0 9 0 0 42 0 0 0 38 0 0 61 4
600000000001100000	1 0 0 0 0 68 0 0 0 0 0 56 0 57
600000000000110000	0 0 0 19 0 0 49 0 0 0 2 45 0 6

图3 JRC-BPEG 码基矩阵 H_1

图3中的每个0表示 72×72 全0矩阵,各非0数值表示 72×72 单位循环偏移矩阵,且非0数值减去1表示该子矩阵的循环偏移量。观察 $\{1,3\}, \{5,7\}, \{9,11\}$ 等 C_{r_j} 集合对应的行组,每个行组内两行的非0元素没有共列,所以该校验矩阵可以支持并行度为 $r=2$ 的JRC译码。可将 H_1 的第14列中标为72的单位循环阵的最右上角的1变为0,作为编码运算的入手点^[21],也可在构造时采用类似WIMAX^[10] LDPC码的校验矩阵的双对角线结构方便地支持编码。

表3给出了一些成功构造码集的参数。这些参数都处于常用码长码率范围, p 值的选取考虑了码

率的精确和校验位数等参数的整数化。表 4 提供了 $r = 2$ 的情况下 2000 次构造中成功的次数。从表 4 数据看,这些常用码长码率是更容易构造成功的。对于基矩阵太小的情况,加大尝试次数,调整维度分布或改变 p 和 r 的配合关系,就可能构造成功。

表 3 JRC-BPEG 构造参数表

码长 N	码率 R	并行度 r	分块大小 p
16128	3/8	2	48,112,252,504,1008
16128	4/8	2	24,48,112,252,504
16128	6/8	2	24,48,112,252
16128	3/8	3	24,48,112,
16128	4/8	3	24,48,112
8064	3/8	2	24,48,112,252,504
8064	4/8	2	24,48,112,252
8064	6/8	2	24,48,112
4032	3/8	2	21,42,126,252
4032	4/8	2	24,48,112,144
4032	6/8	2	24,48
2016	3/8	2	21,42,126
2016	4/8	2	24,48,72,84
2016	6/8	2	24
1008	3/8	2	21,42
1008	4/8	2	12,24,36
1008	6/8	2	12
504	3/8	2	21
504	4/8	2	18

表 4 JRC-BPEG 算法 2000 次构造的成功次数

码长 N	码率	并行度 r	块大小 p	2000 次构造中成功的次数
16128	3/8	2	112	1860
16128	4/8	2	112	1294
16128	6/8	2	112	163
4032	3/8	2	126	51
4032	4/8	2	112	28
4032	6/8	2	48	3
1008	3/8	2	42	26
1008	4/8	2	36	15

对于基矩阵行数不能整除 r 的情形,也可以用本文算法进行构造,只是最后一个行组行数小于 r ,相应 JRC 运算并行度低一些,所得码的平均译码并行度略小于 r 。

3 JRC-BPEG 码误码性能仿真

本节通过误码率仿真,对新构造的 JRC-BPEG 码和其他典型码进行比较。本节所有误码性能仿真均采用加性高斯白噪声信道条件、JRC 译码算法,所采用的 JRC-BPEG 码并行度 $r = 2$,译码迭代次数为 30 次。

首先比较本文码与传统的 BPEG 构造法所设计的码。两种方法使用相同的设置:码长 8064,码率 0.5,分块大小 $p = 48$,列维度 $\{2,3,7,8\}$,对应的维度分布 $\{0.5119, 0.3155, 0.0714, 0.1012\}$ 。图 4 给出了 JRC-BPEG LDPC 码和 BPEG 码的误码率曲线,两曲线基本重合,所以两者误码率性能相当。此例中,JRC-BPEG 构造算法比 BPEG 算法加了更多构造约束,但是误码率性能没有变坏。

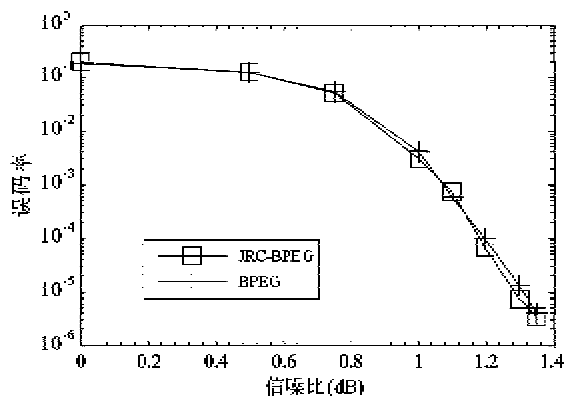


图 4 JRC-BPEG 码和 BPEG 码误码率比较

其次比较本文码与 WIMAX 标准码^[10]。JRC-BPEG 码参数为:码长 2016,码率 0.5,维度 $\{2,3,6\}$,对应维度分布 $\{0.5, 0.3333, 0.1667\}$ 。WIMAX 标准码码长为 2016,码率为 0.5。图 5 所示的是两种码的误码率曲线,从图中看出 JRC-BPEG 码性能略好。

