УДК 007.3
DOI: 10.26102/2310-6018/2024.47.4.019

Development of adaptive exponential min sum decoding algorithm

Zhang Weijia✉, **Ibrahem Mouhamad, V.M. Saklakov**

National Research Tomsk Polytechnic University, Tomsk, the Russian Federation

Abstract. This paper presents an optimized min sum (MS) decoding algorithm with low complexity and high decoding performance for LDPC short codes. The MS algorithm has low computational complexity and is simple to deploy. The MS decoding algorithm, while demonstrating a performance gap compared to the belief propagation (BP) and likelihood ratio BP (LLR-BP) decoding algorithms, shows significant potential for optimization. To improve the decoding performance of traditional MS algorithm, secondary external information is introduced into the control node (CNs) update operations of MS algorithm and optimized as adaptive exponential correction factor (AECF). The optimized MS algorithm is named as adaptive exponential exponential MS decoding algorithm (AEMS). The decoding efficiency of the AEMS algorithm for regular, irregular and LDPC codes of the Consultative Committee on Space Data Systems (CCSDS) was extensively tested, then the complexity of the AEMS algorithm was analyzed and compared with other decoding algorithms. The results show that the AEMS algorithm outperforms the offset MS (OMS) and normalized MS (NMS) algorithms in decoding performance, and outperforms the BP algorithm as the signal-to-noise ratio (SNR) gradually increases.

Keywords: LDPC, adaptive exponential algorithm, min sum, low complexity, LLR-BP.

For citation: Zhang Weijia, Ibrahem Mouhamad, Saklakov V.M. Development of adaptive exponential min sum decoding algorithm. *Modeling, Optimization and Information Technology.* 2024;12(4). URL: https://moitvivt.ru/ru/journal/pdf?id=1725 DOI: 10.26102/2310-6018/2024.47.4.003

Разработка адаптивного экспоненциального алгоритма декодирования минимальной суммы

Чжан Вэйцзя✉, **Ибрагим Мухамад, В.М. Саклаков**

Национальный исследовательский Томский политехнический университет, Томск, Российская Федерация

Резюме. В статье представлен оптимизированный алгоритм декодирования минимальной суммы (MS) с низкой сложностью и высокой производительностью декодирования для коротких кодов LDPC. Алгоритм MS имеет низкую вычислительную сложность и прост в развертывании. По сравнению с алгоритмом декодирования распространения убеждения (BP) и отношения правдоподобия BP (LLR-BP) он показывает разрыв в производительности декодирования, но алгоритм декодирования MS имеет высокий потенциал оптимизации. Для улучшения производительности декодирования традиционного алгоритма MS в операции обновления контрольных узлов (CN) алгоритма MS вводится вторичная внешняя информация и оптимизируется как адаптивный экспоненциальный поправочный коэффициент (AECF). Оптимизированный алгоритм MS назван адаптивным экспоненциальным алгоритмом декодирования MS (AEMS). Эффективность декодирования алгоритма AEMS для обычных, нерегулярных и LDPC-кодов консультативного комитета по системам космических данных (CCSDS) была всесторонне протестирована, затем был проведен анализ и сравнение сложности алгоритма AEMS с другими алгоритмами декодирования. Результаты показывают, что алгоритм AEMS превосходит алгоритмы смещенного MS (OMS) и нормализованного MS (NMS) по производительности декодирования, а также превосходит алгоритм BP по мере постепенного увеличения отношения сигнал/шум (SNR).

Ключевые слова: LDPC, адаптивный экспоненциальный алгоритм, минимальная сумма, низкая сложность, LLR-BP.

