DOI:10.11918/202502060

LDPC 码的双决策残差值置信度传播译码算法

王一珂,孙志国,宁晓燕

(哈尔滨工程大学 信息与通信工程学院,哈尔滨 150001)

摘 要:针对低密度奇偶校验(low-density parity-check,LDPC)码的多种动态调度残差值置信度传播(residual belief propagation,RBP)译码算法存在贪婪性和静默变量节点的问题,引入校验方程和概率残差值共同决策的方法,提出双决策残差值置信度传播译码算法(double decision RBP,DD-RBP)。首先根据计算的概率残差值选择需要更新的变量节点,以减少静默变量节点的个数。然后根据校验方程结果更新相关校验节点对应边的残差值,进一步降低贪婪性。最后结合更新后的残差值,在需要更新的变量节点所连接边的集合中,局部或全局选择残差值最大的边并更新,重复上述过程直至达到设置的最大次数。理论分析与仿真结果表明:对于 IEEE802.16e 标准和 5G NR 标准下的低密度奇偶校验码,所提出的双决策残差值置信度传播译码算法通过增加复杂度,在加性高斯白噪声信道和瑞利衰落信道下的译码性能优于其他译码算法。

关键词:信道编码;低密度奇偶校验码;译码算法;残差值;动态调度

中图分类号:TN911.22 **文献标志码**:A **文章编号**:0367-6234(2025)06-0084-08

Double decision residual confidence propagation decoding algorithm for LDPC code

WANG Yike, SUN Zhiguo, NING Xiaoyan

(College of Information and Communication, Harbin Engineering University, Harbin 150001, China)

Abstract: In order to solve the problems of greediness and silent variable nodes in multiple dynamic scheduling residual belief propagation (RBP) decoding algorithms for low-density parity-check (LDPC) codes, a double decision residual belief propagation decoding algorithm (DD-RBP) is proposed by introducing the method of joint decision based on the check equation and probability residual value. Firstly, the variable nodes that need to be updated are selected according to the calculated probability residual value, which can reduce the number of silent variable nodes. Then, according to the results of the check equation, the residual values of the corresponding edges of the relevant check nodes are updated to further reduce the greediness. Finally, combined with the updated residual value, in the set of edges connected by the variable nodes that need to be updated, the edge with the largest residual value is selected locally or globally and updated, and the above process is repeated until the maximum number of settings is reached. Theoretical and simulation analyses demonstrate that, for low-density parity-check codes under IEEE802.16e standard and 5G NR standard, the proposed double decision residual belief propagation decoding algorithm performs better than other algorithms in decoding under additive Gaussian white noise channels and Rayleigh fading channels with increased complexity.

Keywords: channel coding; low-density parity-check(LDPC); decoding algorithm; residual; dynamic scheduling

低密度奇偶校验(low-density parity-check,LDPC)码最初由 Gallager[1] 于 1962 年提出,后续经 Mackay 和 Neal[2] 研究证明 LDPC 码具有接近香农限的译码性能。由于具有优良的编码效率和性能,LDPC 已广泛应用于航空航天、数字电视和 5G 技术等通信领域。

LDPC 码在置信度传播(belief propagation,BP)译码算法下具有良好的性能,其中调度策略对译码算法的性能有十分重要的影响[3]。BP 译码算法采用的是传统泛洪(flooding)调度方式,其中变量节点到校验节点(variable-to-check,V2C)和校验节点到变量节点(check-to-variable,C2V)的消息更新与传递同时进行,从而限制了 BP 译码算法的收敛速度和译码性能。为了提高消息更新与传递过程中信息的可靠性,洗牌置信度[4](shuffled BP,SBP)传播和分层置信度[5](layered BP,LBP)传播译码算法通过

收稿日期:2025-02-28;**录用日期**:2025-04-28;**网络首发日期**:2025-05-26
网络首发地址:https://link.cnki.net/urlid/23.1235.T.20250523.1417.002
基金项目:国家自然科学基金(62001138)
作者简介:王一珂(2000—),男,硕士研究生;孙志国(1977—),男,教授,博士生导师
通信作者:宁晓燕,ningxiaoyan@hrbeu.edu.cn

改变变量节点（variable nodes，VN）和校验节点（check nodes，CN）的调度顺序提高信息收敛速度，相对于传统泛洪调度收敛速度提升约 2 倍。

虽然分层调度策略可以大大提高信息收敛速度，但不同节点所需信息量是不同的，因此固定的信息更新顺序并不是最佳的选择。为获取更快的收敛速度和更好的译码性能，文献[6]采用了知情动态调度（informed dynamic scheduling，IDS）策略。根据此策略，对基于 C2V 消息残差序列的残差置信度（residual belief propagation，RBP）传播译码算法进行改进。但 RBP 算法每次仅更新 C2V 中残差值最大的边，因此少量的边占用了大量的译码资源，呈现出贪婪性。贪婪性的存在使得 RBP 算法在前几次的迭代信息收敛速度很快，但在较多迭代次数的情况下性能低于静态调度策略。文献[7－9]均通过更新残差值降低贪婪性，文献[7]采用衰减机制来降低 C2V 中的较大的残差值，以避免较少的边占用大量的译码资源。在此基础上，文献[8]采用两个列表筛选出已经稳定的节点，避免引入不可靠的信息，在一定程度上保证了优先传输新更新的可靠信息。文献[9]则直接通过一个禁忌列表降低贪婪性。

译码的最终目的是获得正确的决策，与变量节点密切相关，贪婪性的存在还会使得一部分变量节点进入静默状态，使其不能参与译码过程中的信息传递。为了解决这一问题，文献[10－17]均是通过选择变量节点确定更新过程。文献[10]提出无静默节点残差置信度（silent-variable-node-free RBP，SVNF-RBP）传播算法，通过遍历每一个变量节点，选出每个变量节点连接的 C2V 最大值进行信息更新与传递，因此每个变量节点都有相同的机会参与译码过程中的信息传递，但固定的选择顺序会降低信息的收敛速度。为了获取更快的信息收敛速度，文献[11]在上述算法的基础上提出动态无静默变量节点残差置信度（dynamic silent-variable-node-free scheduling RBP，DSVNF-RBP）传播算法，采用动态的变量节点选择，优先更新不满足条件的变量节点，从而获得更快的信息收敛速度。相较于动态选择，文献[12,17]都提出更加可靠的决策方法选择变量节点，通过概率残差值进行选择。文献[18－19]则通过每层残差值进行译码，在损失译码性能的前提下大大降低译码复杂度。

