光电技术及应用

DOI: 10.16818/j.issn1001－5868.20251111002

一种LDPC码的自适应分层阈值最小和译码算法

袁建国*,熊龙宇,胡 坤
(重庆邮电大学 集成电路学院,重庆 400065)

摘 要: 针对低密度奇偶校验码的最小和译码算法性能有限及传统译码算法收敛速度较慢等问题,提出一种自适应分层阈值最小和(Adaptive Layered Threshold Min-Sum, ALTMS)译码算法。该算法结合阈值归一化最小和译码算法与阈值偏移最小和译码算法的优势,通过动态调整修正因子,使其随迭代次数自适应变化。同时,采用分层调度方式,有效提升了算法收敛速度和译码性能。仿真结果表明,与分层调度下的四种阈值最小和译码算法相比,所提出的ALTMS译码算法实现了一定的性能增益,相比使用固定修正因子的最小和译码算法展现出更优越的纠错性能。

关键词: 低密度奇偶校验码;最小和译码算法;分层调度;修正因子;迭代次数
中图分类号: TN911.22 **文章编号:** 1001－5868(2026)01－0158－06

Adaptive Layered Threshold Min-Sum Decoding Algorithm for Low-Density Parity-Check Codes

YUAN Jianguo, XIONG Longyu, HU Kun
(School of Integrated Circuits, Chongqing University of Posts and Telecommunications, Chongqing 400065, CHN)

Abstract: In this study, an adaptive layered threshold min-sum (ALTMS) decoding algorithm is proposed to address challenges such as the limited performance of minimum-sum decoding algorithms and slow convergence of traditional decoding algorithms for low-density parity-check codes. This algorithm combines the advantages of the threshold normalization and threshold offset minimum-sum decoding algorithms. It dynamically adjusts the correction factor to adaptively change with the iteration number and adopts a hierarchical scheduling method. Thus, the convergence speed and decoding performance of the algorithm are effectively improved. The simulation results show that the proposed ALTMS decoding algorithm performs better than the four hierarchical-scheduling-based threshold minimum-sum decoding algorithms to some certain. In addition, its error correction performance is superior to that of the minimum-sum decoding algorithm utilizing a fixed correction factor.

Key words: low-density parity-check codes; min-sum decoding algorithm; layered scheduling; correction factor; number of iterations

0 引言

低密度奇偶校验(Low-Density Parity-Check, LDPC)码作为一类线性分组码,核心特征在于校验矩阵的稀疏特性[1-2]。其在极长码长条件下能够获得接近香农极限的纠错性能,因此被广泛纳入现代

收稿日期:2025－11－11.
基金项目:国家自然科学基金项目(U21A20447, 61971079).
*通信作者:袁建国

通信标准[3-4]。

LDPC码的译码方法主要包括硬判决与软判决两类。硬判决译码在硬件实现上较为简单，但由于其在译码过程中未充分挖掘信道概率信息，导致纠错能力有限；而软判决译码常与迭代译码结合，表现出优异的译码性能，其中典型代表是置信度传播（Belief Propagation，BP）译码算法。BP译码算法性能逼近香农极限，但在校验节点处理环节计算复杂度较高，对硬件实现构成了挑战[5]。因此，Fossorier等提出最小和（Min-Sum，MS）译码算法，以明显的性能损失为代价，显著降低了计算复杂度[6]。此后，为了在性能与复杂度之间取得更好平衡，研究者进一步提出归一化最小和（Normalized Min-Sum，NMS）译码算法和偏移最小和（Offset Min-Sum，OMS）译码算法[7-8]。这两种算法均基于线性后处理思想，通过引入预设参数的标量函数，对MS译码算法中产生的高估信息进行修正，在仅少量增加复杂度的前提下，有效改善了译码性能。

然而，NMS和OMS译码算法与最优的BP译码之间仍存在可观差距。因此在其后续发展中，文献[9]提出阈值归一化最小和（Threshold Normalized Min-Sum，TNMS）译码算法和阈值偏移最小和（Threshold Offset Min-Sum，TOMS）译码算法，通过设定阈值来选择性使用修正因子，改善了高信噪比区域的译码性能；文献[10]对TOMS译码算法进一步优化，通过引入局部最优选择机制，缓解了其在最小值近零处可能出现的性能损失；文献[11]设计了一种类拟合函数，依据变量节点信息最小值所处阈值范围对校验节点信息进行差异化更新，从而在性能上更接近BP译码算法。