Для цитирования: Чжан Вэйцзя, Ибрагим Мухамад, Саклаков В.М. Разработка адаптивного экспоненциального алгоритма декодирования минимальной суммы. *Моделирование, оптимизация и информационные технологии.* 2024;12(4). (На англ.). URL: https://moitvivt.ru/ru/journal/pdf?id=1725 DOI: 10.26102/2310-6018/2024.47.4.019

Introduction

Low density parity check (LDPC) codes were first proposed by Dr. Gallager in 1962 and are a class of coding techniques with strong error correction capability [1]. Since LDPC codes have the advantages of low decoding complexity, parallel implementation, flexible structure, and low bit error level, they have been widely used in practical systems [2], and have been adopted as the 5G new radio (NR) channel coding scheme in the enhanced mobile broadband (eMBB) scenario [3]. With the deployment of 5G and the emergence of massive Internet of Things (IoT) devices, the importance of short to medium code decoders with high decoding performance, low complexity and low latency continues to grow [4].

LDPC codes achieve close to Shannon channel capacity through the belief propagation decoding algorithm (BP) and the BP likelihood ratio (LLR-BP) decoding algorithm, but usually for LDPC codes with relatively long code lengths [5]. Moreover, the BP and LLR-BP algorithms have high computational complexity, which makes it difficult to deploy for IoT devices with simple structure and low cost. On the basis of the LLR-BP algorithm, the min sum (MS) algorithm is proposed to convert the logarithmic operation in the LLR-BP algorithm into a comparison and summation operation [6], which greatly simplifies the computational complexity and is easy to deploy to IoT devices. The MS algorithm has gained simple computational complexity at the expense of decoding performance. In order to improve the decoding performance of the MS algorithm, the offset MS (OMS) and normalized MS (NMS) algorithms [7, 8] were proposed. These improved MS algorithms often use fixed error correction factors, and compared with the BP algorithm, there is still a decoding gap. In order to achieve better decoding performance, more effective optimizations in decoding are needed.

The derivation from the LLR-BP algorithm to the MS algorithm reveals that the MS algorithm simplifies the control node (CNs) update formula of the LLR-BP algorithm by replacing all external information with the smallest external information, thereby streamlining complex calculations. On this basis, we additionally introduce secondary (sub-smallest) external information value. In order to allow the sub-smallest external information value to optimize the check node update operation of the MS algorithm, we mathematically deform the sub-small external information value and use it as an exponential correction factor to improve the decoding performance of the MS algorithm. Compared with some existing MS algorithm optimization methods, we apply the adaptive exponential correction factor (AECF), which significantly improves the decoding performance of the MS algorithm.

The development process of decoding algorithms

The derivation process of the AEMS algorithm is easy to understand, but the derivation process involves a lot of knowledge about other decoding algorithms, especially the derivation from likelihood ratio BP (LLR-BP) algorithm to MS algorithm. In this section, we will briefly introduce the basic knowledge involved.

As the earliest proposed soft-decision decoding method, the BP algorithm has excellent decoding capabilities [1]. However, the BP algorithm contains a large number of multiplication operations and takes a long time to calculate, which also places higher requirements on hardware implementation. In order to simplify the BP decoding algorithm, the LLR-BP decoding algorithm, MS decoding algorithm, NMS decoding algorithm, OMS decoding algorithm have emerged one after another. Decoding algorithms are generally optimized using mathematical methods [9, 10]. In recent years, with the popularization of machine learning and the improvement of computer computing levels, machine learning are increasingly used to optimize decoding algorithms [4, 11]. Actually, whether the MS algorithm is optimized by mathematical methods or machine learning schemes, it is also difficult for the optimized MS algorithm to surpass the BP algorithm in decoding performance. The evolution process of soft decision decoding can be summarized as shown in Figure 1.

