

◆ 算法研究与应用 ◆

基于量化修正的低复杂度 LDPC 译码算法^{*}杜立婵¹,黄绎琿²,王文静¹,聂晶¹,黎相成^{2,3**}

(1. 南宁职业技术学院人工智能学院,广西南宁 530008;2. 广西大学计算机与电子信息学院,广西南宁 530004;3. 广西多媒体通信与网络技术重点实验室,广西南宁 530004)

摘要:为进一步降低基于可靠度的(Low Density Parity Check, LDPC)译码算法复杂度,提出一种基于量化修正的低复杂度 LDPC 译码算法,该算法在对信道信息预处理时引入量化信息修正处理策略,从而避免在译码迭代过程中进行译码信息修正处理操作,在保持译码性能的同时,较大幅度地降低译码复杂度。针对均匀和非均匀量化方案,本文实现了基于修正系数的均匀量化和基于列重修正的非均匀量化两种译码方案。仿真实验结果表明,所提出的两种译码方案在算法复杂度明显较低的情况下,仍然与 MRBI-MLGD 译码算法具有相同的译码性能与收敛速度。特别地,在基于列重修正的非均匀量化方案中,只需 3-4 bits 的低比特量化即可达到均匀量化中高比特量化(8 bits)相同的迭代次数和误比特率性能。

关键词:低密度奇偶校验码 低复杂度 可靠度 量化修正 预处理

中图分类号:TN911.22 文献标识码:A 文章编号:1005-9164(2022)02-0293-08

DOI:10.13656/j.cnki.gxkx.20220526.009

现代通信系统自出现以来,实现更安全、更高效及更可靠的通信方式是人类一直以来追求的目标。随着信息技术的高速发展,特别是在深度学习和人工智能等领域取得的巨大进步,大大加速了云计算、无人驾驶以及车联网等新型技术的应用。同时,对通信系统也提出了更高的可靠性、更小的时延以及更大的数据带宽要求。信道编码是通信系统中的基础核心技术,(Low Density Parity Check, LDPC)译码作为

优秀的信道编码方案,具有纠错性能好且易于并行译码的特点,吸引了来自国内外众多研究者的关注。LDPC 码由 Gallager^[1,2]首次提出,但受限于当时的硬件技术水平,其译码算法难以实现,在此后的 30 多年间基本被研究者忽略。到 20 世纪 90 年代中后期,随着计算机及硬件技术的发展,特别是 Mackay 等^[3-6]开展的系列研究,重新掀起了人们对 LDPC 码的关注,围绕 LDPC 码的研究取得了大量实用的研

收稿日期:2021-12-01

^{*} 国家自然科学基金项目(61761006,61961004),广西自然科学基金项目(2017GXNSFAA198276,2018GXNSFAA138079)和广西高校中青年教师基础能力提升项目(2022KY1015)资助。

【作者简介】

杜立婵(1979-),女,副教授,主要从事物联网应用技术及信道编译码理论与技术研究。

【**通信作者】

黎相成(1979-),男,博士,工程师,主要从事信道编译码理论与技术研究,E-mail:xcli@gxu.edu.cn。

【引用本文】

杜立婵,黄绎琿,王文静,等.基于量化修正的低复杂度 LDPC 译码算法[J].广西科学,2022,29(2):293-300.

DU L C, HUANG Y H, WANG W J, et al. Low Complexity LDPC Decoding Algorithm Based on Quantification Correction [J]. Guangxi Sciences, 2022, 29(2): 293-300.

究成果^[7-11]。由于众多研究者的努力,在2016年底,3GPP RAN1 # 86次会议决定将准循环LDPC码确定为5G-eMBB通信环境下大数据包(长码)的信道编码方案。

一般情况下,可以将LDPC码的译码算法分为以下3个种类^[12]:①软判决译码算法,②硬判决译码算法,③基于可靠度的译码算法。软判决译码算法主要为和积译码算法(SPA)与其改进算法,在上述算法中性能最优,但由于在算法中采用浮点数乘法以及对数运算,其计算复杂度非常高。早期的硬判决算法主要有Gallager教授^[1]的比特翻转(BF)译码算法,以及Lin等^[13]提出的一步大数逻辑译码(OSMLGD)算法。其中BF译码算法的核心思想为采用该比特参与运算的校验和进行判断是否翻转该位,而OSMLGD是首个类MLGD译码算法,两者的计算复杂度都极低,由于采用信道的硬判决信息进行译码,译码信息损失较大,所以译码性能较差。最近,Liu等^[14]通过引入比特位自适应阈值门限策略,避免了全局最大值搜索处理,实现了高吞吐量、低时延的BF译码算法,特别适合硬件实现。硬判决与软判决译码算法均存在明显的优缺点,研究者们结合两类算法的特点展开研究,提出了一类基于可靠度的译码算法,实现了性能与复杂度之间的均衡。例如,Huang等^[15]提出基于可靠度的迭代大数逻辑译码(RBI-MLGD)算法,该算法对初始信道信息进行量化处理,作为信道信息的可靠度量,在译码迭代过程中,通过在校验节点获取外信息对变量节点的可靠度信息进行更新处理,结合大数逻辑的处理策略实现译码性能的提升。此外,Chen等^[16]在RBI-MLGD算法的基础上提出了改进的译码(MRBI-MLGD)算法,该算法的核心思想是在迭代译码信息更新过程中引入修正系数,同时,只在初始信道信息的基础上进行信息更新处理,提高了译码信息的迭代更新效率,以此实现了译码性能的提升。由于MLGD类算法具有低复杂度、低时延的特点,研究者一直对其开展相关研究。最近,Song等^[17]针对多元LDPC译码算法,基于软可靠度迭代IISRB-MLGD译码算法,提出了一种裁剪的CM-IISRB译码算法。该算法采用非饱和截切策略,在较低复杂度的情况下,实现了更优的译码性能。