最后比较 JRC-BPEG 码和 DVBSII 标准码^[11]的性能。JRC-BPEG 码参数为:码长 16128,码率 0.375,维度为 $\{2,3,12\}$,对应的维度分布为 $\{0.625, 0.2639, 0.1111\}$ 。DVBSII 码参数为:码长 16200,码率 0.4,等效分块大小 $p = 360$ 。图 6 所示的是两者的误码率曲线,可以看出 JRC-BPEG 码性能略优。因此 JRC-BPEG 码误码性能可以与 DVB-SII 码相比。

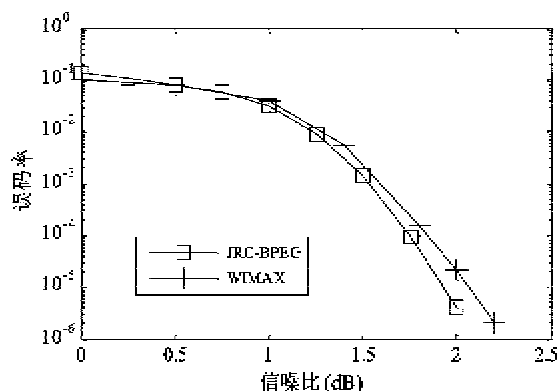


图5 JRC-BPEG 码和 WIMAX 码的误码率比较

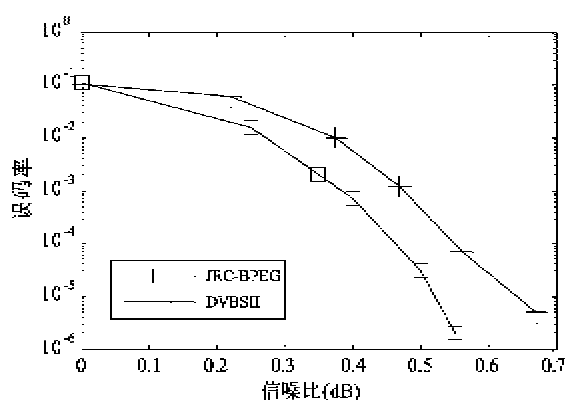


图6 JRC-BPEG 码和 DVBSII 码的误码率比较

仿真表明,采用本文方法,可构造与 BPEG 码或现行标准性能相当的 LDPC 码。

4 JRC-BPEG 码译码器方案

为阐明 JRC-BPEG 码的实现,本节给出图 7 所示的译码器实现结构。图 7 中各模块在 B5 控制模块的集中控制下按 JRC 算法协调地进行流水线式译码操作。

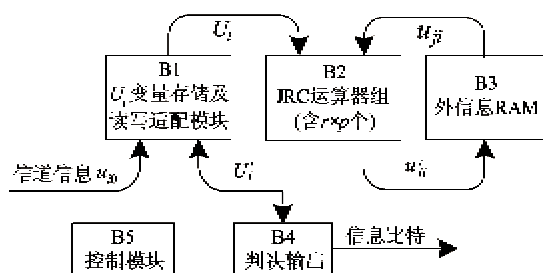


图7 一种并行 JRC 译码器结构设计

整个译码流程可以结合(1)到(3)式,从信号的

流动过程角度来描述。信道信息 u_0 进入 B1 模块,初始化 U_i 变量,B3 将模块输出值切换为 0 以初始化外信息 u_{ji} 变量。然后进行逐行组的 JRC 译码。 $r \times p$ 行对应的 $r \times p$ 个 U_i 变量经过读适配处理后送往 B2 模块。B3 中外信息存储器中的相应的 $r \times p$ 个 u_{ji} 变量也输入到 B2 模块。在 B2 模块中的 $r \times p$ 个 JRC 运算器以图 8 所示的结构进行(1)到(3)式的 JRC 运算。当一个行组所需要的所有 U_i 变量和 u_{ji} 变量被输送到 B2 单元后, U_i 变量和 u_{ji} 变量的更新值也被计算出来。B2 更新后的 U_i 变量被传回 B1 模块,进行适配调整,写入 U_i 变量存储器中。而 u_{ji} 的更新值被传到 B3 的存储器中直接进行存储。这整个运算过程中,能采用 $r \times p$ 个 JRC 运算器,译码并行度得到了保障。如此继续进行下一个行组的运算。对所有行组运算完以后,就完成一次迭代,如此反复,直到满足校验关系或译码次数达到设置值,通过 B4 判决模块输出信息比特。

