

5G 低密度奇偶校验码的低复杂度偏移最小和算法

陈发堂, 张友寿*, 杜 铮

(重庆邮电大学 通信与信息工程学院, 重庆 400065)

(* 通信作者电子邮箱 1375743499@qq.com)

摘 要: 为了提高低密度奇偶校验(LDPC)码偏移最小和(OMS)算法的误码性能, 基于 5G NR 标准提出了一种 5G LDPC 码的低复杂度 OMS 算法。针对传统算法中偏移因子值计算不够准确问题, 使用密度进化获取更加精准的偏移因子值, 用于校验节点更新, 以增强 OMS 算法的性能; 并使用线性近似方法对获得的偏移因子值进行近似处理, 在保证译码性能的情况下降低了算法的复杂度。针对变量节点振荡现象对译码的影响, 将节点更新前后的对数似然比(LLR)消息值加权处理, 削减变量节点的振荡性, 提高了译码器收敛速度。仿真结果表明, 与归一化最小和(NMS)算法和 OMS 算法相比, 在误比特率(BER)为 10^{-5} 时所提算法译码性能可以获得 0.3~0.5 dB 的增益, 平均迭代次数分别降低了 48.1% 和 24.3%, 同时与对数似然比-置信传播(LLR-BP)算法也只相差近 0.1 dB。

关键词: 5G; 低密度奇偶校验码; 偏移最小和; 密度进化; 振荡性; 线性近似

中图分类号: TN911.22 **文献标志码:** A

Low complexity offset min-sum algorithm for 5G low density parity check codes

CHEN Fatang, ZHANG Youshou*, DU Zheng

(College of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: In order to improve the error code performance of Low Density Parity Check (LDPC) code Offset Min-Sum (OMS) algorithm, a low complexity OMS algorithm for 5G LDPC codes was proposed based on 5G NR standard. Aiming at the problem that the offset factor value calculation in the traditional algorithm is not accurate enough, the density evolution was used to obtain a more accurate offset factor value, which was used to the check node updating in order to enhance the performance of OMS algorithm. And the obtained offset factor value was approximated by using the linear approximation method, so as to reduce the complexity of the algorithm while ensuring decoding performance. For the influence of the variable node oscillation phenomenon on the decoding, the Log-Likelihood Ratio (LLR) message values before and after node updating were weighted, so the oscillation of the variable node was reduced, and the convergence speed of the decoder was improved. The simulation results show that compared with Normalized-Min-Sum (NMS) algorithm and OMS algorithm, the proposed algorithm improves the decoding performance by 0.3–0.5 dB when the Bit-Error Rate (BER) is 10^{-5} , and the average iteration times reduced by 48.1% and 24.3% respectively. At the same time, the difference between the performance of the proposed algorithm and LLR-BP (Log-Likelihood Ratio-Belief Propagation) algorithm performance is only nearly 0.1 dB.

Key words: 5th-Generation (5G); Low Density Parity Check (LDPC) code; Offset-Min-Sum (OMS); Density Evolution (DE); oscillation; linear approximation

0 引言

低密度奇偶校验(Low Density Parity Check, LDPC)码作为一种最佳纠错码之一, 具有显著的编码性能和效率, 近年来受到了广泛的研究关注^[1]。在许多情况下, 它展现出比 Turbo 码更好的译码性能^[2-3], 因此 LDPC 码被多种通信标准采用^[4]。此外, 已经确认在增强移动宽带(enhanced Mobile BroadBand, eMBB)场景的 5G 系统中采用 LDPC 码, 满足 5G 新无线(New Radio, NR)的高吞吐量需求。

LDPC 码是具有稀疏奇偶校验矩阵的线性分组码, 其校验矩阵可以用 Tanner 图描述^[5], 图中包含变量节点和校验节点,

变量节点和校验节点之间的连接用边表示。置信传播(Belief Propagation, BP)算法, 也称为和积(Sum-Product, SP)算法^[6]通过迭代使 LDPC 码的译码性能接近香农极限。信息在变量节点和校验节点之间反复传递。SP 算法虽然性能好, 但是需要对数等复杂运算, 给硬件实现带来了困难。为了降低复杂度, 提出了最小和(Min-Sum, MS)算法, 将校验节点中的对数转换为比较和求和来简化复杂的计算^[7]。与 SP 算法相比, MS 算法存在扩大校验节点消息幅度值的错误, 导致性能出现损失。为了减少错误, 提高性能, 提出了两种改进算法, 分别称为归一化最小和(Normalized-MS, NMS)算法和偏移最小和

