

激光与光电子学进展

引入混洗策略和改进变量节点更新的最小和译码算法

刘毓, 郝耀辉*, 黄洲, 王婵, 拜林

西安邮电大学通信与信息工程学院, 陕西 西安 710121

摘要 低密度奇偶校验码属于一种线性的分组码, 近年来得到了很大的关注。目前被广泛应用的最小和(MS)算法译码损失较多, 而且该算法属于洪泛调度, 译码收敛速度较慢。为此提出了引入混洗策略和改进变量节点更新的MS(shuffled-VNU-MS)译码算法, 该改进算法在本次迭代中更新某列校验节点信息时, 可以利用到前序列最新的变量节点信息, 且变量节点信息是通过加权因子平均后的信息。仿真结果表明: 当码长为 256, 码率为 0.5, 误码率为 10^{-5} 时, 本文改进的译码算法 shuffled-VNU-MS 相比 MS 算法, 编码增益提高了约 0.92 dB。当信噪比为 0 时, 本文改进算法译码收敛速度相比 MS 算法提高了约 52%。

关键词 光通信; 低密度奇偶校验码; 最小和算法; 变量节点更新; 混洗策略; 加权因子

中图分类号 TN911.22

文献标志码 A

DOI: 10.3788/LOP202259.1106002

Min-Sum Decoding Algorithm that Introduced Shuffled Strategy and Improved Variable Node Update

Liu Yu, Hao Yaohui*, Huang Zhou, Wang Chan, Bai Lin

School of Communication and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710121, Shaanxi, China

Abstract Low-density parity-check code is a kind of linear block code, which has received much attention in recent years. At present, the min-sum (MS) algorithm, which is widely used, has a lot of decoding loss, and it belongs to flood scheduling, so the decoding convergence speed is slow. In this paper, the MS decoding algorithm that introduced shuffled strategy and improved variable node update (shuffled-VNU-MS) is proposed. When the improved algorithm updates check node information of current column in this iteration, it can use the latest variable node information of the previous sequence, and the variable node information is averaged by the weighted factor. The simulation results show that when the code length is 256, the code rate is 0.5, and the bit error rate is 10^{-5} , shuffled-VNU-MS algorithm increases the coding gain by about 0.92 dB compared with the MS algorithm. When the signal to noise ratio is 0, compared with the MS algorithm, shuffled-VNU-MS algorithm can improve the decoding convergence speed by about 52%.

Key words optical communications; low-density parity-check code; min-sum algorithm; variable node update; shuffled strategy; weighting factor

1 引言

20 世纪 60 年代 Gallager^[1]提出一种称为低密度

奇偶校验(LDPC)码的差错控制编码方案。由于 LDPC 码具有优异的译码性能, 目前广泛使用在卫星数字视频、磁性记录介质、音频广播等领域^[2]。

收稿日期: 2021-05-24; 修回日期: 2021-07-05; 录用日期: 2021-07-14

基金项目: 科技部与烽火通信合作项目(2018YFB2201305)

通信作者: *2649276979@qq.com

置信传播(BP)译码算法^[3-5]是LDPC的传统译码算法,但是该算法采用了很多相乘运算,研究人员在BP译码算法的基础上提出了一些改进方法。其中对数似然比置信传播(LLR-BP)译码算法^[6-8]以对数似然比表示节点的概率信息,与BP算法相比较,大大缩短了运行时间。最小和(MS)算法^[9-10]在校验节点更新中用求最小值代替了LLR-BP的非线性的双曲函数的复杂运算,虽然降低了计算复杂度,但是也牺牲了译码性能。目前最小和算法的改进算法^[11-12]只是通过乘以补偿系数来提升译码性能,而且仅仅利用到上一次迭代的信息。

本文在研究最小和算法的基础上,另外分析了混洗策略在BP译码算法中的作用^[13],提出了一种基于混洗策略的加权最小和(shuffled-VNU-MS)改进算法。该改进算法利用混洗策略,在每次迭代中,利用已更新列最新的变量节点信息,实现对后序列校验节点的信息更新,且变量节点信息是经过加权因子 β 修正的信息^[14],使上次迭代的外部信息值也参与到当前迭代变量节点的更新过程中,减少外部信息的振荡。通过本文改进算法可以使更加可靠的节点信息以更快的方式进行传递,加快了译码收敛速度,提高了译码质量。