在MRBI-MLGD译码算法中,由于引入了缩放因子,即采用了乘法修正系数,不可避免地增加了大量的浮点数乘法运算,其译码复杂度随着迭代次数的

递增而明显增大。受MRBI-MLGD算法在译码迭代过程中引入缩放因子的启发,本文提出了一种基于量化修正的低复杂度LDPC译码算法,该算法在对信道信息预处理时引入量化信息修正处理策略,从而避免在译码迭代过程中进行译码信息修正处理操作,在保持译码性能的同时,较大幅度地降低了译码复杂度。基于上述算法思想,针对均匀和非均匀量化方案,本文具体实现了基于修正系数的均匀量化和基于列重信息修正的非均匀量化两种译码方案设计,期望所提出的译码方案在降低译码复杂度的同时,还能保持一定的译码性能。

1 数字通信模型及相关定义

定义 $\mathbf{H}=[h_{i,j}]_{m \times n}$ ($h_{i,j} \in \mathbb{F}_2$) 为一个稀疏矩阵,如矩阵 \mathbf{H} 具有恒定的行重 ρ ,即每行非零元素个数,恒定的列重 γ ,即每列非零元素个数,则定义为规则LDPC码。为方便接下来对本文算法进行描述,定义以下两个集合:①定义矩阵 \mathbf{H} 中第 i 行非零列下标集合 $N_i=\{j:0 \leq j \leq n-1, h_{i,j} \neq 0\}$;②定义矩阵 \mathbf{H} 中第 j 列非零行下标集合 $M_j=\{i:0 \leq i \leq m-1, h_{i,j} \neq 0\}$ 。设传输的信息序列为 $\mathbf{u}=(u_0, u_1, \dots, u_{k-1})$, $\mathbf{u} \in \mathbb{F}_2^k$ 。经过LDPC编码后可以得到码字序列 $\mathbf{c}=(c_0, c_1, \dots, c_{n-1})$, $\mathbf{c} \in \mathbb{F}_2^n$,其中,满足 $\mathbf{c}\mathbf{H}^T=0$ 。接下来对获得的LDPC码字序列 \mathbf{c} 进行调制,获得发送的实数信号序列 $\mathbf{x}=(x_0, x_1, \dots, x_{n-1})$,可采用BPSK进行调制,本文具体实现方法为 $x_j=1-2c_j$ 。然后,把信号序列 \mathbf{x} 经过加性高斯白噪声信道(AWGN)进行传输,在接收端即可获得一个接收信号序列 $\mathbf{y}=(y_0, y_1, \dots, y_{n-1})$,该序列由发送信号叠加一个高斯白噪声信号,可以表示为 $y_j=x_j+n_j$,其中, n_j 为高斯白噪声,且满足 $n_j \sim N(0, \sigma^2)$ 。

2 MRBI-MLGD 算法描述

在接收端获得信号序列 $\mathbf{y}=(y_0, y_1, \dots, y_{n-1})$,将其进行量化处理后获得量化序列 $\mathbf{q}=(q_0, q_1, \dots, q_{n-1})$,即为来自信道的初始可靠度信息序列,令 $R_j^{(0)}=q_j$,其中,设量化函数为 $Q(y_j)=q_j$ 。通过量化序列 \mathbf{q} ,可计算获得其硬判决码字序列 $\mathbf{z}=(z_0, z_1, \dots, z_{n-1})$,具体计算方法如下:

$$z_j = \begin{cases} 0, & R_j^{(k)} \geq 0 \\ 1, & R_j^{(k)} < 0 \end{cases}, \quad (1)$$