收稿日期: 2019-11-07; 修回日期: 2020-01-24; 录用日期: 2020-02-14。

基金项目: 国家科技重大专项(2017ZX03001021-004)。

作者简介: 陈发堂(1965—), 男, 重庆人, 研究员, 主要研究方向: 移动通信系统; 张友寿(1993—), 男, 湖北武汉人, 硕士研究生, 主要研究方向: 5G 物理层协议与算法; 杜铮(1997—), 男, 河南新乡人, 硕士研究生, 主要研究方向: 5G 物理层协议与算法。

(Offset-MS, OMS)算法^[8]。OMS算法使用求和系数改进MS中校验节点的结果;NMS算法中使用乘法系数,虽然提高了译码性能,但是引入的修正系数与实际期望因子值的差异使得译码性能不如BP算法。而且MS和两种改进算法要求在矩阵的每一行中计算第一和第二最小值,增加了复杂度。为进一步降低复杂性:单最小值最小和(single-minimum MS, smMS)算法仅使用一个常量纠正系数简化了第二最小值计算,却损失了部分性能^[9];在此基础上提出的简化可变权最小和(simplified variable weight MS, svwMS)算法^[10-11],引入随迭代次数变化的加权因子维持第一和第二最小值的比例,性能得到了提升,却增加了复杂度。

另一方面,近年来人们对LLR-BP译码算法的错误类型进行了研究^[12],其中一种就是变量节点的振荡性。这种振荡性被视为变量节点更新前后LLR消息值符号发生了变化,最终导致译码器无法收敛^[13-14]。变量节点的振荡现象与检验矩阵的结构相关而与译码算法无关,出现在几乎所有的BP算法中而且并没有得到足够的重视。

为了解决以上问题,一种新的改进偏移最小和的算法被提出。密度进化理论(Density Evolution, DE)是分析LDPC码译码算法性能,确定关键参数最佳值,并评估有限量化效果的一种重要方法。本文使用密度进化理论计算最佳的偏移因子值,克服了传统OMS算法的不足,提高了译码性能,并使用线性近似方法对获取的最佳偏移因子进行近似处理,降低算法的实现复杂度。现有BP算法中变量节点振荡性问题也被讨论,并提出了一种处理振荡性的方法,将节点更新前后的LLR消息值加权处理。仿真结果说明与其他改进MS算法相比,所提算法不仅提高了译码器收敛速度,而且改进了在中高信噪比下的纠错性能,且具有复杂度较低的优势。

1 LDPC码及其译码算法

1.1 LDPC码的表示方法

5G NR标准LDPC码的校验矩阵 H 包括 N 列和 M 行,其中 $M = N - K$, N 代表变量节点数, M 代表校验节点数,可由基矩阵 H_{BG} 定义。 H_{BG} 包含 M_b 行和 N_b 列,基矩阵每个元素都映射到 H 中的一个维度为 $Z_c \times Z_c$ 子块矩阵。用全0矩阵去替换 H_{BG} 中的-1值,用循环置换矩阵 $I(p_{i,j})$ 替换 H_{BG} 中的1值,其中 $I(p_{i,j})$ 是维数为 $Z_c \times Z_c$ 的单位矩阵右循环 $p_{i,j}$ 次获得, Z_c 是矩阵的扩展因子。LDPC码的校验矩阵也可用Tanner图表示,如果 $H(i, j) = 1$,表示在Tanner图中第 i 校验节点与第 j 变量节点之间相连。假设校验矩阵如下:

$$H = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}$$

对应的Tanner图如图1,其中 c_i 和 v_i 分别表示校验节点和变量节点。令 $N(m) = \{n: H(m, n) = 1\}$ 表示与校验节点 m 相连的变量节点集合, $M(n) = \{m: H(m, n) = 1\}$ 表示与变量节点 n 相连的校验节点集合,同时让 $N(m) \setminus n$ 表示排除变量节点 n 的变量节点集合, $M(n) \setminus m$ 表示排除校验节点 m 的变量节点集合。把校验节点 m 传递给变量节点 n 的消息称为C2V(Check-to-Variable)消息,用 $L_{m \rightarrow n}$ 表示;同理把变量节点 n 传递给校验节点 m 的消息称为V2C(Variable-to-Check)消息,用