上述算法根据校验节点或变量节点计算残差值，没有将两者结合充分利用节点间的信息。为了充分利用译码过程中的残差值信息，本文提出了双决策残差值置信度传播译码算法（double decision RBP，DD-RBP）。首先根据计算的概率残差值选择所需更新的变量节点，然后根据校验方程判决结果，对不满足校验结果的校验节点所对应的残差值进行更新。通过变量节点和校验节点共同确定需要更新的边，在降低静默变量节点数目的同时又能保证较低的贪婪性。

1　RBP 译码算法

设大小为 $M\times N$ 的 LDPC 校验矩阵 $\boldsymbol{H}=[h_{mn}]$，其中 h_{mn} 为 1 时表示变量节点 v_n 和校验节点 c_m 相连接。可以由 Tanner 图表示，见图 1，其中连线表示校验矩阵在第 m 行 n 列取值为 1。

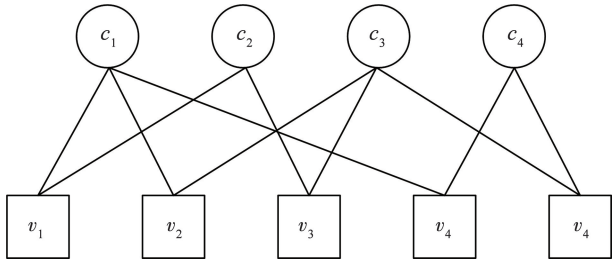


图 1　LDPC 码 Tanner 图

Fig. 1　Tanner graph representation of LDPC codes

假设信息序列 $\boldsymbol{u}=(u_1,u_2,\cdots,u_M)$ 经过编码后得到 $\boldsymbol{c}=(c_1,c_2,\cdots,c_N)$，$c_n=\{0,1\}$，经二进制相移键控（binary phase shift keying，BPSK）映射为 $\boldsymbol{x}=(x_1,x_2,\cdots,x_N)$，其中 $x_n=1-2c_n$，$(n=1,2,\cdots,N)$。采用基于软判决的 RBP 译码算法，记经过加性高斯白噪声信道后接收端序列为 $\boldsymbol{y}=(y_1,y_2,\cdots,y_N)$，$y_n=x_n+k_n$，其中 k_n 是均值为 0，方差为 $N_0/2$ 的加性高斯白噪声，N_0 为噪声功率。译码结果为 $\hat{\boldsymbol{u}}=(\hat{u}_1,\hat{u}_2,\cdots,\hat{u}_N)$。

为便于描述，对下述公式涉及符号说明如下。C2V 为 c_m 到 v_n 信息更新与传递过程，数值记为 $L^{\mathrm{C}}_{m\to n}$；V2C 为 v_n 到 c_m 信息更新与传递过程，数值记为 $L^{\mathrm{V}}_{n\to m}$；$L^{\mathrm{V}}_{n\to m}$ 和 $L^{\mathrm{V}}_{n\to m}$ 的初始化数值分别为 0 和 $4y_n/N_0$。$N(m)$ 表示与校验节点相连的所有变量节点的集合，即 $N(m)=\{n:h_{mn}=1\}$ 是第 m 行中的所有非 0 元素；$M(n)$ 表示与变量节点相连的所有校验节点的集合，即 $M(n)=\{m:h_{mn}=1\}$ 是第 n 列中的所有非 0 元素。$N(m)\backslash n$ 和 $M(n)\backslash m$ 表示除去当前元素的所有非 0 元素集合。

对于 RBP 算法，从 c_m 到 v_n 消息更新计算式为

$$L^{\mathrm{C}}_{m\to n}=2\tanh^{-1}\left[\prod_{n'\in N(m)\backslash n}\tanh\frac{L^{\mathrm{V}}_{n'\to m}}{2}\right] \tag{1}$$

从 v_n 到 c_m 消息更新计算式为

$$L^{\mathrm{V}}_{n\to m}=\frac{4y_n}{N_0}+\sum_{m'\in M(n)\backslash m}L^{\mathrm{C}}_{m'\to n} \tag{2}$$

v_n 判决信息 LLR 为

$$L_n = \frac{4y_n}{N_0} + \sum_{m\in M(n)} L^{C}_{m\to n} \tag{3}$$

RBP 算法[6]信息更新过程步骤如下。

1)计算残差值

$$R^{C}_{m\to n} = |\tilde{L}^{C}_{m\to n} - L^{C}_{m\to n}| \tag{4}$$

式中:$\tilde{L}^{C}_{m\to n}$是由式(1)预先计算这次迭代的 C2V 信息,$L^{C}_{m\to n}$是由式(1)计算上一次迭代的 C2V 信息。

2)找到最大残差值位置

$$(i,j) = \arg\max_{(m,n)} R^{C}_{m\to n} \tag{5}$$

3)更新 C2V 边

$$L^{C}_{i\to j} = \tilde{L}^{C}_{i\to j} \tag{6}$$

4)根据式(2)计算 V2C 信息 $L^{V}_{j\to m'}, m'\in M(j)\backslash m$。

2 双决策残差值译码算法

在译码迭代过程中,不同的校验节点所连接的边对译码过程带来的信息量不同,优先更新最不可靠的边能够获得更好的性能,但如何选择最优的更新顺序是一个 NP-hard 难题[7]。对于传统 RBP 译码算法,尽管一些边可能仅需要少量的译码资源就能给出正确的信息,但贪婪性的存在会导致少量的边占用大量的译码资源。文献[6-9]给出了降低贪婪性的部分思路,这些方法均是在降低贪婪性而不是消除贪婪性。这是因为贪婪性的存在可以使得最不可靠的边优先传输,加快信息收敛速度,但同时也剥夺了其他边传输信息的机会。因此需要一个均衡的方法来削弱贪婪性。此外,C2V 边的选择仅通过残差值来确定是不可靠的,事实上残差值最大的边并不能准确地表示这条边是最不可靠的[13]。译码的最终目的是获得正确的决策,可根据校验方程筛选出不满足译码结果的校验节点,获得可靠的结果。因此,本文提出了双决策残差值置信度传播(double decision RBP,DD-RBP)译码算法,分别通过校验方程决策 CN 和根据概率残差值决策 VN。图 2 为算法基本原理框图。