尽管以上改进算法都在一定程度上提升了MS译码算法的性能，但仅采用固定的修正因子，难以在每次迭代中准确适配校验节点的信息更新，对性能提升幅度有限。为解决该问题，本文提出一种改进的自适应分层阈值最小和（Adaptive Layered Threshold Min-Sum，ALTMS）译码算法，通过使修正因子随迭代过程动态调整，在仅引入少量额外计算负担的前提下，有效提升了算法的译码性能。

1　LDPC译码算法

1.1　BP译码算法

概率域BP译码算法涉及大量乘法运算，计算复杂度高且数值不稳定；而LLR_BP译码算法则是在概率域BP译码算法基础上，通过引入对数似然比，将原本在概率域中进行的复杂乘法运算，转化为在对数域中更为简单的加法运算，从而显著降低了计算复杂度，有效缓解了BP译码算法硬件实现难的瓶颈。

LLR_BP译码的具体步骤为：

（1）初始化

在加性高斯白噪声（Additive White Gaussian Noise，AWGN）信道下进行初始化，由接收端收到的概率信息y_v和噪声方差σ^2计算变量节点v的初始信息，其值$L(x)^0$由先验对数似然比（Log-Likelihood Ratio，LLR）表示：

$$L(x)^0=\ln\frac{P(x=0|y_v)}{P(x=1|y_v)}=\frac{2y_v}{\sigma^2} \tag{1}$$

式中，$P(x=b|y_v)$表示发送比特x取值b时的后验概率。

初始化完成后，$L(x)^0$将作为变量节点传递给校验节点的先验信息，沿着Tanner图中变量节点与校验节点之间的边进行发送，即$M_{vc}=L(x)^0$。

（2）校验节点信息更新

$$M_{cv}=2\tanh^{-1}\left[\prod_{v'\in V_j\backslash v}\tanh\left(\frac{1}{2}M_{v'c}\right)\right] \tag{2}$$

式中，V_j表示连接到校验节点j上的所有变量节点的集合。

（3）变量节点信息更新

对各变量节点更新其后验概率$L(x)^n$，其值也将作为下一次迭代的先验对数似然比：

$$L(x)^n=L(x)^{n-1}+\sum_{c'\in C_i}M_{c'v}^n \tag{3}$$

式中，C_i表示连接到变量节点i上的所有校验节点的集合，$n-1$和n分别为上一次迭代和当前迭代的索引值。

此时，变量节点到校验节点的信息更新为：

$$M_{vc}^{n+1}=L(x)^{n-1}+\sum_{c'\in C_i\backslash c}M_{c'v}^n \tag{4}$$

式中，M_{vc}^{n+1}为本次迭代中变量节点的更新信息，同时也将作为下一次迭代中传入步骤（2）中的变量节点信息。

（4）译码判决

当前迭代下的后验概率信息更新如公式（5）所示：

$$L(x)^n=M_{vc}^{n+1}+M_{cv}^n \tag{5}$$

通过更新后的$L(x)^n$值进行译码判决，判决结果Q_v由公式(6)计算：

$$Q_v=\begin{cases}1, & L(x)^n<0\\ 0, & L(x)^n\geqslant 0\end{cases} \tag{6}$$

当译码判决结果满足校验方程或达到预设的最大迭代次数时，结束译码；否则继续下一次迭代。

1.2 最小和译码算法

LLR_BP译码算法虽然显著减少了乘法运算，但引入了tanh函数，导致逻辑资源被大量占用。为解决该问题，学者们提出了MS译码算法，其校验节点更新公式为：

$$M_{cv}=\min_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}) \tag{7}$$

MS译码算法在降低算法复杂度的同时，也带来了性能的大幅度损失，其主要原因是近似值总是大于BP译码算法中的连乘结果。为解决校验节点信息高估问题，将归一化因子$\alpha(0<\alpha<1)$和偏移因子$\beta(\beta>0)$引入校验节点更新公式，分别得到NMS译码算法和OMS译码算法，如公式(8)和(9)所示：

$$M_{cv}=\alpha\times\min_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}) \tag{8}$$

$$M_{cv}=\max\left[\min_{v'\in V_j\backslash v}(|M_{v'c}|)-\beta,0\right]\times\prod_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}) \tag{9}$$

