

$GF(q)$ 上的 LDPC 分组码和 LDPC 卷积码的比较

孙卓豪

南开大学 & 电子信息科学与技术

2016 年 5 月 23 日

本论文的简要介绍

研究目的

主要工作

LDPC 卷积码与 LDPC 分组码的构造

原模图与基本图

LDPC 码的叠加方式

LDPC 分组码的构造

LDPC 卷积码的定义

LDPC 卷积码的构造

构造 LDPC 卷积码的图释

和积算法与滑动窗口解码算法

和积算法

滑动窗口解码算法

模拟仿真结果

LDPC 分组码码长与 LDPC 卷积码限制长度相等

LDPC 分组码与 LDPC 卷积码的解码延时相等

LDPC 分组码与 LDPC 卷积码的解码复杂度比较

本论文的简要介绍

研究目的

LDPC 分组码是一类接近香农极限的编码。基于原模图构造 LDPC 分组码具有规则的空间结构。类似的，基于原模图构造 LDPC 卷积码具有规则的空间结构，且与 LDPC 分组码的结构类似。这两种 LDPC 码解码时各有特点。本文的目的就是在构造方法类似的前提下，研究两者解码时各方面的差异。

本论文的简要介绍

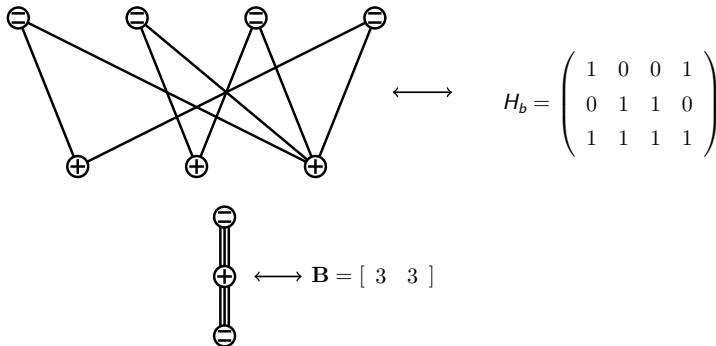
主要工作

本文的主要工作有：

- ▶ 描述 LDPC 卷积码与 LDPC 分组码的构建过程；
- ▶ 描述 LDPC 卷积码与 LDPC 分组码的解码算法，计算两种算法的解码延时；
- ▶ 调整几类参数，比较 LDPC 卷积码与 LDPC 分组码的性能。

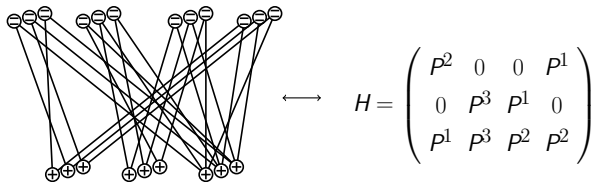
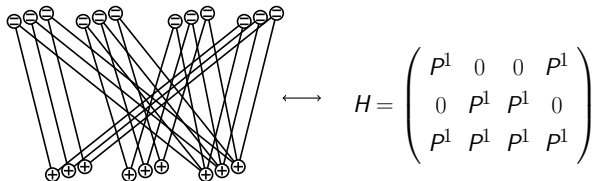
LDPC 卷积码与 LDPC 分组码的构造

原模图与基本图



LDPC 卷积码与 LDPC 分组码的构造

LDPC 码叠加方式



LDPC 卷积码与 LDPC 分组码的构造

LDPC 分组码的构造

$GF(q)$ 为含 $q = 2^m$ 个元素的有限域，其中 m 为在 $GF(q)$ 代表一个符号所需的位数。令 M 为原模图叠加数，其中 M 是一个大整数。通过以下两步从原模图的 $(c - b) \times c$ 邻接矩阵 $\mathbf{B} = [B_{i,j}]$ 构造码长为 $n_{BC} = Mc$ 的 q 元 LDPC 分组码：