2 最小和译码算法

LDPC码是线性分组码,它由 $M \times N$ 维的校验矩阵 $\mathbf{H}=(h_{ij})_{M \times N}$ 来描述。在信息传输过程中,发送的LDPC码组为 $\mathbf{x}=(x_1, x_2, \dots, x_N)$,经过方差为 $\sigma^2=0$ 均值加性高斯白噪声(AWGN)信道,接收序列为 $\mathbf{y}=(y_1, y_2, \dots, y_N)$,对接收序列 \mathbf{y} 进行译码判决,得到译码序列 $\hat{\mathbf{x}}=(\hat{x}_1, \hat{x}_2, \dots, \hat{x}_N)$ 。

最小和译码算法关键步骤如下。

校验节点更新 $L^{(t)}(r_{ij})$ 表示为

$$L^{(t)}(r_{ij}) = \prod_{j \in \mathbf{N}(i) \setminus j} \text{sign}[L^{(t-1)}(q_{ji})] \min |L^{(t-1)}(q_{ji})|, \quad (1)$$

式中: $\mathbf{N}(i) \setminus j$ 表示与校验节点 s_i 连接的变量节点集

中除 x_j 外的其他变量节点集。

变量节点更新 $L^{(t)}(q_{ji})$ 表示为

$$L^{(t)}(q_{ji}) = L(p_j) + \sum_{i \in \mathbf{M}(j) \setminus i} L^{(t)}(r_{ij}), \quad (2)$$

式中: $\mathbf{M}(j) \setminus i$ 表示变量节点 x_j 参加的校验节点集中除 s_i 外的其他校验节点集。

由关键译码步骤可知,最小和算法将LLR-BP算法的节点更新中非线性双曲函数的运算简化为求最小值运算。但是最小和算法并没有充分考虑变量节点的更新过程,外部信息往往会翻转,造成误码率升高的问题。另外该算法属于洪泛调度,该调度方式会导致新的信息得不到及时利用,减缓译码收敛的速度。

3 shuffled-VNU-MS算法

为提高译码的性能,本文提出了新的译码算法——shuffled-VNU-MS。针对最小和算法不能将新信息及时利用的问题,本文改进算法引入混洗策略,在当前迭代对某列节点进行更新时,将前列在本次迭代已更新的最新变量节点信息用于该列的校验节点更新。针对外部信息值波动的问题,本文改进算法引入文献^[14]中减弱信息振荡的方法,在变量节点更新时引入加权因子 β ,使上次迭代的外部信息值也参与到当前迭代变量节点的更新过程中,以此达到修正外部信息的作用。

改进算法的具体译码步骤如下。

1) 初始化。

计算每一个变量节点的先验似然比信息 $L(p_j)$,表示为

$$L^{(0)}(q_{ji}) = L(p_j) = -\frac{2y_j}{\sigma^2}, L^{(0)}(r_{ij}) = 0, j=1, 2, \dots, N. \quad (3)$$

2) 按列进行校验节点更新。

依据本列之前列的本次迭代已更新信息和本列之后列的上一次迭代更新信息,更新本列校验节点信息,表示为

$$L^{(t)}(r_{ij}) = \prod_{j \in \mathbf{N}(i) \setminus j} \text{sign}[L^{(t)}(q_{ji})] \cdot \prod_{j \in \mathbf{N}(i) \setminus j} \text{sign}[L^{(t-1)}(q_{ji})] \cdot \min \left| \min_{\substack{j \in \mathbf{N}(i) \setminus j \\ j < j}} |L^{(t)}(q_{ji})|, \min_{\substack{j \in \mathbf{N}(i) \setminus j \\ j > j}} |L^{(t-1)}(q_{ji})| \right|. \quad (4)$$

3) 按列更新变量节点信息。

$$L^{(t)}(q_{ji}) = L(p_j) + \sum_{i \in \mathbf{M}(j) \setminus i} L^{(t)}(r_{ij}), \quad (5)$$

$$L^{(t)}(q_{ji}) = \beta L^{(t-1)}(q_{ji}) + (1 - \beta) L^{(t)}(q_{ji}), \quad (6)$$

先用传统的最小和变量节点更新算法[式(5)]计算本次变量节点的更新信息;在此基础上,依据式(6)