在获得硬判决序列 \mathbf{z} 后, 可以计算其相应的校验伴随式 $\mathbf{s} = \mathbf{z}\mathbf{H}^T$, 其中, $\mathbf{s} = (s_0, s_1, \dots, s_{n-1})$, 实现计算方法如下:

$$s_i^{(k)} = \sum_{0 \leq j \leq n-1} \oplus z_j^{(k)} h_{i,j}, \quad (2)$$

式中, $h_{i,j}$ 为校验矩阵 \mathbf{H} 中的元素, 符号 \oplus 为模 2 加运算, 上标 k 表示第 k 次迭代。如 $\mathbf{s} = 0$ 则表示当前判决码字校验成功, 可输出码字序列 $\mathbf{z} = (z_0, z_1, \dots, z_{n-1})$; 如 $\mathbf{s} \neq 0$ 则表明校验失败, 继续进入迭代译码过程。接下来对具体的迭代译码过程进行描述。

① 计算外信息。

在迭代过程中, 译码算法在校验节点 C_i 中计算外信息 $\sigma_{i,j}^{(k)}$, 同时, 将该外信息传递至其对应的变量节点 V_j 。校验节点 C_i 传递至变量节点 V_j 的外信息 $\sigma_{i,j}^{(k)}$ 计算方法如下:

$$\sigma_{i,j}^{(k)} = \sum_{j' \in N_{i,j}} \oplus z_{j'}^{(k)} h_{i,j'}, \quad (3)$$

式中, 符号 $N_{i,j}$ 表示除了当前 j 列的其他列的序号集合。

② 计算总外信息。

经过上述计算后, 变量节点 V_j 即可获得所有校验节点的外信息 $\sigma_{i,j}^{(k)}$, 在变量节点 V_j , 可计算该变量节点的总外信息 $\epsilon_j^{(k)}$, 其计算方法如下:

$$\epsilon_j^{(k)} = \sum_{i \in M_j} (1 - 2\sigma_{i,j}^{(k)}). \quad (4)$$

③ 更新可靠度。

设在译码迭代过程中, 设 $R_j^{(k)}$ 为第 k 次迭代时变量节点 V_j 的当前可靠度信息, 其信息更新策略如下:

$$R_j^{(k+1)} = R_j^{(0)} + \alpha \cdot \epsilon_j^{(k)}, \quad (5)$$

式中, α 为译码信息修正系数。从式(5)可以看出, 变量节点的信息处理方法是在初始的信道信息 $R_j^{(0)}$ 基础上进行更新, 这种处理机制对迭代系统的稳定性有帮助。此外, 由于引入了修正系数, MRBI-MLGD 译码算法在收敛速度和译码性能方面, 相较于原 RBI-MLGD 译码算法均获得明显提升。

3 基于量化修正的译码算法设计

在上述 MRBI-MLGD 算法中, 变量节点信息更新函数为 $R_j^{(k+1)} = R_j^{(0)} + \alpha \cdot \epsilon_j^{(k)}$, 通过数值分析发现, $\epsilon_j^{(k)}$ 信息为所有来自相连接校验节点的外信息之和, 校验节点的外信息为二值的, $\epsilon_j^{(k)}$ 值的大小与 LDPC 码的列重相关。而算法为保持一定的稳定性, 采用了在 $R_j^{(0)}$ 的基础上进行信息修正。因此, 对外信息和 $\epsilon_j^{(k)}$ 采用 α 系数进行修正, 以此达到变量节点合适的

信息修正。 α 值一般通过在一定的信噪比范围内仿真搜索出性能表现最好的值。而在原始的 RBI-MLGD 译码算法中, 在对变量节点译码信息更新时, 每一个来自校验节点的外信息修正值为 ± 1 , 在不进行任何信息修正时, 其总外信息绝对值最大为列重 γ 。一般为了获取高译码性能, 在算法中将量化比特 b 设为 8 bits, 则信道信息的量化值一般远大于更新修正值, 在此条件下, 要想实现信息的正确修正, 需要更多的迭代次数。基于此, 改进算法 MRBI-MLGD 中通过引入缩放因子 α , 实现了收敛速度的增长和译码性能的提升。该改进算法由于在迭代过程中引入了乘法运算, 使得计算复杂度大大提升。

基于以上分析, 为降低译码复杂度, 本文提出了通过在信道信息初始预处理时引入量化修正处理, 在迭代过程中不再使用信息修正操作, 实现译码复杂度的降低。针对均匀和非均匀量化方案, 本文提出了两种量化修正方案, 一种为基于修正系数的均匀量化译码方法, 具体地, 该方法通过在量化函数中引进修正系数, 避免在译码迭代中引进浮点乘法运算, 在保持译码性能的前提下, 实现了译码迭代过程中计算复杂度的明显下降; 另一种为基于列重 γ 与量化比特 b 的改进型非均匀量化方案, 在对 3-4 bits 极低量化比特的情况下, 也能保持较优秀的译码性能, 同时, 其算法复杂度和硬件资源消耗也大为降低, 该方案可用于硬件资源遭受限制的场景。