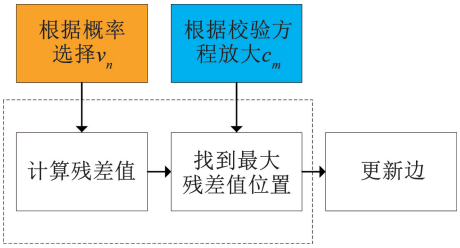


图 2 DD-RBP 算法框图

Fig. 2 Algorithm block diagram of DD-RBP

2.1 校验方程决策 CN

为了获得最不可靠的边,本文首先提出对于不满足奇偶校验方程(parity check,PC)的校验节点残差值放大处理,使残差值较小但错误的节点获得更新的机会。初始化、C2V 和 V2C 信息更新与传递过程与 RBP 相同,主要差别在于残差值的计算。首先需要根据式(3)计算出判决信息$\tilde{L}_n$,判决最终结果为$\tilde{\boldsymbol{L}}$,定义$\boldsymbol{\gamma}=(\gamma_1,\gamma_2,\cdots,\gamma_m)$表示校验方程结果。可得

$$\boldsymbol{\gamma} = \mathrm{mod}\{\boldsymbol{H}\times[(1-\mathrm{sgn}(\tilde{\boldsymbol{L}}^{\mathrm{T}}))/2],2\} \tag{7}$$

式中:$\tilde{\boldsymbol{L}}$为 $\tilde{L}_n$ 组成的向量,$\boldsymbol{H}$ 为校验矩阵,$\boldsymbol{\gamma}$ 中不为 0 的节点表示不满足校验方程。可通过式(8)计算残差值

$$R^{C}_{m\to n} = \begin{cases} \alpha\times|\tilde{L}^{C}_{m\to n} - L^{C}_{m\to n}|, \gamma_m = 1 \\ |\tilde{L}^{C}_{m\to n} - L^{C}_{m\to n}|, \gamma_m = 0 \end{cases} \tag{8}$$

式中 $\alpha>1$。

将不满足校验方程的校验节点所对应的残差值放大,增大该节点在残差值计算过程中被优先选择的概率。但过大的参数 α 使不满足校验方程的校验节点数值过大,又将面临贪婪性的问题,因此合适的数值可在更好的选择所需更新的校验节点的同时又能够避免贪婪性问题。图 3 给出在参数 α 取不同值时,IEEE802.16e 标准,码长为 576 下的 LDPC 码在信噪比为 2.2 dB 时误帧率(frame error rate,FER)性能。

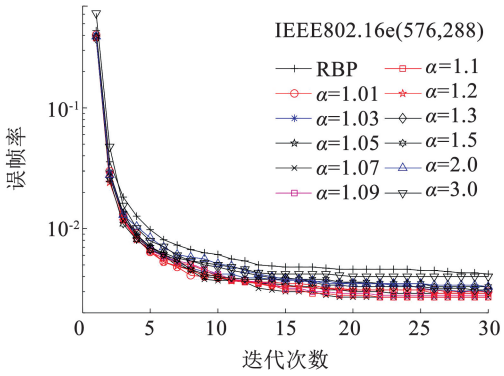


图 3 不同参数 α 下 FER

Fig. 3 FER under different parameters α

由图 3 可以看到,误帧率先随着参数 α 的增大不断降低,但当参数 $\alpha>1.1$ 时,误帧率随着 α 的增大反而升高。在前 3 次迭代过程中,除 $\alpha=3.0$ 时性能低于 RBP,其他参数收敛速度均优于 RBP。因此,选择合适的参数 α 可以优先更新不满足校验结果的校验节点,在一定程度上降低贪婪性现象,提升译码性能。

2.2 概率残差决策 VN

优先更新不稳定的变量节点可以提高判决信息 LLR 的准确性,文献[10-16]均是通过该方法提高译码性能。但是通过符号变换[16]、三重准则[15]和

可靠性指标[14]都忽略了具有较小 LLR 值但不稳定的变量节点。为了更好地选取变量节点，本文采用概率残差值决策的方法。

当接收端第 n 比特译码结果 $\hat{u}_n$ 与发送端信息序列 u_n 不同时表示译码错误，其发生的概率记为 P_{e_n}，

$$P_{e_n} = P(\hat{u}_n \neq u_n) \tag{9}$$

式中译码信息由判决信息 LLR 得到，$\hat{u}_n = \mathrm{sgn}(L_n)$。

设 $\boldsymbol{\rho}_N = (1,2,\cdots,N)$，接收端译码错误概率

$$P(\hat{\boldsymbol{u}} \neq \boldsymbol{u}) = 1 - \prod_{n\in\boldsymbol{\rho}_N}(1 - P_{e_n}) \tag{10}$$

由式(10)可知，如果要降低译码错误的概率需要降低 P_{e_n}，在更新时选择 P_{e_n} 变化最大的变量节点 $v_{n_{\max}}$，经过迭代降低 P_{e_n} 的值。

$$n_{\max} = \arg\max_{n\in\boldsymbol{\rho}_N} |\tilde{P}_{e_n} - P_{e_n}| \tag{11}$$