对前后两次迭代的变量节点更新信息求加权和,作为本次迭代变量节点最终的更新信息。加权系数 β 的值一般情况下设置为 0.9。

4) 更新伪后验概率 $L^{(t)}(q_j)$ 并进行译码判决。

$$L^{(t)}(q_j) = L(p_j) + \sum_{i \in M(j)} L^{(t)}(r_{ij}) \quad (7)$$

5) 对第 j 个码元进行译码判决。

$$\hat{x}_j = \begin{cases} 0, & L^{(t)}(q_j) > 0 \\ 1, & L^{(t)}(q_j) \leq 0 \end{cases} \quad (8)$$

6) 当 $\hat{\mathbf{x}} \cdot \mathbf{H}^T = 0$ 或 t 达到最大迭代次数 T 时,结束译码,把 $\hat{\mathbf{x}}$ 当作译码输出;否则 t 加 1,转入步骤 2 继续进行迭代。

引入混洗策略和改进变量节点更新的 MS 译码算法流程图如图 1 所示。

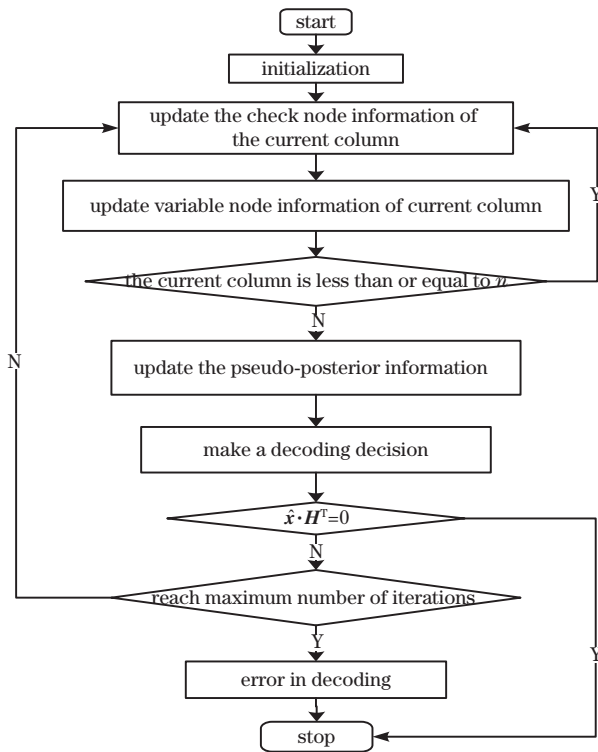


图 1 引入混洗策略和改进变量节点更新的 MS 译码算法流程图

Fig. 1 Flow chart of MS decoding algorithm that introduced shuffled strategy and improved variable node update

假设校验矩阵 \mathbf{H} 表示为

$$\mathbf{H} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} s_1 \\ s_2 \\ s_3 \end{matrix} \quad M=3 \times N=6 \quad (9)$$

$x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6$

现以 x_1, x_2 变量节点的更新为例,来说明 shuffled 策略的实施,以及对变量节点外部信息的加权更新。

在第 t 次迭代时,变量节点 x_1 作为第一个更新的节点,它所参与的校验节点更新 $L^{(t)}(r_{1,1})$ 和 $L^{(t)}(r_{2,1})$ 表示为

$$L^{(t)}(r_{1,1}) = f[L^{(t-1)}(q_{2,1}), L^{(t-1)}(q_{4,1})] \rightarrow L^{(t)}(r_{2,1}) = f[L^{(t-1)}(q_{3,2}), L^{(t-1)}(q_{5,2})] \quad (10)$$

可见,作为第一个更新的节点,它所参与的 $L^{(t)}(r_{1,1})$ 和 $L^{(t)}(r_{2,1})$ 更新只由 $t-1$ 次迭代的变量节点信息来决定。

变量节点 x_1 参与的变量节点更新 $L^{(t)}(q_{1,1})$ 和 $L^{(t)}(q_{1,2})$ 表示为

$$L^{(t)}(q_{1,1}) = f[\beta L^{(t)}(r_{2,1}), (1-\beta)L^{(t-1)}(r_{2,1})] \rightarrow L^{(t)}(q_{1,2}) = f[\beta L^{(t)}(r_{1,1}), (1-\beta)L^{(t-1)}(r_{1,1})] \quad (11)$$

可知, $L^{(t)}(q_{1,1})$ 和 $L^{(t)}(q_{1,2})$ 不仅与 t 次迭代的校验节点信息有关,还与 $t-1$ 次迭代的校验节点信息有关。