$Z_{n \rightarrow m}$ 表示。

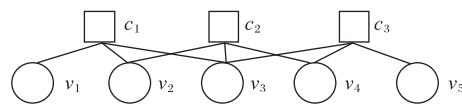


图1 LDPC码的Tanner图表示

Fig. 1 Tanner graph representation of LDPC codes

1.2 LLR-BP算法

LLR-BP算法的核心是对校验节点和变量节点之间传递的外部消息进行迭代处理,并利用校验方程的约束提高译码的可靠度。译码步骤^[15]如下。

1) 初始化: 计算变量节点 v_n 接收到的信道消息值 $l_n = \frac{2}{\sigma^2} y_n$ 。

2) 迭代过程如下:

校验节点更新(C2V):

$$L_{m \rightarrow n}(x_n) = \left(\prod_{n' \in N(m) \setminus n} \text{sign}(Z_{n' \rightarrow m}(x_{n'})) \right) \times 2 \tanh^{-1} \left(\prod_{n' \in N(m) \setminus n} \tanh \left(\frac{|Z_{n' \rightarrow m}(x_{n'})|}{2} \right) \right) \quad (1)$$

变量节点更新(V2C):

$$Z_{n \rightarrow m}(x_n) = l_n + \sum_{m' \in M(n) \setminus m} L_{m' \rightarrow n}(x_n) \quad (2)$$

3) 硬判决:

$$Z_n(x_n) = l_n + \sum_{m \in M(n)} L_{m \rightarrow n}(x_n) \quad (3)$$

$$\hat{x}_n = \begin{cases} 0, & Z_n(x_n) \geq 0 \\ 1, & Z_n(x_n) < 0 \end{cases} \quad (4)$$

4) 终止条件: 如果满足 $\mathbf{X} \cdot \mathbf{H}^T = \mathbf{0}$ 或者达到最大迭代次数时译码终止并返回译码结果;否则重复第2)步。

1.3 MS和OMS算法

MS算法是对BP算法迭代过程中的C2V消息进行简化,其余处理不变^[16],如式(5)所示。显然MS算法计算比较简单,但降低了译码性能。OMS算法对MS算法进行了改进如式(6)^[17],使得该算法的译码性能接近BP算法。

$$L_{m \rightarrow n}(x_n) = \left(\prod_{n' \in N(m) \setminus n} \text{sign}(Z_{n' \rightarrow m}(x_{n'})) \right) \times \min_{n' \in N(m) \setminus n} |Z_{n' \rightarrow m}(x_{n'})| \quad (5)$$

$$L_{m \rightarrow n}(x_n) = \left(\prod_{n' \in N(m) \setminus n} \text{sign}(Z_{n' \rightarrow m}(x_{n'})) \right) \times \max \left(\min_{n' \in N(m) \setminus n} |Z_{n' \rightarrow m}(x_{n'})| - \beta, 0 \right) \quad (6)$$

2 改进的OMS算法

2.1 基本知识

当把C2V消息表示为符号和幅值时,从译码准则可以看出,符号用来译码,幅值表示消息的可靠性。与LLR-BP算法相比,MS算法高估了C2V消息的幅值,所以译码性能下降。通过引入一个常数 β 来改善MS算法中的校验节点幅值过大的错误,被称为OMS算法。让 L_1 和 L_2 分别代表LLR-BP算法和MS算法的C2V消息如式(7)和(8):

$$L_1 = 2 \tanh^{-1} \left(\prod_{n' \in N(m) \setminus n} \tanh \left(\left| Z_{n' \rightarrow m}(x_{n'}) \right| / 2 \right) \right) \quad (7)$$

$$L_2 = \left(\prod_{n' \in N(m) \setminus n} \text{sign} \left(Z_{n' \rightarrow m}(x_{n'}) \right) \right) \times \min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right| \quad (8)$$