1. 将 \mathbf{B} 中的非零元 $B_{i,j}$ 替换为随机选择的 $M \times M$ 置换矩阵，将 \mathbf{B} 中的零元 $B_{i,j}$ 替换为 $M \times M$ 零矩阵。此时得到对应于 \mathbf{B} 的二元校验矩阵 \mathbf{H} ；
2. 将 \mathbf{H} 中非零元替换为从有限域 $GF(q)$ 中随机选取的元素，得到 LDPC 分组码的 q 元校验矩阵 \mathbf{H}_{BC} 。

对于 LDPC 分组码，必须等待整个码块接受完毕才能执行置信传播解码算法。故 q 元 LDPC 分组码的解码延时为

$$T_{BC} = n_{BC} \cdot m = Mmc$$

LDPC 卷积码与 LDPC 分组码的构造

LDPC 卷积码的定义

对于给定码率为 $R = b/c$ 的 LDPC 卷积码, 其定义为, 存在校验矩阵

$\mathbf{H}_{[\infty]}$ 使得无限长向量 $\mathbf{v}_{[\infty]}$ 有 $\mathbf{H}_{[\infty]} \mathbf{v}_{[\infty]}^T = \mathbf{0}_{[\infty]}$, 而 $\mathbf{0}_{[\infty]}$ 是无限长零向量。 $\mathbf{H}_i(t), i = 0, 1, \dots, m_s$ 为 $(c-b) \times c$ 的矩阵满足以下条件:

- ▶ $\mathbf{H}_i(t) = \mathbf{0}$, 当 $i < 0$ 和 $i > m_s, \forall t \geq 1$;
- ▶ $\exists t \geq 0$, 使得 $\mathbf{H}_{m_s}(t) \neq \mathbf{0}$;
- ▶ $\forall t \geq 1, \mathbf{H}_0(1)$ 满秩。

$$\mathbf{H}_{[\infty]} = \begin{bmatrix} \mathbf{H}_0(1) & & & & \\ \mathbf{H}_1(1) & \mathbf{H}_0(2) & & & \\ \vdots & \mathbf{H}_1(2) & \ddots & & \\ \mathbf{H}_{m_s}(1) & \vdots & \ddots & \mathbf{H}_0(t) & \\ & \mathbf{H}_{m_s}(2) & \ddots & \mathbf{H}_1(t) & \ddots \\ & & \ddots & \vdots & \ddots \\ & & & \mathbf{H}_{m_s}(t) & \ddots \\ & & & & \ddots \end{bmatrix}$$

LDPC 卷积码与 LDPC 分组码的构造

LDPC 卷积码的构造

设原模图的基本矩阵为 $(c - b) \times c$ 的 \mathbf{B} 。首先构造 \mathbf{B}_{CC} 如右。其中 m_s 为当前原模图与前一个原模图相连的边数。 $\mathbf{B}_0, \mathbf{B}_1, \dots, \mathbf{B}_{m_s}$ 为 $(c - b) \times c$ 矩阵, 且满足

$$\sum_{i=0}^{m_s} \mathbf{B}_i = \mathbf{B}$$

然后将 \mathbf{B}_{CC} 中的非零元替换为随机选择的 $M \times M$ 置换矩阵, 将 \mathbf{B}_{CC} 中的零元替换为 $M \times M$ 零矩阵, 得到 LDPC 卷积码校验矩阵 \mathbf{H}_{CC} 。

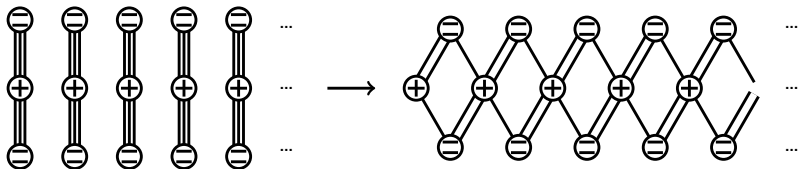
$$\mathbf{B}_{CC} = \begin{bmatrix} \mathbf{B}_0 & & & & \\ \mathbf{B}_1 & \mathbf{B}_0 & & & \\ \vdots & \mathbf{B}_1 & \ddots & & \\ \mathbf{B}_{m_s} & \vdots & \ddots & \ddots & \\ & \mathbf{B}_{m_s} & \ddots & \ddots & \ddots \end{bmatrix}$$