在第 t 次迭代时,变量节点 x_2 参与的校验节点更新公式为

$$L^{(t)}(r_{1,2}) = f[L^{(t)}(q_{1,1}), L^{(t-1)}(q_{4,1})] \rightarrow L^{(t)}(r_{3,2}) = f[L^{(t-1)}(q_{6,3})] \quad (12)$$

可见,当变量节点 x_1 与变量节点 x_2 在与校验节点 s_1 相连的变量节点集中时,作为第二个变量节点 x_2 更新 $L^{(t)}(r_{1,2})$ 时,不仅可以利用 $t-1$ 迭代的变量节点信息 $L^{(t-1)}(q_{4,1})$,还可以利用 t 次迭代的变量节点信息 $L^{(t)}(q_{1,1})$ 。

变量节点 x_2 参与的变量节点更新公式为

$$L^{(t)}(q_{2,1}) = f[\beta L^{(t)}(r_{3,2}), (1-\beta)L^{(t-1)}(r_{3,2})] \rightarrow L^{(t)}(q_{2,3}) = f[\beta L^{(t)}(r_{1,2}), (1-\beta)L^{(t-1)}(r_{1,2})] \quad (13)$$

可知, $L^{(t)}(q_{2,1})$ 和 $L^{(t)}(q_{2,3})$ 与 t 次迭代的校验节点信息和 $t-1$ 次迭代的校验节点信息有关。

从上面的例子可以看出:通过引入混洗策略,在迭代的过程中,按照变量节点先后顺序进行信息更新。先对第一个变量节点参与的变量节点信息和校验节点信息进行更新,当更新第二个变量节点时,若第二个变量节点和第一个变量节点同时与某一个校验节点相连,则会用本次迭代中更新的最新的变量节点信息;若更新时使用到后面节点的信息,则利用到上次迭代的变量节点信息。以此类推,更新其他变量节点的信息。

此外,通过加权因子 β 平均本次迭代和上一次

迭代的变量节点外部信息,使得上一次迭代的信息值也可以参与到当前迭代变量节点更新中,用以改正变量节点信息。

4 仿真实验及性能分析

4.1 仿真结果与分析

本文实验采用四种译码算法进行性能仿真,它们分别是 MS 算法、只引入混洗策略的 MS (shuffled-MS)算法、采用变量节点更新改进最小和 (VNU-MS)算法^[14]和本文 shuffled-VNU-MS 算法。

1) 采用结构化的准循环(QC)-LDPC 码构造校验矩阵 H ,码长 $N=256$,码率为 $1/2$,码组为 10000,最大迭代次数 T 设置为 25,用二进制相移键控(BPSK)调制方式在 AWGN 信道传输信息时,MS 算法、shuffled-MS 算法、VNU-MS 算法、shuffled-VNU-MS 算法的误码率随信噪比的变化曲线如图 2 所示。

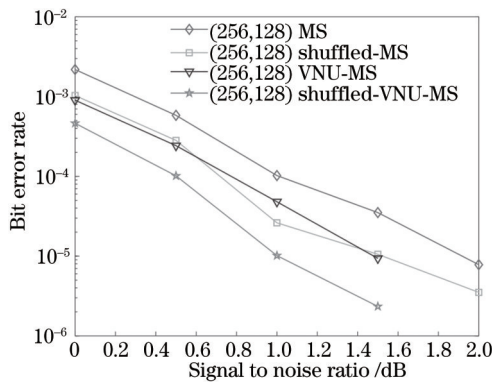


图 2 不同算法的误码率曲线图

Fig. 2 Bit error rate curves of different algorithms

从图 2 可以看出,码长 N 取 256 时,shuffled-VNU-MS 译码算法的误码性能优于 MS 算法、shuffled-VNU 算法、VNU-MS 算法。当误码率为 10^{-5} 时,shuffled-VNU-MS 译码算法的信噪比是 1 dB,shuffled-MS 算法信噪比是 1.5 dB,VNU-MS 算法的信噪比是 1.5 dB,MS 算法的信噪比是 1.92 dB。改进算法的编码增益相比其他三种算法得到了明显的提高。这是因为与其他三种算法相比,该改进算法既采用混洗策略,利用了最新的变量节点消息,从而使校验节点信息更加可靠,又通过引入加权因子 β ,使上次迭代的外部信息值也参与到当前迭代变量节点的更新过程中,减少了外部信息的振荡,进一步提升了信息的可靠性,从而提高译码质量。