3.1 基于修正系数的均匀量化及译码方案设计

由译码判决信息规则和量化原理可知, 在不改变信道信息可靠度极性的情况下, 不会影响 LDPC 码字的判决, 因此, 可以对可靠度值进行缩放处理。为了避免在迭代译码过程中引入乘法修正运算, 降低迭代译码过程的计算复杂度, 本文提出的算法 1 为在对信道信息量化预处理时引入修正系数 β , 对量化值进行预修正处理。具体计算方法如下:

$$q_j = \begin{cases} +\beta(2^b - 1), & y_j \geq (2^b - 1)\Delta \\ \beta \left\lfloor \frac{y_j}{\Delta} \right\rfloor, & |y_j| < (2^b - 1)\Delta \\ -\beta(2^b - 1), & y_j \leq -(2^b - 1)\Delta \end{cases}, \quad (6)$$

式中, b 为量化比特位数, Δ 为量化间隔。采用修正系数 β 对量化值进行修正, 通过修正处理, 使信道初始可靠度值与译码迭代过程中的译码信息更新在数值上相适配, 从而实现更有效的译码信息更新, 获得译码性能的提升。其中, β 值可以通过仿真搜索获得。

此外,在 MRBI-MLGD 算法中变量节点信息更新策略式(5)修改如下:

$$R_j^{(k+1)} = R_j^{(0)} + \epsilon_j^{(k)}, \quad (7)$$

从式(7)可以看出,相比式(5)取消了修正系数 α , 由于在对信道信息量化时便引入了信息修正策略,此时的初始可靠度 $R_j^{(0)}$ 进行了相应修正处理,与外信息和 $\epsilon_j^{(k)}$ 之间已做了数值上的适配,因此不需要在更新规则中引入修正系数 α 。相较于 MRBI-MLGD, 本文算法可靠度 $R_j^{(k+1)}$ 的更新规则省掉浮点乘法运算,译码迭代过程中的浮点乘法计算量实现大幅削减。

对本文提出的算法 1——基于修正系数的均匀量化方案及 LDPC 译码算法具体实现步骤描述如下。

步骤 1: 输入。

最大迭代次数 I_{\max} , 接收信号向量 \mathbf{y} , 修正系数 β , 量化参数 b 和 Δ 。

步骤 2: 初始化。

按式(6)对接收信号 \mathbf{y} 进行量化得到 q_j , 将迭代次数 k 置 0, 对 $0 \leq j \leq n-1$, 令初始可靠度 $R_j^{(0)} = q_j$ 。

步骤 3: 迭代过程。 对 $k < I_{\max}$,

- ① 按式(1)计算硬判决序列 $\mathbf{z}^{(k)}$;
- ② 按式(2)计算校验和 $\mathbf{s}^{(k)}$, 如果 $\mathbf{s}^{(k)} = 0$, 则执行步骤 4;
- ③ 遍历 $0 \leq i \leq m-1$, 按式(3)计算外信息 $\sigma_{i,j}^{(k)}$;
- ④ 遍历 $0 \leq j \leq n-1$, 按式(4)计算总外信息 $\epsilon_j^{(k)}$;
- ⑤ 遍历 $0 \leq j \leq n-1$, 按式(7)更新可靠度 $R_j^{(k+1)}$;
- ⑥ 迭代次数 $k = k + 1$ 。

步骤 4: 输出。

输出硬判决序列 $\mathbf{z}^{(k)}$ 。

3.2 基于列重修正的非均匀量化及译码方案设计

在均匀量化方案中,由于采用固定步长进行量化,因此在对接近 0 区域的信号进行量化时,其量化精度较粗糙,不能更精细地对信道信息进行有效表达。特别是在采用较低量化比特时,会严重影响译码性能。上述总外信息计算式(4)可以看出,每次译码迭代时,可靠度的修正幅度 $\epsilon_j^{(k)}$ 与列重 γ 相关。当取量化比特 b 较大,而此时列重 γ 较低时,则每次迭代信息的更新幅度有限,大大降低了可靠度 $R_j^{(k+1)}$ 的更新效率。例如,量化比特 $b = 8$, 此时量化数值的范围比较大,如果仅仅简单采用列重 γ 个单位外信息对变量节点进行修正,由于列重 γ 的数值与量化范围数值

相差较大,通过有限的 k 次迭代可能无法正确改变可靠度 $R_j^{(0)}$ 极性,也就是无法对错误比特进行有效翻转,实现正确译码。在改进的 MRBI-MLGD 译码算法中,通过在可靠度更新时引入修正系数 α 实现与量化范围的适配,获得收敛速度和译码性能的提升。由于在译码迭代过程中引入了乘法操作,大大增加了计算复杂度。基于上述描述,本文提出的算法 2 为基于列重 γ 修正的非均匀量化预处理译码方案,目的在对信道信息预处理阶段,引入列重 γ 信息,实现量化值与译码方案的适配,获得收敛速度和译码性能的提升。由于常用 LDPC 码的列重 γ 的数值一般比较小,因此本设计采用低量化比特的非均匀量化方法。具体的计算方法如下:

$$q_j = \begin{cases} \theta \operatorname{sgn}(y_j) (2^b - 1), & |y_j| \geq r^0 \\ \theta \operatorname{sgn}(y_j) (2^b - 1 - p), & r^{p+1} \leq |y_j| < r^p, \\ \theta \operatorname{sgn}(y_j), & 0 \leq |y_j| < r^{2^b-2} \end{cases} \quad (8)$$