图 7 中 B2 模块的每个 JRC 运算器执行(1)到(3)式的运算。图 8 是 B2 的 JRC 运算器的一种实现结构,其中减法器实现(1)式,加法器实现(3)式。而图 8 中虚框部分^[2]实现(2)式。虚框中的 MIN 运算(求最小)预处理模块求出所有第 j 行对应的所有 v_{kj} 值的符号积和最小绝对值以及次小绝对值,作为后面 MIN 运算的预备,此过程需要多个节拍处理完成,所以用 FIFO 模块进行延迟适配,然后再用预处理输出和 FIFO 延迟后的 v_{kj} 值,通过简单的组合逻辑方便地算出(2)式结果 u'_{ji} 。

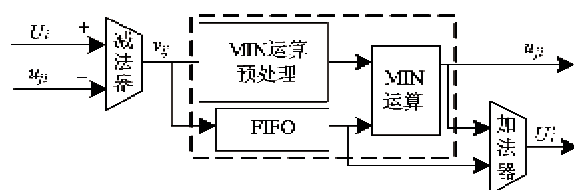


图8 JRC 运算器单元结构示意图

图 7 的实现结构规整,采用集中式控制,易于修改、调试。可以通过调整 JRC 运算器个数、总线宽度、存储器的配置来调节运算器的并行度。图 8 的 JRC 处理器因译码速率要求不同,也可对其结构做相应的并串行调整和流水线级数调整。

由于 JRC-BPEG 码也具备 QC-LDPC 校验阵特点,也可选用文献[12]的优良普通 QC-LDPC 译码结构,但采用图 7 的译码速率更高。以 $r=2$ 且码率为 $1/2$,码长为 8064 的情形为例和文献[12]同码长

码率译码器比较,图 7 结构处理外信息的并行度为文献[12]的 $r(r=2)$ 倍,且采用 JRC 译码方法迭代次数可减少约一半。另外,本文图 7 采用 JRC 运算器组只有一种运算流水线,而文献[12]的实现中有变量节点处理器组和校验节点处理器组两种流水线,而且这两个流水线以串行分时工作,所以图 7 结构一次迭代所需的运算拍数显著减少。译码器具体资源速率情况参见表 5,表中都采用 ALTERA 公司同系列现场可编程门阵列 (field programming gate array, FPGA) 器件,用 QUARTUS 软件综合。根据 FPGA 编译综合情况,图 7 中译码器结构在资源略有增加的情况下,能达到文献[12]普通 QC-LDPC 译码器结构吞吐率的 3.6 倍以上,速率有大的提高。

表 5 本文的 JRC 译码器和文献[12]译码器的对比

译码器类型	文献[12]	本文
FPGA 器件类型	EP2S60-C4	EP2S90-C4
逻辑资源 ALUT	32656	41123
存储资源 (BIT)	552133	817638
最高时钟 (MHz)	168.18	145.69
译码速率 (MBIT · S ⁻¹)	80	290

5 结 论

本文选择 JRC 译码方法作为 LDPC 码高速并行译码算法,研究了其并行实现所需的码结构,提出了基于 BPEG 算法的适于 JRC 并行译码的 LDPC 码构造方法 (JRC-BPEG)。该方法能构造比普通 QC-LDPC 码并行度更高的 LDPC 码,因而可以支持更高的译码速率。仿真结果表明,在使译码速度提高的同时, JRC-BPEG 算法构造的码和 Block-PEG 算法构造出的 LDPC 码误码性能相当。最后给出本文构造码的高并行度 JRC 译码器实现结构。