2) 采用结构化的准循环 QC-LDPC 码构造校验矩阵 H ,码长 $N=256$,码率为 $1/2$,码组为 10000,最大迭代次数 T 设置为 25 次,用 BPSK 调制方式在 AWGN 信道传输信息时,MS 算法、shuffled-MS 算法、VNU-MS 算法、shuffled-VNU-MS 算法的平均迭代次数如图 3 所示。

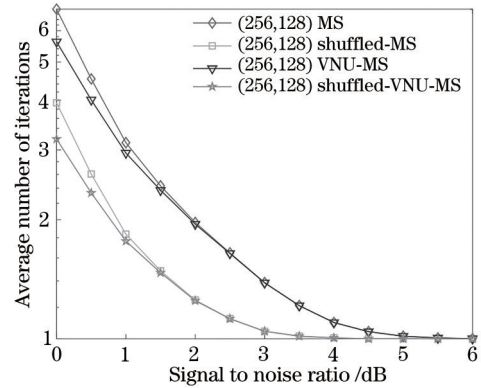


图 3 不同算法的平均迭代次数曲线图

Fig. 3 Average number of iterations of different algorithms

从图 3 可以看出,码长 N 取 256,在小信噪比时,平均迭代次数从多到少,依次是 MS 算法、VNU-MS 算法、shuffled-MS 算法、shuffled-VNU-MS 算法。当小信噪比为 0 时,本文改进的 shuffled-VNU-MS 算法相比 shuffled-MS 算法,译码速度提升约 20%;相比 VNU-MS 算法,译码速度提升约 43%;相比 MS 算法,译码速度提升约 52%。这表明,本文改进的 shuffled-VNU-MS 算法能够有效解决其他几种算法收敛速度慢的问题。这是由于 shuffled-VNU-MS 算法能够在本次迭代过程中,利用已更新列最新经过加权因子 β 修正的变量节点信息来更新当前列校验节点信息。这样就可以将更加可靠的变量节点信息用更快的方式进行传递,从而加快译码的收敛速度,减少译码所需的迭代次数。

4.2 算法复杂度分析

设准循环 QC-LDPC 码构造的校验矩阵 H ,行数为 M ,列数为 N ,行重为 d_r 。 e 表示 Tanner 图^[15]中边总数,则对于规则 LDPC 码而言 $e = d_r \times M$ 。作为洪泛调度的 MS 算法,在一次迭代的过程中,需要进行 e 次 $L(r_{ij})$ 的更新和 e 次 $L(q_{ji})$ 的更新。混洗策略的引入,只是节点信息按列进行更新,所以 shuffled-MS 算法复杂度与 MS 算法复杂度相同,都需要进行 e 次 $L(r_{ij})$ 的更新和 e 次 $L(q_{ji})$ 的更新。由于 VNU-MS 算法和 shuffled-VNU-MS 算法在变量

节点更新时,均需要增加一次加权求和运算,所以 VNU-MS 算法和 shuffled-VNU-MS 算法在进行 e 次 $L(q_{ji})$ 的更新时,两种算法分别需要增加 $d_r \times M$ 次的加法运算量。由于校验矩阵稀疏性 H 的特点, d_r 远远小于 M , 所以相比 MS 算法, 本文改进算法增加的这部分运算量很少。

5 结 论

本文针对最小和算法存在的不足,提出了一种基于混洗策略和变量节点更新改进最小和算法。所提 shuffled-VNU-MS 改进算法通过引入混洗策略,能够在本次迭代的下一列校验节点更新中,利用已更新列最新经过加权因子 β 修正的变量节点信息,使更加可靠的信息以更快的方式进行传递。仿真结果表明,当码长为 256, 码率为 0.5, 误码率为 10^{-5} 时, 本文改进的译码算法 shuffled-VNU-MS 相比 MS 算法、shuffled-MS 算法、VNU-MS 算法, 编码增益均得到提高。尤其相比 MS 算法, 编码增益提高了约 0.92 dB。当信噪比为 0 时, 本文改进算法译码收敛速度相比其他三种算法均得到明显提高。下一步将深入研究列重、度分布对 LDPC 译码性能的影响, 进一步提升译码纠错性能, 加快译码的收敛速度。