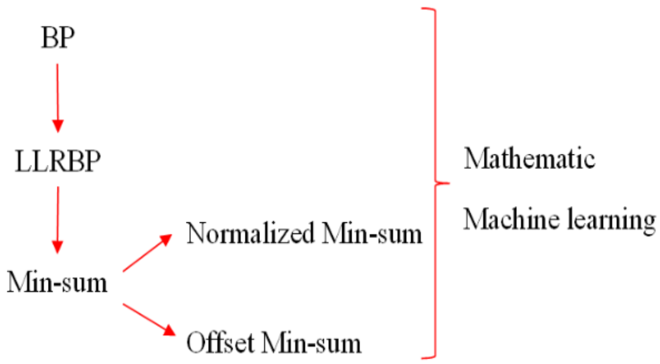


Figure 1 – Evolution process of soft decision decoding method
Рисунок 1 – Процесс эволюции метода декодирования мягкого решения

The soft decision decoding algorithm is an iterative decoding method based on the Tanner graph, where messages are exchanged back and forth between VNs and CNs during the iterative process. After several iterations, the message values stabilize, allowing for an optimal decision to be made accordingly [12]. In the decoding process, the information generated from the updates of CNs and VNs is referred to as external information, while the initial information received from the channel at the start of the decoding is known as the posterior probability.

Table 1 – Symbol explanation in the calculation process
Таблица 1 – Интерпретация символов при расчетах

Symbols	Meaning
v_i	The *I* variable node
c_j	The *j* check node
$r_{ji}^{l}(b)$	The external information passed from check node *j* to variable node *i* in the *l*th iteration, b = 0, 1
$q_{ij}^{l}(b)$	The external information passed from variable node *i* to check node *j* in the *l*th iterations, b = 0, 1
$C(i)$	The set of all check nodes connected to the *i*th variable node
$V(j)$	The set of all variable nodes connected to the *j*th check node
$C(i)$	The set of check nodes connected to the *i*th variable node except the *j*th check node
$V(j) \setminus i$	The set of variable nodes connected to the *j*th check node except the *i*th variable node
$P_i(b)$	The posterior probability of receiving y_i at the receiving end, corresponding to the code word $c_i = b$ at the sending end, b = 0, 1
$q_i^{l}(b)$	The posterior probability information of the *i*th variable node of the *l*th iteration, b = 0, 1

The LLR-BP algorithm is a logarithmic version of the BP algorithm and they have the same decoding performance. By taking logarithms on both sides of the equal sign of the BP decoding algorithm formula, multiple multiplications are turned into logarithmic domain addition calculations, making the BP algorithm easier to implement [13]. Since the BP algorithm computational process, and the derivation of BP to LLR-BP algorithm is not the focus of this paper, we start with LLR-BP algorithm in detail. The specific LLR-BP decoding algorithm as follows and the variable symbols designed in the calculation formula are shown in Table 1.

1) Initialization: Calculate the initial message of VNs as shown in (1).

$$L^0(q_{ij}) = L(P_i) = \ln\frac{P_i(0)}{P_i(1)} \quad (1)$$

2) CNs Update: The external message sent from the check nodes (CNs) to the variable nodes (VNs) is calculated as described in (2).

$$L^l(r_{ji}) = 2\tanh^{-1}\left(\prod_{i'\in V(j)\setminus i} \tanh\left(\frac{L^{l-1}(q_{i'j})}{2}\right)\right) \quad (2)$$

3) VNs Update: Calculate the external message passed from the VNs to the CNs as shown in (3).

$$L^l(q_{ij}) = \ln\frac{P_i(0)\prod_{j'\in C(i)\setminus j} r^l_{j'i}(0)}{P_i(1)\prod_{j'\in C(i)\setminus j} r^l_{j'i}(1)} = L(P_i) + \sum_{j'\in C(i)\setminus j} L^l(r_{j'i}) \quad (3)$$

4) Calculate the total VNs information: Calculate all the messages obtained by the VNs as shown in (4). If the value of $L^l(q_i)$ is larger than 0, v_i is determined to be 0, otherwise v_i is determined to be 1, then the code word $\tilde{v}$ obtained.

$$L^l(q_i) = \ln\frac{P_i(0)\prod_{j\in C(i)} r^l_{ji}(0)}{P_i(1)\prod_{j\in C(i)} r^l_{ji}(1)} = L(P_i) + \sum_{j\in C(i)} L^l(r_{ji}) \quad (4)$$

5) Decoding decision: If $\tilde{v}H^T = 0$ or the number of iterations reaches the maximum value, then the decoding stops, otherwise the algorithm returns to step 2.