1.3 阈值最小和译码算法

NMS和OMS译码算法在低信噪比区域的性能优于MS译码算法，但在高信噪比区域不一定有良好的性能。因此，文献[9]提出阈值最小和译码算法，通过设定变量节点信息最小值阈值来筛选不可靠信息进行修正，以提高算法在高信噪比区域的性能。

文献[10]在TOMS译码算法基础上，提出一种优化分层阈值最小和(Improved Layered Threshold Offset Min-Sum, ILTOMS)译码算法。该算法通过局部选择最优的NMS译码算法来规避TOMS译码算法可能的近零处损失，从而进一步提升了译码性能。其校验节点更新公式为：

$$M_{cv}=\begin{cases}\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}), & \text{if }\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)>\tau\\ \alpha\times\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}), & \text{if }\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)<\beta\\ \max\left[\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)-\beta,0\right]\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}), & \text{other}\end{cases} \tag{10}$$

文献[11]提出的分层类拟合修正最小和(Layered Class Fitting Modified Minimum Sum, LCFMMS)译码算法则通过对MS译码算法向下取线性近似，划分出三个变量节点信息最小值区间，并用线性函数拟合，使性能逼近BP译码算法。其校验节点更新公式为：

$$M_{cv}=\begin{cases}0.8\times\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}), & \text{if }\min\limits_{v'\in V_j\backslash v}(|M_{v'c}/2|)<0.53\\ \min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}), & \text{if }\min\limits_{v'\in V_j\backslash v}(|M_{v'c}/2|)\geqslant 1\\ 2\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c})\times\left[1.29\times\min\limits_{v'\in V_j\backslash v}(|M_{v'c}/2|)-0.26\right], & \text{other}\end{cases} \tag{11}$$

2 一种自适应分层阈值最小和译码算法

2.1 调度方式

消息调度策略是影响LDPC译码性能的一个关键因素。传统译码算法常采用泛洪调度，主要依赖于简单的矩阵运算和消息传递，无需复杂的调度控制逻辑，易于软件仿真实现。然而，泛洪调度收敛速度慢，在高信噪比环境下易陷入局部最优解，导致译码失败。此外，泛洪调度译码还需要更大的迭代开销，不利于硬件实现。

分层调度方式则是将校验矩阵以行或列进行分层，并以层为单位进行更新，极大提升了译码收敛速度和硬件效率。在分层译码算法中，每层的变量节点信息都是基于上一层更新信息计算得到，相较于泛洪调度，分层调度的LLR信息更新更为频繁，且能将最新产生的信息及时反馈。因此，为改善传统译码算法收敛速度较慢的问题，本文所提算

法采用分层调度方式以加快收敛。分层调度下变量节点更新公式优化为：

$$M_{vc}^{lj}=L(x)^{li}-M_{cv}^{n-1} \tag{12}$$

式中，lj为当前层的索引；li表示变量节点v上次更新所在层的索引。

此时当前层的对数似然比信息$L(x)^{lj}$由公式(13)计算：

$$L(x)^{lj}=M_{vc}^{lj}+M_{cv}^{n} \tag{13}$$

$$M_{cv}=\begin{cases}\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c}), & \text{if } \min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)>\tau\\ \alpha\times\prod\limits_{v'\in V_j\backslash v}\operatorname{sgn}(M_{v'c})\times\max\left[\min\limits_{v'\in V_j\backslash v}(|M_{v'c}|)-\beta,\ 0\right], & \text{else}\end{cases} \tag{14}$$

式中，阈值τ采用文献[10]所选用的值1.5；α和β是随迭代次数变化的参数，其函数更新公式通过多次仿真验证获得。

考虑到算法复杂度方面，所提算法的修正因子更新公式优先考虑线性函数，并通过大量仿真验证，仿真验证结果如图1所示，发现线性函数在低信噪比时对修正因子的更新能带来较好的译码性能。在高信噪比情况下，线性函数提升并不明显，甚至没有提升，而二次函数则有较好的译码效果。因此，在所提算法中，修正因子将根据信噪比区间选择不同更新公式：

$$\alpha=\alpha_1+(N-\alpha_1)\times(n/iter) \tag{15}$$

$$\beta=\begin{cases}\beta_1\times(n/iter), & L_ENB\\ \beta_1\times(1-n/iter)^2, & H_ENB\end{cases} \tag{16}$$