第一次迭代时的 β 可以通过式 (9) 计算得到, 其中 $E(\left| L_1 \right|)$ 和 $E(\left| L_2 \right|)$ 通过式 (10) 和式 (11)^[18] 计算:

$$\beta = E(\left| L_1 \right|) - E(\left| L_2 \right|) \quad (9)$$

$$E(\left| L_1 \right|) = 2 \sum_{n=1}^{\infty} \frac{E \left(\left(\tanh \left(Z_{n \rightarrow m} / 2 \right) \right)^{2^{d_c}-1} \right)}{2n-2} \quad (10)$$

其中: d_c 代表检验节点的度, $E(\left| L_1 \right|)$ 由无穷项的和组成, 但通常取前几次之和用作近似^[18]。

$$E(\left| L_2 \right|) = \int_0^{\mu} \left[1 - Q \left(\frac{\mu - y}{\sigma} \right) + Q \left(\frac{\mu + y}{\sigma} \right) \right]^{d_c-1} dy \quad (11)$$

$Q(\cdot)$ 代表 V2C 消息的概率密度函数 (Probability Distribution Function, PDF), 由信道消息初始化: $Q(x) = (1/\sqrt{2\pi}) \int_x^{+\infty} e^{-(x^2/2)} dx$, $\mu = 4/N_0$, $\sigma^2 = 8/N_0$ 。

当使用式 (11) 计算 $E(\left| L_2 \right|)$, 一旦迭代开始, V2C 消息发生了改变, 其概率密度函数 (PDF) 也会变化, 并且这种变化的 PDF 无法获取, 因此 β 的值只能通过第一次迭代由式 (9) 计算。

一般做法, 会在所有的迭代中使用相同的 β 值以达到简化译码处理的效果。然而, 真实的 β 值与迭代次数有关。如果 β 的值随着迭代次数变化而变化并用于迭代译码, 这样才能彻底改进译码性能, 因此提出一种基于密度进化 (Density Evolution, DE) 计算每次迭代 β 值的方法。

在引进密度进化 (DE) 理论之前, 说明两点: 1) 在使用 DE 理论进行求值时, 有必要对消息进行量化。需要的是 V2C 消息的概率质量函数 (Probability Mass Function, PMF) 而不是概率密度函数^[19]。2) 使用 Matlab 工具可以获取每次迭代中 V2C 消息的 PMF。

2.2 基于 DE 的最优 β 值计算

首先, 用 L_3 表示 OMS 算法中 C2V 消息, 如式 (12), 其中 k 代表迭代次数:

$$L_3 = \left(\prod_{n' \in N(m) \setminus n} \text{sign} \left(Z_{n' \rightarrow m}(x_{n'}) \right) \right) \times \max \left(\min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right| - \beta_k, 0 \right) \quad (12)$$

均值 $E(\left| L_3 \right|)$ 由式 (13) 计算得到:

$$E(\left| L_3 \right|) = E \left[\max \left(\min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right| - \beta_k, 0 \right) \right] \quad (13)$$

令 $X = \max \left(\min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right| - \beta_k, 0 \right)$, $M = \min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right|$, 所以 X 的离散概率分布函数由式 (14) 计算:

$$\begin{aligned} F_X(x) &= P(X \leq x) = P(\max(M - \beta_k) \leq x) = \\ &P(M - \beta_k \leq x, 0 \leq x) = P(0 \leq x) \cdot \\ &P \left(\min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right| \leq x + \beta_k \right) = \\ &\left\{ 1 - \left[1 - P \left(\left| Z_{n' \rightarrow m}(x_{n'}) \right| \leq x + \beta_k \right) \right]^{d_c-1} \right\} \cdot \\ &P(0 \leq x) \end{aligned} \quad (14)$$