式中, $\theta = \frac{\gamma}{(2^{b+1} - 2)}$, p 为整数,且满足 $0 \leq p \leq 2^b - 3$, 参数 r 决定非均匀量化特性,取值范围为 $0 < r < 1$ 。其中, r 的取值决定信息的量化精度,信道信息在 0 附近量化精度高,随着距离的加大量化精度越低。可通过合理地分配量化距离,在最小比特量化时实现译码性能的提升。该 r 值可通过仿真搜索获取。

本量化设计方案也是采用在量化预处理阶段对信道信息进行相应处理,因此,其译码迭代时可靠度的更新规则采用式(7)进行。

对本文提出的算法 2——基于列重修正的非均匀量化及 LDPC 译码算法具体实现步骤描述如下。

步骤 1: 输入。

最大迭代次数 I_{\max} , 接收信号向量 \mathbf{y} , 修正系数 β , 量化参数 b 和 Δ 。

步骤 2: 初始化。

按式(8)对接收信号 \mathbf{y} 进行量化得到 q_j , 将迭代次数 k 置 0, 对 $0 \leq j \leq n-1$, 令初始可靠度 $R_j^{(0)} = q_j$ 。

步骤 3: 迭代过程。 对 $k < I_{\max}$,

- ① 按式(1)计算硬判决序列 $\mathbf{z}^{(k)}$;
- ② 按式(2)计算校验和 $\mathbf{s}^{(k)}$, 如果 $\mathbf{s}^{(k)} = 0$, 则执行步骤 4;
- ③ 遍历 $0 \leq i \leq m-1$, 按式(3)计算外信息 $\sigma_{i,j}^{(k)}$;
- ④ 遍历 $0 \leq j \leq n-1$, 按式(4)计算总外信息 $\epsilon_j^{(k)}$;

⑤ 遍历 $0 \leq j \leq n-1$, 按式(7)更新可靠度 $R_j^{(k+1)}$;

⑥ 迭代次数 $k = k + 1$ 。

步骤 4: 输出。

输出硬判决序列 $\mathbf{z}^{(k)}$ 。

4 译码复杂度和性能仿真分析

4.1 译码复杂度

本文提出的基于量化修正的译码算法复杂度主要由以下几个部分组成:

① 式(1)计算硬判决 $\mathbf{z}^{(k)}$ 时, 需要进行 n 次逻辑运算;

② 式(2)计算校验和 $\mathbf{s}^{(k)}$ 时, 需要 $m(\rho - 1) = n\gamma - m$ 次逻辑运算(其中 ρ 为行重, $m\rho = n\gamma$ 为校验矩阵 \mathbf{H} 的非零因子的个数);

③ 在计算总外信息 $\epsilon_j^{(k)}$ 时, 采用式(3)和(4), 共需要 $n(\gamma - 1)$ 次加法运算和 $n\gamma$ 次逻辑运算;

④ 最后, 可靠度 $R_j^{(k+1)}$ 更新时, 采用式(7)需要 n 次整数加法。

同时, 本文对 RBI-MLGD、MRBI-MLGD 及经典的 SPA 算法进行了复杂度分析。详细的复杂度对比情况如表 1 所示。

表 1 单次译码迭代计算复杂度比较

Table 1 Comparison table of computational complexity of single decoding iteration

算法名称 Name of algorithm	单次迭代运算量 Number of operations per iteration			
	BO	AO	RM	Log
Proposed	$2n\gamma + n - m$	$n\gamma$		
RBI-MLGD	$2n\gamma + n - m$	$n\gamma$		
MRBI-MLGD	$2n\gamma + n - m$	$n\gamma$	n	
SPA			$6n\gamma$	n

其中, BO 为二进制操作, AO 为加法运算, RM 为实数乘法, Log 为对数运算。从表 1 可以看出, 本文提出的改进算法与原始的 RBI-MLGD 算法复杂度基本一致, 本算法相比 MRBI-MLGD 算法, 省掉了译码迭代过程中可靠度更新时的实数乘法计算, 将会明显降低译码迭代过程中的复杂度。为直观地看出单次计算复杂度的数值情况, 下面对具体的 LDPC 码单次计算复杂度进行数值计算。

表 2 为 $\mathbb{F}_2(1\ 023, 781)$ 规则循环 LDPC 码在不同译码算法中单次迭代计算的具体复杂度, 其列重和行重均为 32, 码率为 0.76。

表 3 为 $\mathbb{F}_2(255, 175)$ 规则循环 LDPC 码在不同译码算法中单次迭代计算的具体复杂度, 其列重和行重均为 16, 码率为 0.69。