式中，n表示当前迭代次数，$iter$为预设的最大迭代次数；参数N选为NMS译码算法中的最优归一化因子，本文使用0.8[10]；α_1和β_1由网格搜索法获得，分别以3 dB和5 dB模拟低信噪比和高信噪比区域，对不同修正因子下的译码仿真情况进行统计，绘制出多次实验下的平均误比特率如图2与图3所示。

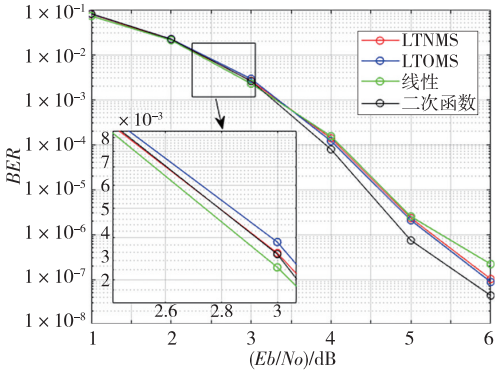


图1　修正因子更新函数的选择

2.2　改进的ALTMS译码算法

TNMS和TOMS译码算法虽然带来了一定性能提升，但因为修正因子固定，导致性能提升十分有限，甚至在一些极端情况下性能反而不如MS译码算法。鉴于此，考虑修正因子随迭代次数动态调整，并采用收敛更快的分层调度，提出一种ALTMS译码算法。该算法采用的校验节点更新公式优化为：

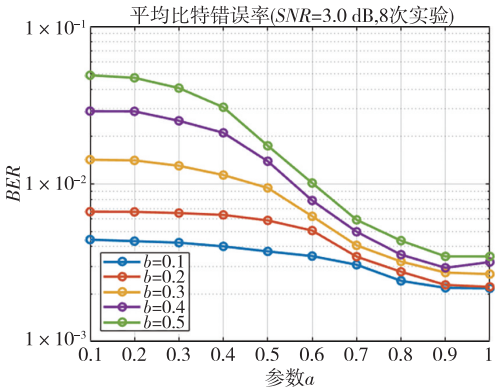


图2　低信噪比下修正因子的选择

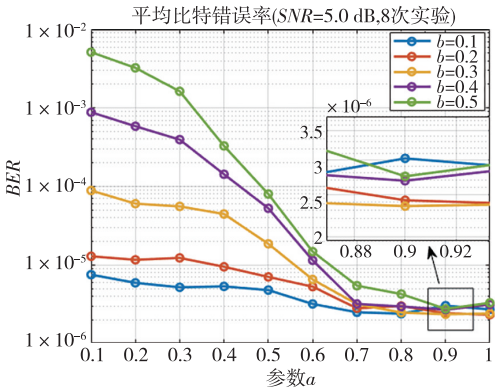


图3　高信噪比下修正因子的选择

在低信噪比区域，α_1和β_1分别取0.9和0.1时有较好的译码效果；在高信噪比下，α_1和β_1分别取0.9和0.3时有较好的译码效果。因此，本文选取α_1为0.9，β_1在低信噪比和高信噪比下分别为0.1和0.3。

具体算法为：

1）初始化：$L(x)=L(x)^0$，$M_{cv}^0=0$

2）if $ENB\in L_ENB$，$\alpha_1=0.9$，$\beta_1=0.1$

3）else $\alpha_1=0.9$，$\beta_1=0.3$

4）end if

5）迭代：for迭代次数$n=1:iter$

6）根据公式（15）和（16）计算α和β

7）for所有校验节点$c=1:Z$

8）for所有变量节点$v\in V_j$

9）加载$L(x)$

10）计算$M_{vc}^{l}=L(x)-M_{cv}^{l,n-1}$

11）根据公式（14）计算$M_{cv}^{l,n}$

12）更新$L(x)=M_{vc}^{l}+M_{cv}^{l,n}$

13）end for

14）end for

15）if不满足迭代终止条件

16）跳转到步骤6进行下一次迭代

17）end if

18）end for

3 仿真结果与复杂度分析

3.1 仿真结果分析

为验证所提算法性能上的优越性，将四种阈值最小和译码算法进行仿真对比分析。仿真平台采用Matlab软件，使用码长为155的准循环LDPC（155，64）码，其校验矩阵形式如下：

$$H=\begin{bmatrix} I_1 & I_2 & I_4 & I_8 & I_{16} \\ I_5 & I_{10} & I_{20} & I_9 & I_{18} \\ I_{25} & I_{19} & I_7 & I_{14} & I_{28} \end{bmatrix} \tag{17}$$