每个量化点的 PMF 由式 (15) 计算, 其中 s 代表量化步长:

$$P(x = i) = F_X(x = i) - F_X(x = i - s) \quad (15)$$

最后, 均值 $E(\left| L_3 \right|)$ 和最优的偏移因子 β 可由式 (16)、(17) 计算得到:

$$\begin{aligned} E(\left| L_3 \right|) &= E \left[\max \left(\min_{n' \in N(m) \setminus n} \left| Z_{n' \rightarrow m}(x_{n'}) \right| - \beta_k, 0 \right) \right] = \\ E[X] &= \sum_i i \times P(x = i) \end{aligned} \quad (16)$$

$$\beta_k = \begin{cases} E(\left| L_2 \right|) - E(\left| L_1 \right|), & k = 1 \\ E(\left| L_3 \right|) - E(\left| L_1 \right|), & k \geq 2 \end{cases} \quad (17)$$

2.3 迭代次数与不同的译码算法

基于 DE 理论计算得到的偏移因子 β_k 与迭代次数的关系如图 2 所示。在相同的信噪比下经过几次迭代过后, β_k 的值越来越小并收敛于 0。这说明 L_3 的概率密度函数非常接近 L_1 的概率密度函数, 而且偏移因子的值对译码的影响越来越小。说明本文所提译码算法是正确的。

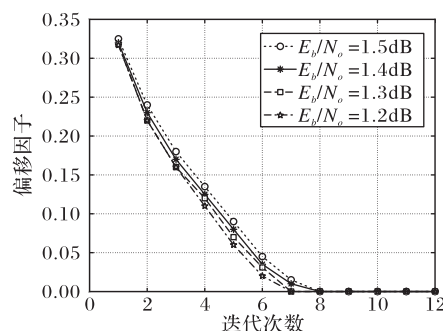


图 2 偏移因子与迭代次数的关系

Fig. 2 Relationship between offset factor and number of iterations

理论上, 在不同的信噪比下每次迭代使用不同的 β_k , 则译码性能能够得到提升。然而这会增加硬件复杂度。为了降低复杂度, 对前 5 次迭代的 β_k 值采用加权平均处理^[20]。改进的偏移因子的值如式 (18), 其中 λ_k 是加权平均系数, 对应分别设置为 $\lambda_1 = 0.25$, $\lambda_2 = 0.25$, $\lambda_3 = 0.2$, $\lambda_4 = 0.2$, $\lambda_5 = 0.1$ 。使用该方法改进的 β 被称为 DEOMS-1 算法。

$$\beta = \sum_{k=1}^5 \lambda_k \beta_k \quad (18)$$

另外从图 2 可以看出, 在每个信噪比下偏移因子与迭代次数的曲线斜率近似于常量, 并且易于计算。于是提出一种线性近似方法, 这种关系可写为:

$$\beta_k = \gamma_k \beta_1 \quad (19)$$

于是式 (18) 可以近似等价于式 (20):

$$\beta = \sum_{k=1}^5 \gamma_k \lambda_k \beta_1 \quad (20)$$

根据上面的分析,线性因子 $\sum_{k=1}^5 \gamma_k \lambda_k$ 被当成一个常量计算

且取值为0.658。这种改进的参数被用于之后的迭代译码,即对所有信噪比在所有迭代中使用相同的参数,既降低了硬件复杂度又确保了译码性能。这种算法在本文中被称为DEOMS-2算法。

由于LDPC码的校验矩阵中变量节点存在两个及以上的环,同时节点正确和错误的LLR消息在这些环中循环传递并在不同时刻到达该变量节点,造成变量节点振荡现象,导致译码器无法收敛。该现象与校验矩阵的结构有关,出现在几乎所有的BP译码算法中。然而最近所提出的译码算法并未采用特定方法消除变量节点的振荡性。当变量节点发生振荡,很难区分更新前后的LLR消息哪个是正确的,或者达到稳定状态。为了减弱其对迭代译码的影响,本文对变量节点更新前后的LLR消息进行加权处理,减弱变量节点LLR消息的振荡,提高译码的收敛性。具体处理规则如下:

$$Z_{n \rightarrow m}(x_n) = Z_{n \rightarrow m}^{(k-1)}(x_n) + \mu (Z_{n \rightarrow m}^{(k)}(x_n) - Z_{n \rightarrow m}^{(k-1)}(x_n)) = (1 - \mu) \times Z_{n \rightarrow m}^{(k-1)}(x_n) + \mu \times Z_{n \rightarrow m}^{(k)}(x_n) \quad (21)$$