The LLR-BP algorithm significantly reduces the number of multiplication operations while maintaining decoding performance. However, the calculation of the CN updates involves many *tanh* functions, which can be efficiently implemented using lookup table operations. When the LDPC code length is very long and a large number of table lookup operations will cause memory usage problems. In order to solve this problem, MS algorithm was proposed. MS algorithm simplifies the calculation of (2) even further by recognizing that the term corresponding to the smallest $L^{l-1}(q_{i'j})$ dominates the product term and so the product can be approximated by a minimum a simplified LLR-BP algorithm was developed [6]. Rewrite the *tanh* function of (2) into the product of the sign function and the absolute value function as shown in (5).

$$L^l(r_{ji}) = 2\prod_{i'\in V(j)\setminus i} sgn\left(L^{l-1}(q_{i'j})\right)\tanh^{-1}\left(\prod_{i'\in V(j)\setminus i} \tanh\left(\frac{\left|L^{l-1}(q_{i'j})\right|}{2}\right)\right) \quad (5)$$

Then replace $\prod_{i'\in V(j)\setminus i} \tanh\left(\frac{|L^{l-1}(q_{i'j})|}{2}\right)$ in (5) with $\tanh\left(\frac{|L^{l-1}(q_{i'j})_{min1}|}{2}\right)$. Combing with the $2\tanh^{-1}$ ($\tanh(\frac{x}{2})$) curve shown in Figure 2, we obtain the CNs information update formula of the MS algorithm as shown in (6).

$$L^l(r_{ji}) = \prod_{i' \in V(j)\setminus i} sgn\left(L^{l-1}(q_{i'j})\right) \min_{i' \in V(j)\setminus i} \left|L^{l-1}(q_{i'j})\right| \quad (6)$$

The calculation framework of the MS algorithm is the same as that of the LLR-BP algorithm, except that the CNs update formula is simpler.

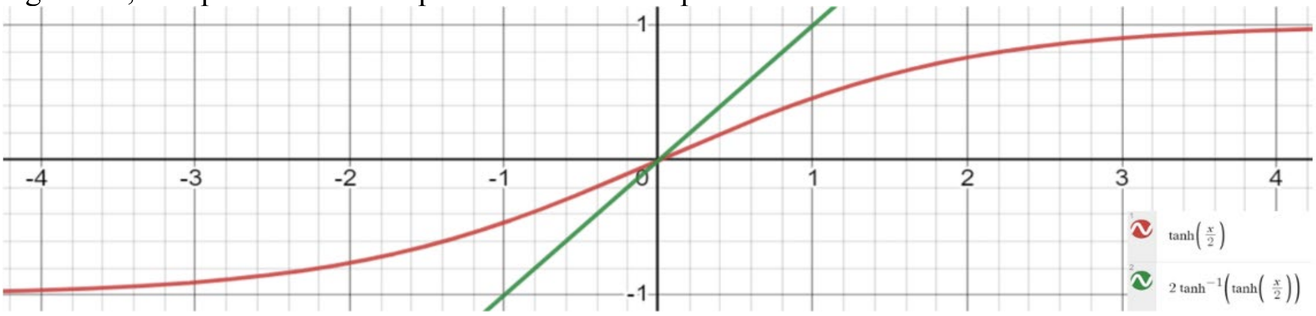


Figure 2 – MS related different curves
Рисунок 2 – Связанные с MS различные кривые функции тангенса