式中，I_x为一个31×31的单位矩阵，其中x表示单位矩阵行向左的循环移位数。

仿真时，选用二进制相移键控（Binary Phase Shift Keying，BPSK）调制和加性高斯白噪声信道，迭代次数选用50次。对比算法中涉及的归一化因子采用0.8，偏移因子选用0.2[10]，对所提算法与分层阈值归一化最小和（Layered Threshold Normalized Min-Sum，LTNMS）、分层阈值偏移最小和（Layered Threshold Offset Min-Sum，LTOMS）译码算法进行对比仿真，结果如图4所示。分析可知，当误码率为1×10^{-6}时，ALTMS译码算法比LTNMS译码算法和LTOMS译码算法在性能上分别提升了0.22 dB和0.14 dB。

如图5为ALTMS译码算法在LDPC（155，64）码时与两种改进的分层阈值最小和译码算法的纠错性能对比。分析可知，ALTMS译码算法相较于两种译码算法在性能上更优异，且在误码率为1×10^{-6}时，ALTMS译码算法性能分别比ILTOMS算法和LCFMMS算法提升了0.06 dB和0.14 dB。

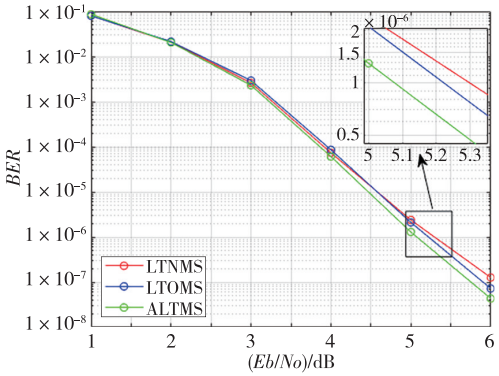


图4 LDPC（155，64）码中3种译码算法性能对比

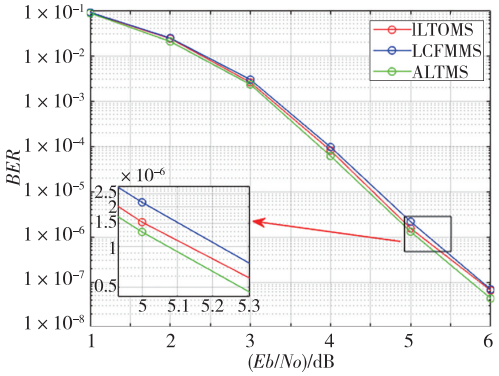


图5 ALTMS算法与其他两种阈值译码算法的纠错性能对比

为说明所提算法在不同码长码率的LDPC码下译码性能的优越性，使用码长为256，码率为1/2的LDPC码进行对比仿真分析，结果如图6所示。由图6可知，在误码率为1×10^{-7}时，ALTMS译码算法相比LCFMMS译码算法和ILTOMS译码算法分别有0.13 dB和0.18 dB的增益。

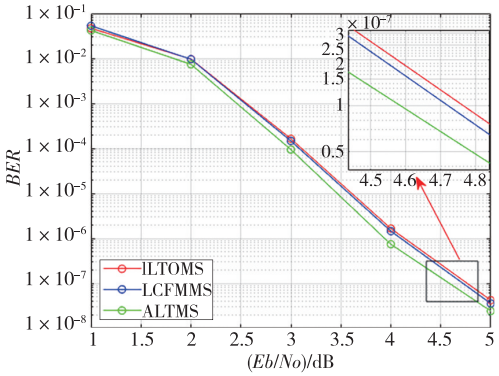


图6 LDPC（256，128）码中3种译码算法的纠错性能对比

3.2 复杂度分析

对本文涉及的五种译码算法，在单次迭代下分析各自所需的运算次数如表1所示。其中m表示校

验矩阵行数，w_r表示行重。由表1可知，在单次迭代中，本文所提的ALTMS译码算法相较于LTNMS和LTOMS译码算法，各运算次数有少量增加，而对比于另外两种改进阈值最小和译码算法复杂度甚至略有下降。因此，所提算法复杂度相较于同类分层阈值最小和译码算法并未有明显增加。