式中 $\tilde{P}_{e_n}$ 表示下一次迭代概率。

根据现有条件无法计算式(11)的值，本文定义新的变量 D_n 来替代 $|\tilde{P}_{e_n} - P_{e_n}|$，

$$D_n = |\tilde{P}(\hat{u}_n = 0) - P(\hat{u}_n = 0)| \tag{12}$$

首先需要求得 $P(\hat{u}_n = 0)$ 的值，先验信息 L_n 可以表示为[20]

$$L_n = \ln\frac{P(\hat{u}_n = 0)}{P(\hat{u}_n = 1)} \tag{13}$$

根据 $P(\hat{u}_n = 1) = 1 - P(\hat{u}_n = 0)$，则可求得 $P(\hat{u}_n = 0)$，即

$$P(\hat{u}_n = 0) = \frac{\mathrm{e}^{L_n}}{1 + \mathrm{e}^{L_n}} \tag{14}$$

式(11)可更新为

$$n_{\max} = \arg\max_{n\in\boldsymbol{\rho}_N}\left|\frac{\mathrm{e}^{\tilde{L}_n}}{1+\mathrm{e}^{\tilde{L}_n}} - \frac{\mathrm{e}^{L_n}}{1+\mathrm{e}^{L_n}}\right| \tag{15}$$

式中 $\tilde{L}_n$ 为预先计算的判决信息。

D_n 数值越大表示译码结果为 0 的变化概率越大，即表明该数值变化幅度越大，这与文献[14－16]的思想一致，且式(14)表明通过概率转换可以使数值较小但不稳定的变量节点获得更新的机会。

2.3　DD-RBP 算法

所提出的 DD-RBP 算法将根据校验方程选择校验节点与根据概率残差值选择变量节点两个准则共同确定所需更新的节点，表 1 给出 DD-RBP 算法的伪代码，其中 $\beta = 0.1$[12]。

表 1　DD-RBP 算法

Tab. 1　DD-RBP decoding algorithm

算法：DD-RBP 算法

1：初始化 $L^{\mathrm{C}}_{m\to n} = 0$，$L^{\mathrm{V}}_{n\to m} = L_n = 4y_n/N_0$

2：根据式(2)计算 $\tilde{L}^{\mathrm{C}}_{m\to n}$ 并计算 $R^{\mathrm{C}}_{m\to n}$

3：根据式(3)计算 $\tilde{L}_n$ 并计算 γ_m 和 D_n

4：计算 $n_{\max} = \arg\max\{D_n \mid n\in\boldsymbol{\rho}_N\}$

5：**if**　$D_{n_{\max}} < \beta$

6：　　计算 $m^* = \{m \mid \gamma_m = 1\}$，$n^* \in N(m^*)$，更新 $R^{\mathrm{C}}_{m^*\to n^*} = \alpha \times R^{\mathrm{C}}_{m^*\to n^*}$

7：　　$(m_{\max}, n_{\max}) = \arg\max_{(i,j)}\{R^{\mathrm{C}}_{i\to j} \mid h_{ij} = 1\}$

8：**else**

9：　　计算 $m^* = \{m \mid \gamma_m = 1, m\in M(n_{\max})\}$，更新 $R^{\mathrm{C}}_{m^*\to n_{\max}} = \alpha \times R^{\mathrm{C}}_{m^*\to n_{\max}}$

10：　　$m_{\max} = \arg\max_{m^*}\{R^{\mathrm{C}}_{m^*\to n_{\max}}\}$

11：**end if**

12：计算 $L^{\mathrm{C}}_{m_{\max}\to n_{\max}} = \tilde{L}^{\mathrm{C}}_{m_{\max}\to n_{\max}}$，更新 $L_{n_{\max}}$、$D_{n_{\max}}$ 令 $R^{\mathrm{C}}_{m_{\max}\to n_{\max}} = 0$

13：**for**　$i\in M(n_{\max})\backslash m_{\max}$　**do**

14：　　根据式(2)计算 $L^{\mathrm{V}}_{n_{\max}\to i}$

15：　　**for**　$j\in N(i)\backslash n_{\max}$　**do**

16：　　计算 $\tilde{L}^{\mathrm{C}}_{i\to j}$、$R^{\mathrm{C}}_{i\to j}$ 和 $\tilde{L}_j$，并更新 D_j

17：　　**end for**

18：　　更新 γ_m

19：**end for**

20：如果不满足判决条件，跳转至第 5 行

图 4 为 DD-RBP 算法数据迭代过程中节点间信息传递的简单示例,其中有 5 个校验节点和 6 个变量节点。首先图 4(a)为译码信息传递过程,假设 v_3 为首次根据概率残差值计算出的 VN,找到与 v_3 相连接 CN(c_2,c_3,c_4),根据式(8)更新不满足校验方程的 CN 残差值。然后在所有连接边 $c_2\rightarrow v_3$、$c_3\rightarrow v_3$ 和 $c_4\rightarrow v_3$ 找到最大残差值 $r(c_3\rightarrow v_3)$,$c_3\rightarrow v_3$ 根据式(4)进行 V2C 更新。最后对剩余的边 $v_3\rightarrow c_2$ 和 $v_3\rightarrow c_4$ 根据式(2)进行 V2C 更新,上述过程完成了一次 CN 和 VN 之间的信息传递。图 4(b)为 DD-RBP算法中残差值计算与更新,对已更新过的 CN(c_2 和 c_4),找到与其相连接但不为前一次选择的 VN,分别为 v_1、v_2、v_4 和 v_6,根据式(15)重新计算概率残差值,假设概率残差值最大是 v_2,则开始下一轮的迭代,如图 4(c)所示,过程与图 4(a)一致。

通过 T-EXIT 图对所提出的 DD-RBP 译码算法的平均互信息量(average mutual information,AMI)进行分析[21]。采用 IEEE802.16e 标准和 5G NR 标准,通过加性高斯白噪声信道,信噪比为 2 dB,最大迭代次数为 30 次时的 AMI 大小如图 5 所示,其中,$I_{A,\mathrm{VN}}$和 $I_{E,\mathrm{CN}}$分别为检验节点传递给变量节点的输入和输出似然信息与码元序列之间的互信息量,$I_{A,\mathrm{CN}}$ 和 $I_{E,\mathrm{VN}}$分别为变量节点传递给校验节点的输入和输出似然信息与码元序列之间的互信息量。