最后将 \mathbf{H}_{CC} 中的非零元替换为从有限域 $\text{GF}(q)$ 中随机选取的元素, 得到 LDPC 卷积码的 q 元校验矩阵 \mathbf{H}_{CC} , 其限制长度为 $v_s = (m_s + 1)Mc$ 。

LDPC 卷积码与 LDPC 分组码的构造

构造 LDPC 卷积码的图释

下图为通过原模图叠加方法构造 LDPC 卷积码的示意图。首先复制几次 (3,6) 正则 LDPC 码的原模图，设定记忆因子 $m_s = 1$ ，然后将原模图之间连接起来。其中，原模图的基本矩阵为 $\mathbf{B} = \begin{bmatrix} 3 & 3 \end{bmatrix}$ ，使用到的组成矩阵为 $\mathbf{B}_0 = \begin{bmatrix} 2 & 1 \end{bmatrix}$, $\mathbf{B}_1 = \begin{bmatrix} 1 & 2 \end{bmatrix}$ 。



和积算法与滑动窗口解码算法

和积算法

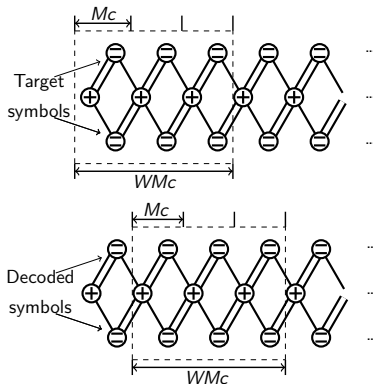
其具体步骤为

1. 给信息节点发送到校验节点的信息 $M_{j,i}$ 赋值为 R_i ;
2. 计算校验节点发送到信息节点的信息 $E_{j,i}$;
3. 计算信息节点的 LLR, L_i 。生成预测码字 \hat{c} , 代入校验方程, 若满足, 则停止算法, 或者达到最大迭代次数, 停止算法;
4. 计算信息节点发送到校验节点的信息 $M_{j,i}$, 遍历次数加一。继续第二步。

如果该算法收敛, 经过足够多次迭代后, 将渐近求出码字中各位为 1 或者 0 的概率, 进而实现逐符号最大后验概率译码。

和积算法与滑动窗口解码算法

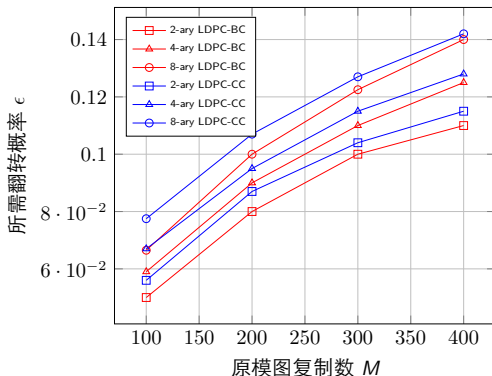
滑动窗口解码算法



模拟仿真结果

LDPC 分组码码长与 LDPC 卷积码限制长度相等

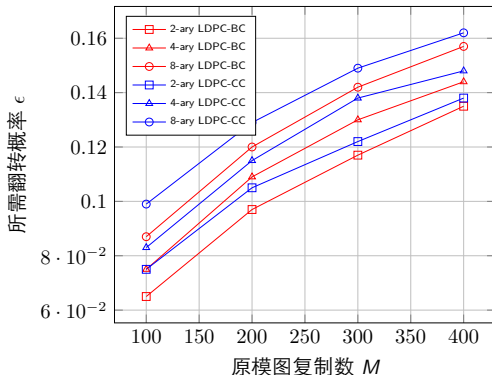
右图展示的是 q 元 LDPC 分组码及 q 元 LDPC 卷积码解码时误比特率 BER 达到 10^{-4} 所需翻转概率 ϵ 与原模图复制数 M 之间的关系。 q 元 LDPC 分组码及 q 元 LDPC 卷积码都由码率为 $R = 1/2$ 的 (2,4) 正则原模图构造。 q 元 LDPC 卷积码的滑动窗口解码器的窗口大小为 $W = 12$ 。