比照本文的思路,结合优良码构造方法,可为多种译码设计提供更合适的 LDPC 码集。

参考文献

- [1] Kschischang F R, Frey B J, Loeliger H A. Factor graphs and the sum-product algorithm. *IEEE Trans on Information Theory*, 2001, 47(2): 498-519
- [2] Chen J, Dholakia A, Eleftheriou E, et al. Reduced-complexity decoding of LDPC codes. *IEEE Trans On Communications*, 2005, 53(8): 1288-1299
- [3] Wu X, Song Y, Jiang M. Adaptive-normalized / offset min-sum algorithm. *IEEE Communications Letters*, 2010, 14(7): 667-669
- [4] Xu M, Wu J, Zhang M. A modified offset min-sum decoding algorithm for LDPC codes. In: Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology, Chengdu, China, 2010. 19-22
- [5] Hocevar D E. A reduced complexity decoder architecture via layered decoding of LDPC codes. In: Proceedings of the IEEE Workshop on Signal Processing Systems, Austin, USA, 2004. 107-112
- [6] He Z, Roy S, Fortier P. FPGA implementation of LDPC decoders based on joint row-column decoding algorithm. In: Proceedings of the IEEE International Symposium on Circuits and Systems, New Orleans, USA, 2007. 1653-1656
- [7] Jin J, Tsui C. An energy efficient layered decoding architecture for LDPC decoder. *IEEE Trans on VLSI Systems*, 2010, 18(8): 1185-1195
- [8] Wang Z, Cui Z, Sha J. VLSI design for low-density parity-check code decoding. *IEEE Circuits and Systems Magazine*, 2011, 11(1): 52-69
- [9] Chen Y, Chen X, Zhao Y, et al. Design and implementation of multi-mode QC-LDPC decoder. In: Proceedings of the 12th IEEE International Conference on Communication Technology, Nanjing, China, 2010. 1145-1148
- [10] IEEE-SA. IEEE 802.16E, IEEE standard for local and metropolitan area networks part 16: air interface for broadband wireless access systems. <http://standards.ieee.org/getieee802/download/802.16-2009.pdf>; IEEE, 2009
- [11] ETSI. ETSI EN 302 307, Second generation framing structure, channel coding and modulation system for broadcasting, interactive services, news gathering and other broadband satellite applications. http://pda.etsi.org/exchangefolder/en_302307v010201p.pdf; ETSI, 2009
- [12] 管武,董明科,项海格.应用循环移位矩阵设计 LDPC 码译码器. *应用科学学报*, 2009, 27(2): 117-123
- [13] Eroo M, Sun F W, Lee L N. An innovative low-density parity-check code design with near-shannon-limit performance and simple implementation. *IEEE Trans on Communications*, 2006, 54(1): 13-17
- [14] Zhang L, Huang Q, Lin S, et al. Quasi-cyclic LDPC codes: an algebraic construction, rank analysis, and codes on latin squares. *IEEE Trans on Communications*, 2010, 58(11): 3126-3139
- [15] Sha J, Wang Z, Gao M, et al. Multi-Gb/s LDPC code design and implementation. *IEEE Trans on VLSI System*,

- 2009, 17(2): 262-268
- [16] Hu X Y, Evangelos E, Arnold D M. Regular and irregular progressive edge-growth Tanner graphs. *IEEE Trans on Information Theory*, 2005, 51(1): 386-398
- [17] 乔华, 管武, 董明科等. 一种基于循环移位矩阵的 LDPC 码构造方法. 电子与信息学报, 2008, 30(10): 2384-2387
- [18] Xiong L, Yao D, Wu Y. A modified PEG algorithm for construction of LDPC codes with polynomial of cycle. In: Proceedings of the 3rd IEEE International Symposium on MAPE, Beijing, China, 2009. 524-527
- [19] Venkiah A, Declercq D, Poulliat C. Design of cages with a randomized progressive edge-growth algorithm. *IEEE Communications Letters*, 2008, 12(4): 301-303
- [20] Chung S Y, Richardson T J, Urbanke R L. Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation. *IEEE Trans on Information Theory*, 2001, 47(2): 657-670
- [21] 乔华, 管武, 董明科等. 非规则置换矩阵 LDPC 码的构造方法及装置. 中国专利, No. 200610011170. 2008-11-26

Quasi-cyclic LDPC code construction for implementation of parallel joint row-column decoding

Dong Mingke, Wang Da, Zheng Yadan, Xiang Haige

(State Key Laboratory of Advanced Optical Communication Systems & Networks,
School of Electronics Engineering and Computer Science, Peking University, Beijing 100871)

Abstract

Aiming at the restriction of normal low-density parity check (LDPC) codes on the parallelism lift of joint row-column (JRC) decoding algorithms, this paper proposes a new LDPC code construction method for parallel JRC decoding based on an improved block progressive edge-growth (BPEG) algorithm. When using this method, the rows of the quasi-cyclic LDPC (QC-LDPC) base parity check matrix are organized into groups of r rows. Parallel JRC decoding can be performed within each group. The simulation result shows that the performance of the LDPC codes constructed by the proposed method is similar to that by the BPEG method. However, the hardware implementation on the codes constructed by the proposed method shows that the throughput of the parallel decoder is more than three times of that of the typical traditional QC-LDPC decoders. This provides an example of decoder-oriented LDPC code design.

Key words: low density parity check (LDPC) code, joint row-column (JRC) decoding, progressive edge-growth (PEG), block progressive edge-growth (BPEG), quasi-cyclic LDPC (QC-LDPC), decoder oriented LDPC code design