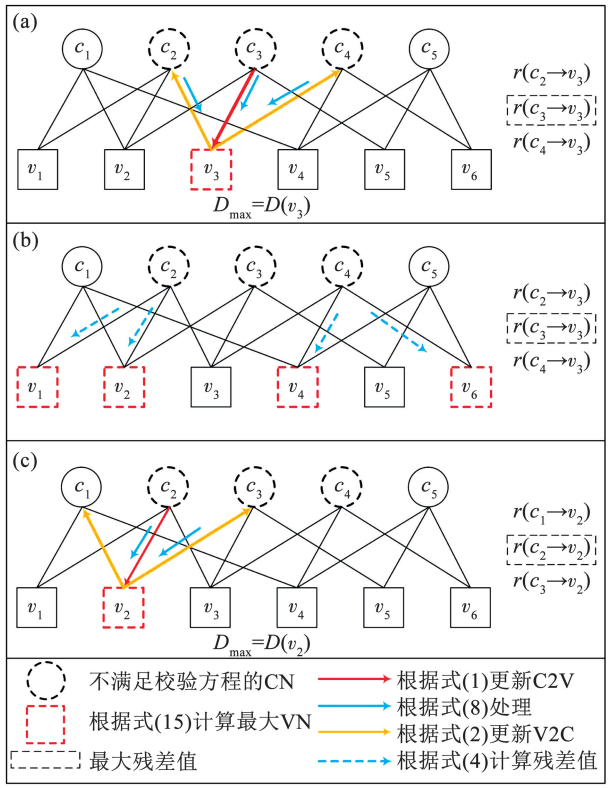
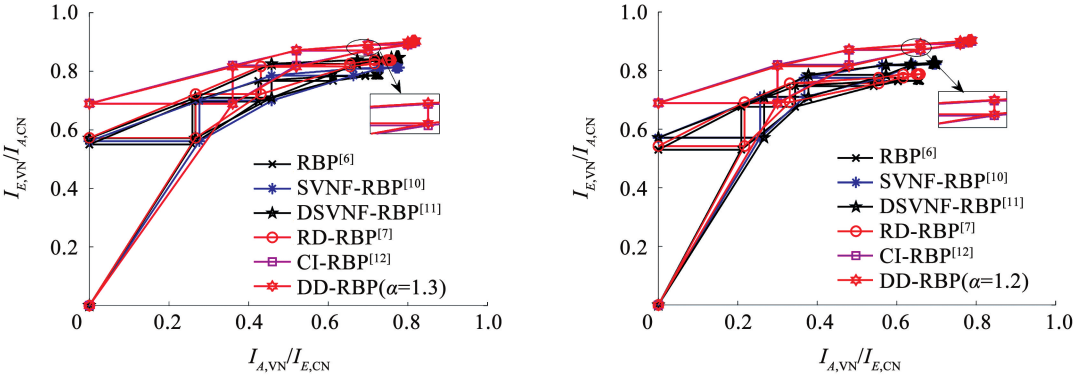


图 4 CN 与 VN 之间信息传递过程示例

Fig. 4 Example of the information transmission process between CN and VN



(a) IEEE802.16e标准(码长1 056) (b) 5G NR标准(码长512)

图 5 不同标准在 AWGN 信道下 T-EXIT 图

Fig. 5 T-EXIT diagram under AWGN channel with different standards

由图 5 可知,在采用 IEEE802.16e 标准和 5G NR 标准下,本文所提出的 DD-RBP 译码算法的 AMI 值明显高于其他译码算法,略优于 CI-RBP 译码算法。证明在译码过程中 DD-RBP 译码算法传递信息量大,收敛速度比其他译码算法快,验证了本文所提出的 DD-RBP 译码算法的有效性。

3 算法性能分析

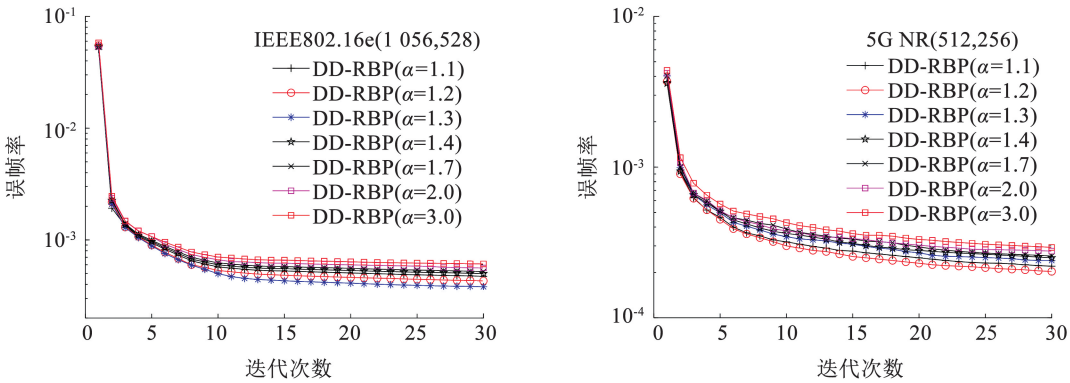
本次仿真分析所使用的是 IEEE802.16e 和 5G NR 两种标准下的校验矩阵,分别在 AWGN 信道和 Rayleigh 衰落信道下,采用 BPSK 调制解调方式,最大迭代次数为 30 次,码率均为 1/2,其中 IEEE802.16e 标准采用 1 056 码长,5G NR 采用 576 码长。通过仿真对本文提出的 DD-RBP 算法进行评估,与 RBP[6] 算法、SVNF-RBP[10] 算法、DSVNF-RBP[11] 算法、RD-RBP[7] 算法和 CI-RBP[12] 算法进行比较。为方便理解,后续用 BER 表示误比特率,FER 表示误帧率。

3.1 译码性能分析

图 6(a)和(b)分别为 IEEE802.16e 标准和 5G NR 标准在 AWGN 信道,信噪比分别为 2.0 和 2.4 dB 下的 FER 收敛速度。对于 IEEE802.16e 标准,当$\alpha=1.1$