表 2 $\mathbb{F}_2(1\ 023, 781)$ 规则循环 LDPC 码单次迭代计算复杂度
Table 2 $\mathbb{F}_2(1\ 023, 781)$ single iteration computational complexity of regular cyclic LDPC codes

算法名称 Name of algorithm	单次迭代运算量 Number of operations per iteration			
	BO	AO	RM	Log
Proposed	65 472	32 736		
RBI-MLGD	65 472	32 736		
MRBI-MLGD	65 472	32 736	1 023	
SPA			196 416	1 023

表 3 $\mathbb{F}_2(255, 175)$ 规则循环 LDPC 码单次迭代计算复杂度

Table 3 $\mathbb{F}_2(255, 175)$ single iteration computational complexity of regular cyclic LDPC codes

算法名称 Name of algorithm	单次迭代运算量 Number of operations per iteration			
	BO	AO	RM	Log
Proposed	8 160	4 080		
RBI-MLGD	8 160	4 080		
MRBI-MLGD	8 160	4 080	255	
SPA			24 480	255

4.2 译码性能

根据上文的算法描述, 继续通过计算机仿真的方法, 对提出的译码算法进行仿真实验, 考察其译码性能。

实验 1: 实验采用基于欧式几何方法构造的 $\mathbb{F}_2(1\ 023, 781)$ 规则循环 LDPC 码^[18]进行仿真实验, 其仿真参数采用以下设置: ① 对均匀量化, 量化比特 $b = 8$ bit, 量化间隔 $\Delta = 0.0156$; ② 对 MRBI-MLGD 译码算法, 设置缩放因子 $\alpha = 3.1$; ③ 对算法 1, 设置修正系数 $\beta = 0.32258$; ④ 对算法 2 采用非均匀量化, 设置量化比特 $b = 4$ bit, $r = 0.88$ 。所有算法中, 最大仿真迭代次数均设置为 $I_{max} = 30$ 。译码 BER 性能仿真曲线如图 1 所示。

从图 1 可以看出, 在中高信噪比区域, 算法 1 与 MRBI-MLGD 算法拥有几乎相同的译码性能, 而相较于 RBI-MLGD 算法具有约 0.3 dB 的性能增益, 与 SPA 译码算法仅有约 0.6 dB 的译码性能差距; 算法 2 由于采用了较低的量化比特, 译码性能稍差于算法 1, 但仍然明显优于 RBI-MLGD 算法。

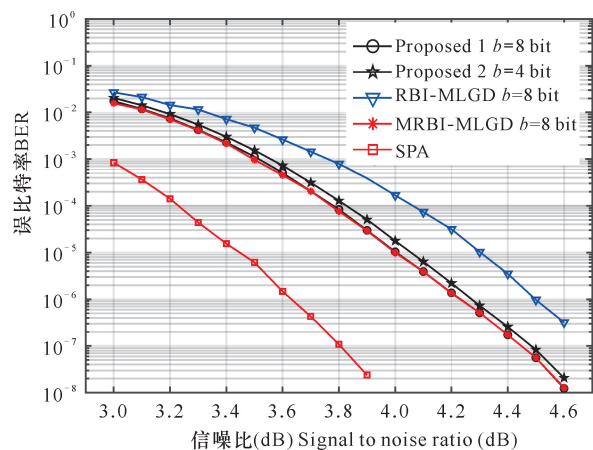
图1 对 $F_2(1023,781)$ LDPC 码的译码 BER 性能比较

Fig. 1 BER performance comparison of $F_2(1023,781)$ LDPC code

图2显示了对 $F_2(1032,781)$ LDPC 码进行译码的平均迭代次数,在中高信噪比区域,本文算法均有着与 MRBI-MLGD 算法一致的迭代收敛速度,且收敛速度明显优于 RBI-MLGD 算法。

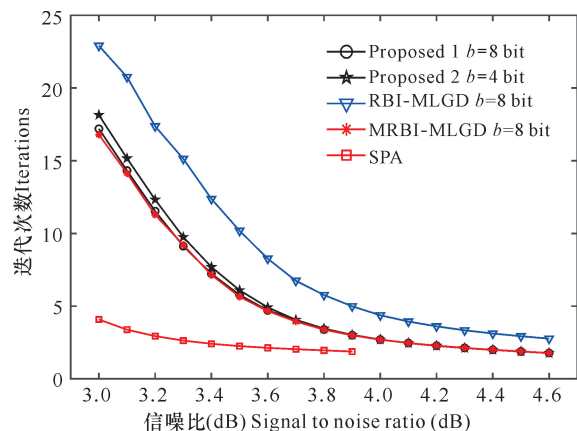
图2 对 $F_2(1023,781)$ LDPC 码的译码平均迭代次数比较

Fig. 2 Comparison of decoding average iterations of $F_2(1023,781)$ LDPC code

实验 2: 实验采用 $F_2(255,175)$ 规则循环 LDPC 码^[18] 进行仿真实验,其仿真参数采用以下设置:①对均匀量化,量化比特 $b = 8$ bit,量化间隔 $\Delta = 0.0156$;②对 MRBI-MLGD 译码算法,设置修正系数 $\alpha = 7.0$;③对算法 1,设置修正系数 $\beta = 0.143$;④对算法 2 采用非均匀量化,设置量化比特 $b = 4$ bit, $r = 0.88$ 。所有算法中,最大仿真迭代次数均设置为 $I_{\max} = 30$ 。译码 BER 性能仿真曲线如图 3 所示。