参 考 文 献

- [1] Gallager R. Low-density parity-check codes[J]. IRE Transactions on Information Theory, 1962, 8(1): 21-28.
- [2] 陈发堂, 刘一帆, 唐成. LDPC 码的一种低复杂度归一化最小和译码算法[J]. 重庆邮电大学学报(自然科学版), 2020, 32(1): 92-98.
Chen F T, Liu Y F, Tang C. A low-complexity normalized min-sum decoding algorithm for LDPC codes[J]. Journal of Chongqing University of Posts and Telecommunications (Natural Science Edition), 2020, 32(1): 92-98.
- [3] Dong J R, Yin H, Wang S J. An improved message passing schedule of BP-based decoding algorithm for LDPC codes[C]//2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data, December 15-18, 2016, Chengdu, China. New York: IEEE Press, 2016: 480-484.
- [4] Zhang H G, Yuan D F, Wang C X. An improved normalized BP based decoding algorithm for LDPC codes[C]//IET 2nd International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN 2008), Beijing, China. New York: IET, 2008: 223-226.
- [5] Han G J, Guan Y L, Huang X M. Check node reliability-based scheduling for BP decoding of non-binary LDPC codes[J]. IEEE Transactions on Communications, 2013, 61(3): 877-885.
- [6] Amaricai A, Bahrami M, Vasić B. A log-likelihood ratio based generalized belief propagation[C]//IEEE EUROCON 2019-18th International Conference on Smart Technologies, July 1-4, 2019, Novi Sad, Serbia. New York: IEEE Press, 2019: 19046821.
- [7] 郑娟毅, 孙宇, 张帆. 对数似然比置信传播算法的改进[J]. 现代电子技术, 2019, 42(15): 5-7, 12.
Zheng J Y, Sun Y, Zhang F. Improvement of log-likelihood ratio belief propagation algorithm[J]. Modern Electronics Technique, 2019, 42(15): 5-7, 12.
- [8] 高杰, 龙华, 邵玉斌, 等. 一种改进的 LLR BP 译码算法研究[J]. 传感器与微系统, 2017, 36(7): 70-72.
Gao J, Long H, Shao Y B, et al. Research on improved LLR BP decoding algorithm[J]. Transducer and Microsystem Technologies, 2017, 36(7): 70-72.
- [9] Kim N I, Lee S Q, Kim J U. A modified min sum decoding algorithm based on approximation enhancement for LDPC codes[C]//2020 International Conference on Information and Communication Technology Convergence (ICTC), October 21-23, 2020, Jeju, Korea (South). New York: IEEE Press, 2020: 1407-1410.
- [10] 钱方磊, 王秀敏, 常虹. 基于密度进化理论改进的 LDPC 码偏移最小和算法[J]. 电子器件, 2019, 42(3): 693-697.
Qian F L, Wang X M, Chang H. Improved offset Min sum algorithm of LDPC codes based on density evolution theory[J]. Chinese Journal of Electron Devices, 2019, 42(3): 693-697.
- [11] 杨卫国. 基于减少过估计的改进 LDPC 码最小和译码算法[J]. 指挥控制与仿真, 2017, 39(6): 53-57.
Yang W G. Modified LDPC codes min-sum algorithm based on reducing the over-estimation[J]. Command Control & Simulation, 2017, 39(6): 53-57.
- [12] Zhang X M, Tai Y. High-speed multi-block-row layered decoding for quasi-cyclic LDPC codes[C]//2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP), December 3-5, 2014, Atlanta, GA, USA. New York: IEEE Press, 2014: 11-14.

- [13] Zhao P Y, Qian C, Wang Z C, et al. Shuffled iterative receiver for LDPC-coded MIMO systems [C]//2015 IEEE International Conference on Communications, June 8-12, 2015, London, UK. New York: IEEE Press, 2015: 2768-2773.
- [14] 范芳, 冯雪林. 基于变量节点更新改进的最小和算法 [J]. 无线电工程, 2018, 48(2): 121-125.
Fan F, Feng X L. An improved Min-sum algorithm based on variable node update[J]. Radio Engineering, 2018, 48(2): 121-125.
- [15] Wang Y X, Yu H, Xu Y Y. Quasi-cyclic low-density parity-check convolutional code[C]//2011 IEEE 7th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), October 10-12, 2011, Shanghai, China. New York: IEEE Press, 2011: 351-356.