MS algorithm reduces the computational complexity of calculating $L^l(r_{ji})$, the CNs update formula (6) is a simplified scheme rather than an accurate calculation, so there will be a difference between $L^l(r_{ji})_{MS}$ and $L^l(r_{ji})_{LLRBP}$. According to the tanh $(\frac{x}{2})$ curve in Figure 2, it can be intuitively found that the any tanh $(\frac{|L^{l-1}(q_{i'j})|}{2} < 1$, and as the number of consecutive multiplications increases, the $\prod_{i' \in V(j)\setminus i}$ tanh $(\frac{|L^{l-1}(q_{i'j})|}{2})$ value must become smaller and smaller, so $\prod_{i' \in V(j)\setminus i}$ tanh $(\frac{|L^{l-1}(q_{i'j})|}{2}) <$ tanh $(\frac{|L^{l-1}(q_{i'j})_{min1}|}{2})$. Therefore, the obtained $|L^l(r_{ji})_{MS}|$ must be overestimated. Therefore, two more effective solutions, the NMS algorithm and the OMS algorithm are proposed [7] [8]. The main idea of the NMS algorithm is to adjust the calculation result by multiplying by a correction factor α (0 < α < 1) in the step of calculating the CNs update information value as shown in (7).

$$L^l(r_{ji}) = \alpha\left(\prod_{i' \in V(j)\setminus i} sgn\left(L^{l-1}(q_{i'j})\right) \min_{i' \in V(j)\setminus i} \left|L^{l-1}(q_{i'j})\right|\right) \quad (7)$$

The main idea of the OMS algorithm is to subtract a correction factor β (β > 0) in the step of calculating the CNs update information value as shown in (8).

$$L^l(r_{ji}) = \prod_{i' \in V(j)\setminus i} sgn\left(L^{l-1}(q_{i'j})\right) max\left(\min_{i' \in V(j)\setminus i} \left|L^{l-1}(q_{i'j})\right| - \beta, 0\right) \quad (8)$$

The NMS algorithm and the OMS algorithm, although processed in different ways, the core idea of both is to compensate for overestimating the update information value of CNs. However, both NMS algorithm and OMS algorithm have their own drawbacks [14]. The normalization factor α and the offset factor β can effectively correct the CNs update information value in the MS algorithm and improve the decoding performance, but α and β are constant during the decoding process and cannot be changed adaptively. Therefore, the NMS and OMS algorithms can only achieve limited improvements. In fact, NMS algorithm and OMS algorithm can hardly achieve the decoding performance of LLR-BP algorithm.

Although the NMS and OMS algorithms process CNs update formula differently, both fundamentally aim to address the overestimation of update information values of CNs. However, each algorithm has its limitations [14]. The normalization factor α and the offset factor β can effectively adjust the CNs update information values in the MS algorithm, enhancing decoding performance, but α and β remain constant during the decoding process, they cannot adaptively change. As a result, the improvements offered by the NMS and OMS algorithms are limited, and neither can match the decoding performance of the LLR-BP algorithm.

Adaptive exponential min sum algorithm model

In this paper, we directly use a simple mathematical scheme to optimize the MS algorithm. Its essence is to optimize the calculation method of $L^l(r_{ji})_{MS}$ so that its value will be closer to $L^l(r_{ji})_{LLRBP}$. The $|L^{l-1}(q_{i'j})_{min1}|$ has a great influence on $L^l(r_{ji})$. Just introducing $|L^{l-1}(q_{i'j})_{min1}|$ can make the MS algorithm obtain considerable decoding ability, although it is not as good as the LLR-BP algorithm. Based on the MS algorithm, we introduce $|L^{l-1}(q_{i'j})_{min2}|$ to reduce $L^l(r_{ji})_{MS}$, which can further improve the decoding performance of MS algorithm.