时前两次收敛速度较快，但随着迭代次数的增加，$\alpha=1.3$ 表现出更好的 FER 收敛性能。对于 5G NR 标准，当 $\alpha=1.2$ 时表现出更好的 FER 收敛性能。且两种标准均随着 α 的不断增大，FER 收敛性能逐渐变差，这是由于过大的数值反而会加剧贪婪性，为了避免一张图中曲线过多，后续仿真均选用 $\alpha=1.2$ 和 $\alpha=1.3$ 两个参数。

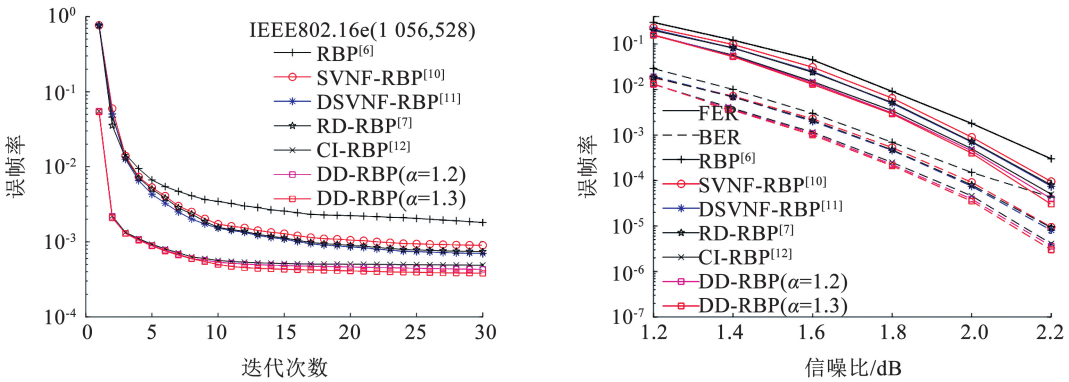
图 7(a) 为 IEEE802.16e 标准在 AWGN 信道，信噪比为 2.0 dB 条件下的 FER。在 FER$\approx10^{-3}$时，RBP 译码算法性能最差，本文提出的 DD-RBP 算法 FER 的收敛速度与 SVNF-RBP、DSVNF-RBP 和 RD-RBP算法相比快约 10 次迭代。在迭代次数较少的情况下，CI-RBP 与 DD-RBP 译码算法收敛速度明显优于其他译码算法。当迭代次数大于 10 次后，DD-RBP 算法性能优于 CI-RBP 算法。图 7(b) 为 IEEE802.16e 标准，最大迭代次数为 30 次，在 AWGN 信道下的 FER 与 BER。在低信噪比条件下，本文提出的 DD-RBP 与 CI-RBP 译码算法性能基本一致，且均优于其他改进译码算法，在中高信噪比条件下，DD-RBP 译码算法优于 CI-RBP 译码算法。



(a) IEEE802.16e标准下的FER　　(b) 5G NR标准下的FER

图 6　不同标准在 AWGN 信道下 FER 信息收敛速度

Fig. 6　Convergence speed of FER information under AWGN channel with different standards



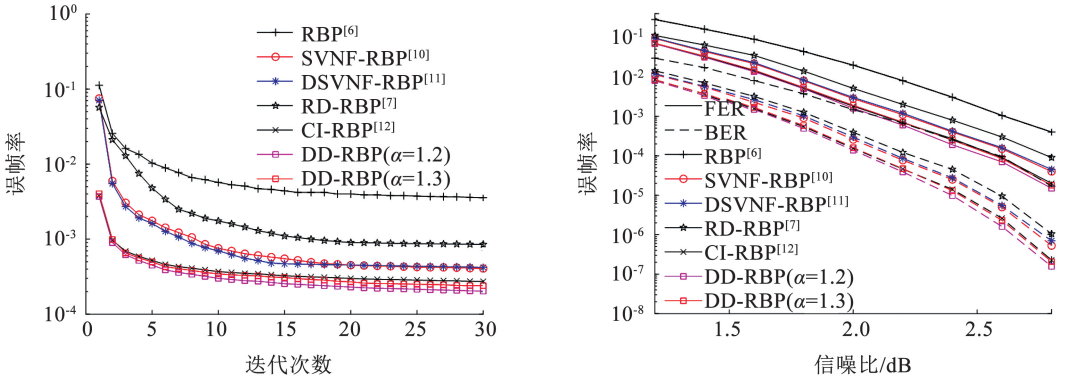
(a) FER的收敛速度　　(b) 不同算法的FER与BER

图 7　不同算法在 AWGN 信道下 FER 与 BER 性能(IEEE802.16e 标准，码长 1 056)

Fig. 7　FER and BER performance of different algorithms in AWGN channels (IEEE802.16 standard, code length 1 056)

图 8(a)为 5G NR 标准在 AWGN 信道，信噪比为 2.4 dB 条件下的 FER，各算法译码性能与 IEEE802.16e 标准下的趋势基本保持一致，但几种算法之间的差距明显增大。在 FER$\approx10^{-3}$时，本文提出的 DD-RBP 算法 FER 收敛速度比 SVNF-RBP 与 DSVNF-RBP 快约 5 次迭代，比 RD-RBP 算法快约 15 次迭代。当参数 $\alpha=1.2$ 时，DD-RBP 译码算法 FER 收敛速度优于 CI-RBP 译码算法，但当参数 $\alpha=1.3$ 时 FER 收敛速度与 CI-RBP 基本一致。图 8(b)为 5G NR 标准，最大迭代次数为 30 次，在 AWGN 信道下的 FER 与 BER。在中高信噪比条件下，本文所提出的 DD-RBP 译码算法在参数 $\alpha=1.2$ 时 FER 与 BER 性能优于其他改进算法，在参数 $\alpha=1.3$ 时，性能与 CI-RBP 译码算法基本保持一致。