用式(21)计算得到的 $Z_{n \rightarrow m}(x_n)$ 值替换式(2),其中 k 为迭代次数, μ 是加权系数,经实验本文最终选取 μ 等于0.5,则式(21)可以简化为:

$$Z_{n \rightarrow m}(x_n) = 0.5 \times (Z_{n \rightarrow m}^{(k-1)}(x_n) + Z_{n \rightarrow m}^{(k)}(x_n)) \quad (22)$$

该方法稍微增加了复杂度,但可以很好地降低振荡影响,加快收敛。基于以上方法提出vwDEOMS-3算法。

3 仿真结果分析

本文使用5G NR标准构造的准循环LDPC(Quasi Cyclic Low Density Parity Check, QC-LDPC)码对不同译码算法LLR-BP、MS、NMS、OMS的译码性能进行仿真并与本文所提出的低复杂度偏移最小和算法的译码性能进行比较。选择5G NR LDPC码的基矩阵 H_{BG2} , 采用信息长度 $K=1\,000$, 码字长度 N 为2500, 码率 R 为0.4, 评估了不同信噪比下各译码算法的误码率性能。在码率为0.4时NMS算法的归一化因子取值0.65。所有的仿真都在加性高斯白噪声(Additive White Gaussian Noise, AWGN)信道模型下,经过BPSK(Binary Phase Shift Keying)调制,最大迭代次数设置为30,每个信噪比下设置帧数为2000。

图3展示了不同译码算法的误比特率(Bit-Error Rate, BER)性能比较,显然LLR-BP译码算法拥有最佳的性能。在BER为 10^{-5} 时,相对于LLR-BP算法,MS算法性能损失接近1.2 dB,这是校验节点的幅值过大造成的。与MS算法相比,OMS算法与NMS算法的译码性能可以获得0.6~0.8 dB的增益,但是还与LLR-BP算法的译码性能相差0.4~0.6 dB,说明虽然引入了校正因子对MS算法进行改进,但是该校正因子并不是最佳的值。DEOMS-1算法使用更准确的偏移因子,在BER为 10^{-5} 时,相比MS算法可以获得大约1 dB的增益。在BER为 10^{-5} 时,DEOMS-2算法与DEOMS-1算法误码性能非常接近,大约只有0.01 dB的差距,说明线性近似处理是合理有效的;而vwDEOMS-3算法比OMS算法的性能提高约0.5 dB,比NMS算法提高约0.3 dB,与LLR-BP算法相比性能上也只相差近0.1 dB。通过使用更接近真实的偏移因子和削弱变量

节点的振荡性,所提算法可以实现较好的译码性能和迭代性能。

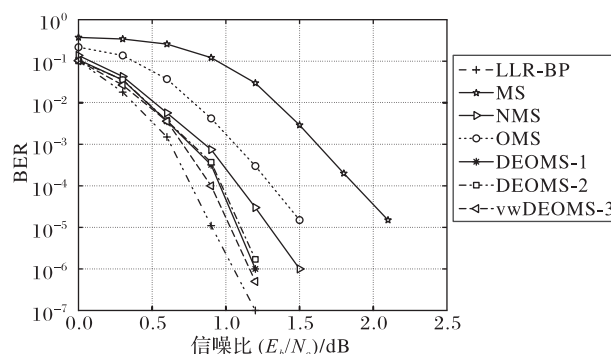


图3 不同算法的误比特率性能对比

Fig. 3 BER performance comparison of different algorithms

4 复杂度分析

结合上面的分析,表1给出了不同译码算法的校验节点更新操作中的计算复杂度。可以看出LLR-BP算法需要更多的乘法操作,复杂度最高,其次是MS算法、NMS算法、OMS算法。DEOMS-1算法中的偏移因子是基于密度进化计算的,增加了些许复杂度,但极大提高了译码性能,考虑到这一点,DEOMS-2算法采用线性近似方法,进一步降低计算的复杂度。vwDEOMS-3算法对变量节点的加权操作在硬件资源上并不需要多余的乘法器,只是信息位的右移。本文使用Matlab计算最优的偏移因子值并保存在硬件中,而不会消耗多余的硬件资源。经典的OMS算法通过蒙特卡洛方法获取偏移因子或者仅仅基于DE计算第一次迭代时得到的偏移因子的值,需要大量的加法和比较操作,计算复杂度总体来说较高。总的来说,DEOMS算法拥有较低的计算复杂度和较少的译码延迟,译码性能也有0.3~0.5 dB的提升。