模拟仿真结果

LDPC 分组码码长与 LDPC 卷积码限制长度相等

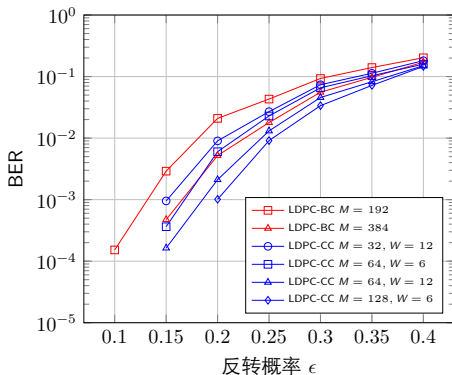
右图描述的是 q 元 LDPC 分组码及 q 元 LDPC 卷积码解码时误比特率 BER 达到 10^{-4} 所需翻转概率 ϵ 与原模图复制数 M 之间的关系。 q 元 LDPC 分组码及 q 元 LDPC 卷积码都由 $R = 1/2$ 的 (3,6) 正则原模图构造。



模拟仿真结果

LDPC 分组码与 LDPC 卷积码的解码延时相等

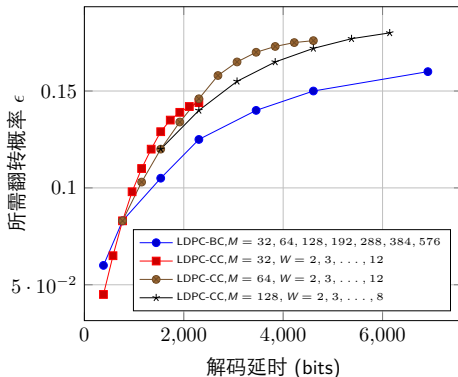
右图中 8 元 LDPC 分组码
及 8 元 LDPC 卷积码都由
码率为 $R = 1/2$ 的 (3,6)
正则原模图构造。



模拟仿真结果

LDPC 分组码与 LDPC 卷积码的解码延时相等

上图展示在 $GF(8)$ 上比较
要达到相同的误比特率所
需的翻转概率与解码延时
之间的关系。其中 8 元
LDPC 分组码及 8 元
LDPC 卷积码都由码率为
 $R = 1/2$ 的 (3,6) 正则原模
图构造。



模拟仿真结果

LDPC 分组码与 LDPC 卷积码的解码延时相等

解码 延时	LDPC-BC			LDPC-CC		
	GF(2)	GF(4)	GF(8)	GF(2)	GF(4)	GF(8)
2304 bits	0.14	0.15	0.15	0.12	0.16	0.18
4608 bits	0.16	0.17	0.16	0.18	0.19	2.0
6912 bits	0.17	0.18	0.17	0.19	0.21	0.21
9216 bits	0.17	0.19	0.19	0.21	0.22	0.22

上表展示了, (3,6) 正则 q 元 LDPC 分组码与 (3,6) 正则 q 元 LDPC 卷积码在不同的 $GF(q)$ 以及不同的解码延时的条件下, 误比特率 BER 达到 10^{-4} 所需翻转概率 ϵ 。

模拟仿真结果

LDPC 分组码与 LDPC 卷积码的解码复杂度比较

解码 延时	I_{BC}			$I_{CC}(W=6)$		
	GF(2)	GF(4)	GF(8)	GF(2)	GF(4)	GF(8)
4608 bits	13.8	12.3	11.1	3.3	3.2	3.0
6912 bits	15.6	14.1	12.6	3.9	3.7	3.4

基于相等的解码延时的条件，比较 q 元 LDPC 卷积码与 q 元 LDPC 分组码的计算复杂度。上表展示了，在解码延时分别为 4608, 6912 比特时，(3,6) 正则 q 元 LDPC 卷积码与 (3,6) 正则 q 元 LDPC 分组码解码误比特率达到 10^{-4} 所需的平均迭代次数 I_{CC} 和 I_{BC} 。

模拟仿真结果

LDPC 分组码与 LDPC 卷积码的解码复杂度比较

右图展示了, (3,6) 正则 q 元 LDPC 卷积码与 (3,6) 正则 q 元 LDPC 分组码每解码一比特的计算复杂度与 $GF(q)$ 的关系。