表1　不同译码算法复杂度分析

译码算法	乘法运算	加法运算	比较运算
LTNMS	$4\times m\times w_r$	$2\times m\times w_r$	$5\times m\times w_r$
LTOMS	$3\times m\times w_r$	$4\times m\times w_r$	$6\times m\times w_r$
ILTOMS	$4\times m\times w_r$	$4\times m\times w_r$	$7\times m\times w_r$
LCFMMS	$4\times m\times w_r$	$4\times m\times w_r$	$6\times m\times w_r$
ALTMS	$4\times m\times w_r$	$4\times m\times w_r$	$6\times m\times w_r$

4　结束语

为了改善LDPC码的最小和译码算法性能有限及传统译码算法收敛速度较慢等问题，本文在阈值最小和译码算法基础上，采用收敛速度更快的分层调度方式，提出一种ALTMS译码算法。该算法通过动态调整修正因子，使其随迭代次数实时改变，并使用分层调度以加快收敛。仿真结果表明，所提出的ALTMS译码算法实现了一定的性能增益，相比固定修正因子的MS译码算法具有更优越的纠错性能。

参考文献：

[1] 袁建国，蒯家松，张帅康．基于Stanley序列的QC-LDPC码新颖构造方法[J]．重庆邮电大学学报(自然科学版)，2023，35(3)：427-433.
Yuan Jianguo，Kuai Jiasong，Zhang Shuaikang. Novel construction method on QC-LDPC codes based onthe Stanley sequence[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition)，2023，35(3)：427-433.

[2] 张晓芳，黎　勇．GPU的QC-LDPC码译码器设计与实现[J]．重庆邮电大学学报(自然科学版)，2021，33(4)：577-583.
Zhang Xiaofang，Li Yong. Design and implementation of QC-LDPC decoder based on GPU[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition)，2021，33(4)：577-583.

[3] Jia X，Zhou Y，Qin Z，et al. Reduced complexity dynamic schedule for layered decoding of LDPC codes[J]. IEEE Transactions on Vehicular Technology，2025，74(8)：13156-13160.

[4] 林宇晨，谢肇鹏，陈平平．物理层网络编码的分层调制LDPC系统性能研究[J]．重庆邮电大学学报(自然科学版)，2025，37(3)：315-323.
Lin Yuchen，Xie Zhaopeng，Chen Pingping. Performance study of hierarchical modulation LDPC systems for physical layer network coding[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition)，2025，37(3)：315-323.

[5] Zhang Q，Xu L，Huang J，et al. Distributed satellite information networks：architecture，enabling technologies，and trends[J]. Science China Information Sciences，2025，68(9)：190301.

[6] 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.

[7] Chen J，Dholakia A，Eleftheriou E，et al. Reduced-complexity decoding of LDPC codes [J]. IEEE Transactions on Communications，2005，53(8)：1288-1299.

[8] Chen J，Fossorier M P C. Density evolution for two improved BP-Based decoding algorithms of LDPC codes[J]. IEEE Communications Letters，2002，6(5)：208-210.

[9] Hatami H，Mitchell D G M，Costello D J，et al. A threshold-based Min-sum algorithm to lower the error floors of quantized LDPC decoders[J]. IEEE Transactions on Communications，2020，68(4)：2005-2015.

[10] 袁建国，张育宁，王姿现，等．一种LDPC码的优化分层阈值偏移最小和译码算法[J]．电讯技术，2024，64(9)：1415-1419.
Yuan Jianguo，Zhang Yuning，Wang Zixian，et al. An improved layered threshold offset Min-sum decoding algorithm for LDPC codes [J]. Telecommunication Engineering，2024，64(9)：1415-1419.

[11] 宁晓燕，孙晶晶，孙志国，等．LDPC码的分层类拟合修正最小和译码算法[J]．哈尔滨工业大学学报，2022，54(11)：88-94.
Ning Xiaoyan，Sun Jingjing，Sun Zhiguo，et al. Layered class fitting modified minimum sum decoding algorithm for LDPC codes[J]. Journal of Harbin Institute of Technology，2022，54(11)：88-94.

作者简介：

袁建国(1968—)，男，重庆市人，教授，博士，硕士生导师，主要研究方向为通信技术与信道编码技术。