表1 不同算法校验节点消息计算复杂度

Tab. 1 Check node message computational complexities of different algorithms

算法	乘法	加法	其他操作
LLR-BP	$2d_c$	—	$d_c \tanh(x), d_c \tanh^{-1}(x)$
MS	—	$d_c - 2 + \lceil \log d_c \rceil$	—
NMS	2	$d_c - 2 + \lceil \log d_c \rceil$	—
OMS	—	$d_c + \lceil \log d_c \rceil$	—
DEOMS	—	$d_c + \lceil \log d_c \rceil$	—

收敛速度是衡量LDPC算法复杂度和译码性能的另一个关键指标,图4使用平均迭代次数来评估各种算法的收敛速度。显然,相比其他算法,MS算法和OMS算法需要更多的迭代次数。可以观察到,与MS算法相比,所提出的DEOMS算法的平均迭代次数得到显著降低。尽管LLR-BP算法的迭代次数略低于所提的DEOMS算法,但由于其校验节点操作的复杂度较高,每次迭代都可能花费大量的计算时间,而DEOMS算法拥有较小的译码延迟。从图中可得到的另一个重要结论是DEOMS-2算法的平均迭代次数非常接近DEOMS-1算法,这证明DEOMS-2算法对迭代次数的影响非常轻微。而vwDEOMS-3算法的平均迭代次数比DEOMS-1/2算法有所改进,说明该

算法是合理有效的。

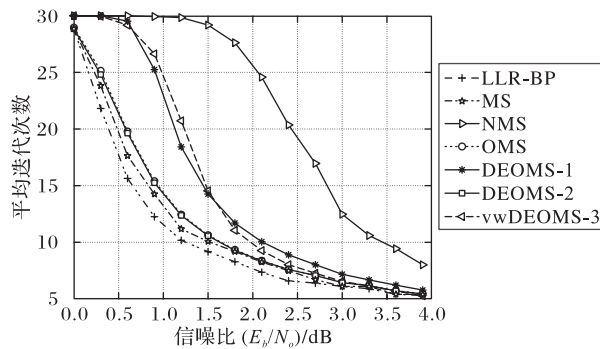


图4 不同算法的平均迭代次数

Fig. 4 Average numbers of iterations of different algorithms

5 结语

本文提出了一种5G LDPC码的低复杂度偏移最小和算法,所提算法的优点是使用密度进化理论获得了最佳的偏移因子,并对其使用线性近似方法进行处理,降低了计算复杂度;提出了一种处理变量节点振荡性的方法,对变量节点LLR消息加权处理,加快了译码器的收敛。仿真结果表明,对于5G NR标准的QC-LDPC码,所提算法与MS及其改进算法相比存在误码性能和迭代次数优势。

参考文献 (References)

- [1] GALLAGER R. Low-density parity-check codes[J]. Journal of Circuits and Systems, 2008, 8(1):3-26.
- [2] 张博,林伟,刘春元,等. 突发错误信道下的多元LDPC码设计与性能分析[J]. 通信学报, 2013, 34(7): 98-104. (ZHANG B, LIN W, LIU C Y, et al. On the design and performance of nonbinary LDPC codes on burst error channels[J]. Journal on Communications, 2013, 34(7): 98-104.)
- [3] 王冬梅,王秀芳,浦晓威,等. 基于Matlab的LDPC码的研究[J]. 科学技术与工程, 2010, 10(10):2472-2475. (WANG D M, WANG X F, PU X W, et al. Study of LDPC codes based on Matlab[J]. Science Technology and Engineering, 2010, 10(10):2472-2475.)
- [4] HAILES P, XU L, MAUNDER R G, et al. A flexible FPGA-based quasi-cyclic LDPC decoder[J]. IEEE Access, 2017(5): 20965 - 20984.
- [5] 朱庆,吴乐南,杨永标,等. Tanner图中基于矩阵运算的短环分布高效计算方法[J]. 通信学报, 2017, 38(4): 76-85. (ZHU Q, WU L N, YANG Y B, et al. Efficient algorithm for calculating short cycles in Tanner graph based on matrix computation[J]. Journal on Communications, 2017, 38(4):76-85.)
- [6] RICHARDSON T J, URBANKE R L. The capacity of low-density parity check codes under message-passing decoding [J]. IEEE Transactions on Information Theory, 2001, 47(2):599-618.
- [7] 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.
- [8] ZHANG J, FOSSORIER M, GU D. Combined normalized and off-set min-sum decoding algorithm for irregular LDPC codes[EB/OL]. [2019-06-20]. <https://pdfs.semanticscholar.org/87a5/9371d718c1877d9fe25d3f251d017daaca7c.pdf>.