从译码性能曲线(图 3)可以看出,算法 1 在中信噪比范围区间与 MRBI-MLGD 算法几乎具有相同的译码性能,同时优于 RBI-MLGD 译码算法近 0.4 dB

的译码性能。与 SPA 译码算法具有约 0.7 dB 的性能差距;算法 2 在量化比特 $b = 4$ bit 的情况下,仍实现了与 MRBI-MLGD 算法几乎相近的译码性能,同时,与 SPA 算法有约 0.7 dB 的性能差距。而在高信噪比区域,本文算法仍具有较优异的译码性能,出现了译码性能曲线明显优于 MRBI-MLGD 算法的趋势。

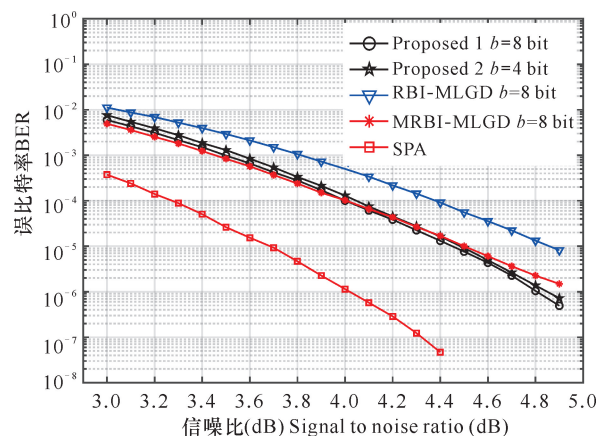
图3 对 $F_2(255,175)$ LDPC 码的译码 BER 性能比较

Fig. 3 BER performance comparison of $F_2(255,175)$ LDPC code

从译码平均迭代次数曲线(图 4)可以看出,算法 1 的译码迭代收敛速度明显高于 RBI-MLGD 译码算法,与 MRBI-MLGD 算法保持了一致的迭代收敛速度。此外,在较高信噪比时,其译码收敛速度几乎接近 SPA 算法。算法 2 在量化比特较低的情况下,仍然实现了较高的译码收敛速度,收敛速度性能与算法 1 一致。

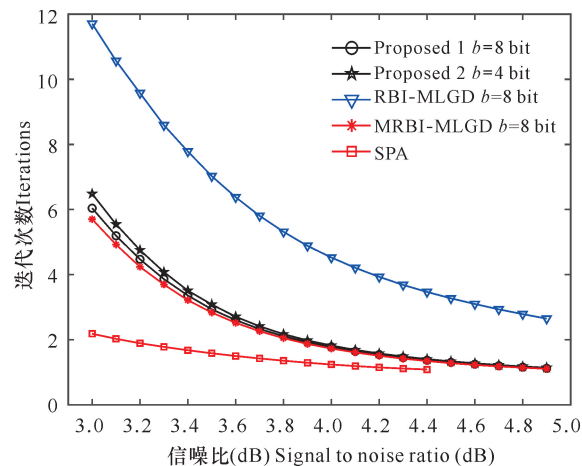
图4 对 $F_2(255,175)$ LDPC 码的译码平均迭代次数比较

Fig. 4 Comparison of decoding average iterations of $F_2(255,175)$ LDPC code

5 结束语

本文在对信道信息进行量化预处理时,通过引入量化修正的信息处理策略,提出了基于量化修正的低复杂度 LDPC 译码算法。该算法避免了在译码迭代过程中进行信息修正处理操作,较大幅度地降低了译码复杂度,同时保持了较好的译码性能。本文具体实现了算法 1 为一种基于修正系数的均匀量化方案及译码算法设计;算法 2 为一种基于列重修正的非均匀量化方案及译码算法设计。其中,算法 1 通过在信道信息量化预处理时,在量化函数中引入修正系数,避免在译码迭代中的浮点乘法运算,实现了译码迭代过程中计算复杂度的明显下降;算法 2 通过非均匀量化预处理时引入列重 γ 信息,实现了信道信息量化可靠度值与列重 γ 相适配的比拟关系,在迭代过程中获得更高效的译码信息传递和更新。仿真结果表明,在保持译码性能的情况下,本设计有效地降低了译码复杂度。