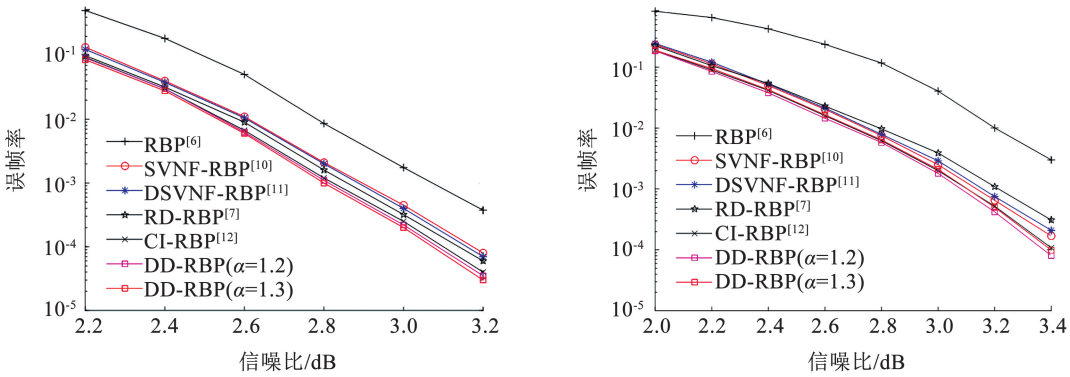
图 9(a)和(b)分别为 IEEE802.16e 标准和 5G NR标准，在瑞利衰落信道下的 FER，瑞利衰落随机数的功率与传输信号的功率比为 1∶1，其中 IEEE802.16e 标准码长为 1 056，5G NR 标准码长为 512。



(a) FER的收敛速度 (b) 不同算法的FER与BER

图 8 不同算法在 AWGN 信道下 FER 与 BER 性能(5G NR 标准,码长 512)

Fig. 8 FER and BER performance of different algorithms in AWGN channels (5G NR standard, code length 512)



(a) IEEE802.16e标准 (b) 5G NR标准

图 9 不同算法在不同标准下通过 Rayleigh 衰落信道的 FER 性能

Fig. 9 FER performance of different algorithms over Rayleigh fading channels under different standards

由图 9 可知,相对于 AWGN 信道,除 RBP 算法外不同改进算法之间的差距明显缩小。但整体来看,本文所提出的 DD-RBP 译码算法在两种不同标准下译码性能优于 RBP、SVNF-RBP、DSVNF-RBP 和 RD-RBP 译码算法,在参数 α 选择合适的情况和高信噪比条件下性能均优于 CI-RBP 译码算法。

3.2 复杂度分析

表 2 给出了不同算法所需的 C2V、V2C、残差值计算与更新、比较和 D_n 计算模块。其中 $M \times N$ 为校验矩阵的大小,E 为校验矩阵中元素 1 的个数,$\bar{d}_v$ 为变量节点边平均数,$\bar{d}_c$ 为校验节点边平均数,δ 为 $D_j \leqslant \beta$ 的概率。

表 2 计算复杂度

Tab. 2 Computational complexity

算法	C2V	V2C	残差值计算与更新	比较	D_n 计算
RBP	E	$E(\bar{d}_v-1)$	$E(\bar{d}_v-1)(\bar{d}_c-1)$	$E(E-1)$	0
SVNF-RBP	$\lfloor E/N \rfloor N$	$\lfloor E/N \rfloor N(\bar{d}_v-1)$	$\lfloor E/N \rfloor N(\bar{d}_v-1)(\bar{d}_c-1)$	$\lfloor E/N \rfloor N[\bar{d}_v(\bar{d}_c-1)-1]$	0
DSVNF-RBP	$\lfloor E/N \rfloor N$	$\lfloor E/N \rfloor N(\bar{d}_v-1)$	$\leqslant \lfloor E/N \rfloor N(\bar{d}_v-1)(\bar{d}_c-1)$	$\leqslant \lfloor E/N \rfloor N[\bar{d}_v(\bar{d}_c-1)-1]$	0
RD-RBP	E	$E(\bar{d}_v-1)$	$E(\bar{d}_v-1)(\bar{d}_c-1)$	$E(E-1)$	0
CI-RBP	E	$E(\bar{d}_v-1)$	$E(\bar{d}_v-1)(\bar{d}_c-1)$	$E[N+(1-\delta)(\bar{d}_v-1)+\delta(E-1)]$	$N+E(\bar{d}_v-1)(\bar{d}_c-1)$
DD-RBP	E	$E(\bar{d}_v-1)$	$E(\bar{d}_v-1)(\bar{d}_c-1)$	$E[N+(1-\delta)(\bar{d}_v-1)+\delta(E-1)]$	$N+E(\bar{d}_v-1)(\bar{d}_c-1)$

由表 2 可知,DD-RBP 译码算法 C2V、V2C 和残差值计算与更新过程计算量与除 SVNF-RBP 和 DSVNF-RBP 外的其他算法基本保持一致,分别需要 E 次式(1)、$E(\bar{d}_v-1)$ 次式(2)和 $E(\bar{d}_v-1)(\bar{d}_c-1)$ 次式(4)计算。DD-RBP 译码算法在比较模块与其他算法有所不同,首先需要搜索 N 次寻找最大变量节点概率残差值,然后根据是否满足概率残差值对校验节点残差值进行比较。当满足校验残差值时进行 $\delta(E-1)$ 次比较,相反,当不满足校验残差值时进行 $(1-\delta)(\bar{d}_v-1)$ 次比较。DD-RBP 译码算法还需

要计算 D_n，即变量节点的概率残差值，需要在初始化阶段进行 N 次计算和在后续更新中进行 $E(\bar{d}_v-1)$，$(\bar{d}_c-1)$次计算，计算公式为式(15)。此外，本文所提出的 DD-RBP 译码算法相对于 CI-RBP 译码算法还需要对校验方程的结果进一步处理，而在实际使用过程中，更新所需要的参数已预先通过仿真得到并存储在寄存器中，只需要根据校验方程的结果对相应的值进行更新，相对于其他模块计算量较小。综上认为，本文所提出的 DD-RBP 译码算法与 CI-RBP译码算法复杂度基本一致，与其他算法复杂度相比较高。