9371d718c1877d9fe25d3f251d017daaca7c. pdf.

- [9] LI H, GUO J, GUO C. A low-complexity min-sum decoding algorithm for LDPC codes [C]// Proceedings of the IEEE 17th International Conference on Communication Technology. Piscataway: IEEE, 2017: 102-105.
- [10] ANGARITA F, VALLS J, ALMENAR V, et al. Reduced-complexity min-sum algorithm for decoding LDPC codes with low error-floor [J]. IEEE Transactions on Circuits and Systems I: Regular Papers, 2014, 61(7):2150-2158.
- [11] LEE H C, LI M R, HU J K, et al. Optimization techniques for the efficient implementation of high-rate layered QC-LDPC decoders [J]. IEEE Transactions on Circuits and Systems I: Regular Paper, 2017, 64(2): 457-470.
- [12] LECHNER G, SAYIR J. On the convergence of log-likelihood values in iterative decoding [EB/OL]. [2019-06-20]. <https://pdfs.semanticscholar.org/d0db/5dd40d144c003590b8ab377789f2bc7e7b29.pdf>.
- [13] KIM J H, NAM M Y, SONG H Y. Variable-to-check residual belief propagation for LDPC codes[J]. Electronics Letters, 2009, 45(2):117-118.
- [14] LEE H C, UENG Y L, YEH S M, et al. Two informed dynamic scheduling strategies for iterative LDPC decoders[J]. IEEE Transactions on Communications, 2013, 61(3):886-896.
- [15] ELEFTHERIOU E, MITTELHOLZER T, DHOLAKIA A. Reduced-complexity decoding algorithm for low-density parity-check codes[J]. Electronics Letters, 2001, 37(2):102-104.
- [16] 野晓东,马林华,王卫民,等. 基于整数运算的LDPC码最小和译码算法[J]. 通信学报, 2010, 31(6):112-117. (YE X D, MA L H, WANG W M, et al. Min-sum decoding algorithm for LDPC codes based on integer operation[J]. Journal on Communications, 2010, 31(6): 112-117.)
- [17] CAO Y. An improved LDPC decoding algorithm based on min-sum algorithm [C]// Proceedings of the 11th International Symposium on Communications and Information Technologies. Piscataway: IEEE, 2011: 26-29.
- [18] WANG X, CAO W, LI J, et al. Improved min-sum algorithm based on density evolution for low-density parity check codes[J]. IET Communications, 2017, 11(10): 1582-1586.
- [19] KAMENI NGASSA C, SAVIN V, DUPRAZ E, et al. Density evolution and functional threshold for the noisy min-sum decoder[J]. IEEE Transactions on Communications, 2015, 63(5):1497-1509.
- [20] WU Z, SU K, GUO L. A modified Min Sum decoding algorithm based on LMMSE for LDPC codes[J]. AEU — International Journal of Electronics and Communications, 2014, 68(10):994-999.

This work is partially supported by the National Science and Technology Major Project (2017ZX03001021-004).

CHEN Fatang, born in 1965, research fellow. His research interests include mobile communication system.

ZHANG Youzhou, born in 1993, M. S. candidate. His research interests include 5G physical layer protocol and algorithm.

DU Zheng, born in 1997, M. S. candidate. His research interests include 5G physical layer protocol and algorithm.