参考文献

- [1] GALLAGER R G. Low-density parity-check codes [J]. IRE Transaction on Information Theory, 1962, 8(1): 21-28.
- [2] GALLAGER R G. Low-density parity-check codes [M]. Cambridge: MIT Press, 1963.
- [3] MACKAY D J C, NEAL R M. Near Shannon limit performance of low density parity check codes [J]. Electronics Letters, 1996, 32(18): 1645-1646.
- [4] MACKAY D J C, NEAL R M. Near Shannon limit performance of low density parity check codes [J]. Electronic Letters, 1997, 33(6): 457-458.
- [5] MACKAY D J C. Good error-correcting codes based on very sparse matrices [J]. IEEE Transactions on Information Theory, 1999, 45(2): 399-431.
- [6] MACKAY D J C, WILSON S T, DAVEY M C. Comparison of constructions of irregular Gallager codes [J]. IEEE Transactions on Communications, 1999, 47(10): 1449-1454.
- [7] LIU X C, ZHOU Z Z, CUI R, et al. Informed decoding algorithms of LDPC codes based on dynamic selection strategy senior member [J]. IEEE Transactions on Communications, 2016, 64(4): 1357-1366.
- [8] 李锦明,王国栋,刘梦欣,等. CCSDS 标准下 LDPC 码的编译码算法研究 [J]. 电子学报, 2020, 48(11): 2114-2121.
- [9] 许欣. 5G URLLC 数据信道中 LDPC 编译码方法研究 [D]. 北京: 北京交通大学, 2020.
- [10] OUYANG S J, HAN G J, FANG Y, et al. LLR-distribution-based non-uniform quantization for RBI-MSD algorithm in MLC flash memory [J]. IEEE Communications Letters, 2018, 22(1): 45-48.
- [11] 胡东伟. 5G LDPC 码译码器实现 [J]. 电子与信息学报, 2021, 43(4): 1112-1119.
- [12] 陈海强. LDPC 编译码技术及其在 Relay 系统中的应用研究 [D]. 广州: 中山大学, 2011.
- [13] LIN S, COSTELLO D J. Error control coding: Fundamentals and applications [M]. 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2004.
- [14] LIU Y H, ZHANG M L. Hard-decision bit-flipping decoder based on adaptive bit-local threshold for LDPC codes [J]. IEEE Communications Letters, 2019, 23(5): 789-792.
- [15] HUANG Q, KANG J, ZHANG L, et al. Two reliability-based iterative majority-logic decoding algorithms for LDPC codes [J]. IEEE Transactions on Communications, 2009, 57(12): 3597-3606.
- [16] CHEN H Q, ZHANG K, MA X, et al. Comparisons between reliability-based iterative min-sum and majority-logic decoding algorithms for LDPC codes [J]. IEEE Transactions on Communications, 2011, 59(7): 1766-1771.
- [17] SONG S W, CUI H X, TIAN J, et al. A novel iterative reliability-based majority-logic decoder for NB-LDPC codes [J]. IEEE Transactions on Circuits and Systems II: Express Briefs, 2020, 67(8): 1399-1403.
- [18] KOU Y, LIN S, FOSSORIER M P C. Low-density parity-check codes based on finite geometries: A discovery and new results [J]. IEEE Transactions on Information Theory, 2001, 47(7): 2711-2736.

Low Complexity LDPC Decoding Algorithm Based on Quantification Correction

DU Lichan¹, HUANG Yihui², WANG Wenjing¹, NIE Jing¹, LI Xiangcheng^{2,3}

(1. School of Artificial Intelligence, Nanning College for Vocational Technology, Nanning, Guangxi, 530008, China; 2. School of Computer, Electronics and Information, Guangxi University, Nanning, Guangxi, 530004, China; 3. Guangxi Key Laboratory of Multimedia Communications and Network Technology, Nanning, Guangxi, 530004, China)

Abstract: In order to further reduce the complexity of reliability-based LDPC decoding algorithm, a low complexity LDPC decoding algorithm based on quantization correction is proposed. The algorithm introduces quantization information correction strategy in channel information preprocessing, which avoids correcting decoding information during the decoding iteration. The decoding complexity is greatly reduced while maintaining the decoding performance. For the uniform and non-uniform quantization schemes, in this article, two decoding schemes of uniform quantization based on correction coefficient and non-uniform quantization based on column re-correction are realized. The simulation results show that the two proposed decoding schemes still have the same decoding performance and convergence speed as the MRBI-MLGD decoding algorithm under the condition of significantly lower algorithm complexity. In particular, in the non-uniform quantization scheme based on column re-correction, only 3–4 bits of low-bit quantization can achieve the same number of iterations and bit error rate performance as medium-high-bit quantization (8 bits) in uniform quantization.

Key words: low-density parity-check codes; low complexity; reliability; quantization correction; pre-processing

责任编辑: 陆媛峰



微信公众号投稿更便捷

联系电话: 0771-2503923

邮箱: gxkx@gxas.cn

投稿系统网址: <http://gxkx.ijournal.cn/gxkx/ch>