4　结　语

本文提出了一种新的基于残差值的动态调度译码算法，通过引入校验方程和概率残差值共同决策的方法，选取更为准确的残差值进而提升译码性能。首先根据计算概率残差值选择需要更新的变量节点，而后根据校验方程结果对相关的校验节点残差值进行更新，最后更新两节点共同连接的边和对应的残差值，不断重复上述过程直至达到最大更新次数，开始下一轮迭代过程。通过概率残差值可以使得数值较小但不稳定的变量节点获得更新的机会，减少静默变量节点的数量，根据校验方程校验结果选择合适的参数对校验节点残差值进行更新，可以更加准确地选择需要更新的校验节点并降低贪婪性。理论分析与仿真结果表明，本文所提出的 DD-RBP算法复杂度略高于其他部分算法，性能有所提升。在复杂度基本一致的情况下性能优于 CI-RBP。在 IEEE802.16e 和 5G NR 两种标准下，误帧率约为10^{-3}时，收敛速度相较于其他算法快约 5～15 次迭代，且在最大迭代次数为 30 次时，最终的误帧率与误比特率性能在加性高斯白噪声与瑞利衰落信道下优于其他译码算法。

参考文献

[1] GALLAGER R G. Low density parity check codes [J]. IRE Transactions on Information Theory, 1962, 8(1): 21. DOI: 10.1109/TIT.1962.1057683

[2] MACKAY D J C, NEAL R M. Near shannon limit performance of low density parity check codes[J]. Electronics Letters, 1996, 33: 457. DOI: 10.1049/el:19961141

[3] MAO Yongyi, BANIHASHEMI A H. Decoding low-density parity-check codes with probabilistic scheduling[J]. IEEE Communications Letters, 2001, 5(10): 414. DOI: 10.1109/4234.957379

[4] ZHANG Juntan, FOSSORIER M P C. Shuffled iterative decoding[J]. IEEE Transactions on Communications, 2005, 53(2): 209. DOI: 10.1109/TCOMM.2004.841982

[5] HOCEVAR D E. A reduced complexity decoder architecture via layered decoding of LDPC codes[C]//IEEE Workshop on Signal Processing Systems, 2004. Austin: IEEE, 2004: 107. DOI: 10.1109/SIPS.2004.1363033

[6] CASADO A I V, GRIOT M, WESEL R. D. Informed dynamic scheduling for belief-propagation decoding of LDPC codes[C]//2007 IEEE International Conference on Communications. Glasgow: IEEE, 2007: 932. DOI: 10.1109/ICC.2007.158

[7] ZHANG Huilian, CHEN Shaoping. Residual-decaying-based informed dynamic scheduling for belief-propagation decoding of LDPC codes[J]. IEEE Access, 2019, 7: 23656. DOI: 10.1109/ACCESS.2019.2899106

[8] YAN Zipin, WU Guan, LIANG Liping. List-based residual belief-propagation decoding of LDPC codes[J]. IEEE Communications Letters, 2024, 28(5): 984. DOI: 10.1109/LCOMM.2024.3381220

[9] LIU Xingcheng, FAN Chunlei, CHEN Xuechen. Dynamic scheduling decoding of LDPC codes based on tabu search[J]. IEEE Transactions on Communications, 2017, 65(11): 4612. DOI: 10.1109/TCOMM.2017.2732950

[10] 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. DOI: 10.1109/TCOMM.2013.012313.120172

[11] ASLAM C A, GUAN Y L, CAI K. Low-complexity belief-propagation decoding via dynamic silent-variable-node-free scheduling[J]. IEEE Communications Letters, 2017, 21(1): 28. DOI: 10.1109/LCOMM.2016.2615016

[12] CHANG T C Y, WANG P H, WENG J J, et al. Belief-propagation decoding of LDPC codes with variable node-centric dynamic schedules[J]. IEEE Transactions on Communications, 2021, 69(8): 5014. DOI: 10.1109/TCOMM.2021.3078776

[13] LIU Xingcheng, ZHANG Yuanbin, CUI Ru. Variable-node-based dynamic scheduling strategy for belief-propagation decoding of LDPC codes[J]. IEEE Communications Letters, 2015, 19(2): 147. DOI:10.1109/LCOMM.2014.2385096

[14] LIU Xingcheng, ZI Lie, YANG Dong, et al. Improved decoding algorithms of LDPC codes based on reliability metrics of variable nodes[J]. IEEE Access, 2019, 7: 35769. DOI: 10.1109/ACCESS.2019.2904173

[15] LIU Xingcheng, ZHOU Zhenzhu, CUI Ru. Informed decoding algorithms of LDPC codes based on dynamic selection strategy[J]. IEEE Transactions on Communications, 2016, 64(4): 1357. DOI: 10.1109/TCOMM.2016.2527642

[16] GONG Yi, LIU Xingcheng, HAN Guojun. Effective informed dynamic scheduling for belief propagation decoding of LDPC codes[J]. IEEE Transactions on Communications, 2011, 59(10): 2683. DOI: 10.1109/TCOMM.2011.072011.100438

[17] XIA Tian, WU H C, HUANG S C H. A novel fast LDPC decoder using app-based dynamic scheduling scheme[C]//2015 IEEE Global Communications Conference. San Diego: IEEE, 2015: 1. DOI: 10.1109/GLOCOM.2015.7417208

[18] WANG Bingbing, ZHU Yan, KANG Jing. Two effective scheduling schemes for layered belief propagation of 5G LDPC codes[J]. IEEE Communications Letters, 2020, 24(8): 1683. DOI: 10.1109/LCOMM.2020.2991473

[19] SUN Rong, HOU Xiaogeng, SUN Jingyuan. Reliability-based-layered belief propagation for iterative decoding of LDPC codes[C]//2018 IEEE International Symposium on Information Theory(ISIT). Vail: IEEE, 2018: 1156. DOI: 10.1109/ISIT.2018.8437926

[20] ASLAM C A, GUAN Yongliang, CAI Kui. Edge-based dynamic scheduling for belief-propagation decoding of LDPC and RS codes[J]. IEEE Transactions on Communications, 2017, 65(2): 525. DOI: 10.1109/TCOMM.2016.2637913

[21] DAI Jincheng, TAN Kailin, SI Zhongwei, et al. Learning to decode protograph LDPC codes[J]. IEEE Journal on Selected Areas in Communications, 2021, 39(7): 1983. DOI: 10.1109/JSAC.2021.3078488

（编辑　吕雪梅）