The inability of the normalization factor α and the offset factor β to change adaptively with the iterative process of the algorithm is the main reason for the limited performance of the NMS and OMS algorithms. $|L^{l-1}(q_{i'j})_{min2}|$ itself changes adaptively with the iterative process. In order to make $|L^{l-1}(q_{i'j})_{min2}|$ reduce the value of $L^l(r_{ji})_{MS}$, and considering the large number of nonlinear *tanh* operations in the $L^l(r_{ji})_{LLRBP}$ calculation, $|L^{l-1}(q_{i'j})_{min2}|$ can be used as an adaptive exponential correction factor (AECF) to reduce the value of $L^l(r_{ji})_{MS}$ and provide nonlinear characteristics. $|L^{l-1}(q_{i'j})_{min2}|$ cannot be directly used as an exponential correction factor, and needs to be deformed and optimized to keep its reduction effect on the min1 value within a reasonable range before it can play an optimization role. In the following we describe the derivation process of the adaptive exponential min sum (AEMS) algorithm. Since $|L^{l-1}(q_{i'j})_{min1}|$ and $|L^{l-1}(q_{i'j})_{min2}|$ are frequently used, they represent the minimum and second minimum of external information, so we use EI_{m1} and EI_{m2} instead of them to simplify the expression.

Calculating $L^l(r_{ji})_{AEMS}$ requires determining whether $EI_{m2} \le 1$ or not. If $EI_{m2} \le 1$, then EI_{m1} and EI_{m2} have the mathematical relationship as shown in (9).

$$0 < EI_{m1} \le EI_{m2} \le 1 \tag{9}$$

According to the characteristics of the *tanh* function, the mathematical relationship shown in (10) can be obtained.

$$\tanh^2\left(\frac{EI_{m1}}{2}\right) \le \tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right) \le \tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{1}{2}\right) \tag{10}$$

We follow the idea of MS algorithm and replace $\prod_{i'\in V(j)\backslash i} \tanh\left(\frac{|L^{l-1}(q_{i'j})|}{2}\right)$ in (5) with $\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right)$. Combined with the $2\tanh^{-1}\left(\tanh\left(\frac{x}{2}\right)\tanh\left(\frac{x}{2}\right)\right)$ and $2\tanh^{-1}\left(\tanh\left(\frac{x}{2}\right)\tanh\left(\frac{1}{2}\right)\right)$ curves shown in Figure 3, the following mathematical relationship can be obtained as shown in (11), (12) and it can be seen that $2\tanh^{-1}\left(\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right)\right)$ value will be located in the area 1, 2. Different areas are indicated by red numbers in Figure 3.

$$2\tanh^{-1}\left(tanh^2\left(\frac{EI_{m1}}{2}\right)\right) \le 2\tanh^{-1}\left(\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right)\right) \tag{11}$$

$$2\tanh^{-1}\left(\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right)\right) \le 2\tanh^{-1}\left(\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{1}{2}\right)\right) \quad (12)$$

The influence of AECF on the CNs update information value should be controlled within a certain range to ensure that AECF can improve the decoding performance of the MS algorithm. Use $diff = EI_{m2} - EI_{m1}$ to represent the difference between the two. As $diff$ becomes smaller, this means that EI_{m2} is closer to EI_{m1} , and in the extreme case $EI_{m2} = EI_{m1} \to 0$, so EI_{m2} will make a greater degree of reduction to $L^l(r_{ji})_{MS}$. As $diff$ becomes larger, this means that EI_{m2} is closer to 1 and in the extreme case $EI_{m1} \to 0, EI_{m2} = 1$, so EI_{m2} will make a smaller degree of reduction to $L^l(r_{ji})_{MS}$. According to the changing trend of $diff$ and the mathematical relationship between EI_{m1} and EI_{m2}, we can set up the AECF λ for the AEMS algorithm as shown in (13). Bringing in the values of the extreme cases above provides a clearer understanding.

$$\lambda = 2 - (EI_{m2} - EI_{m1}) \quad (13)$$

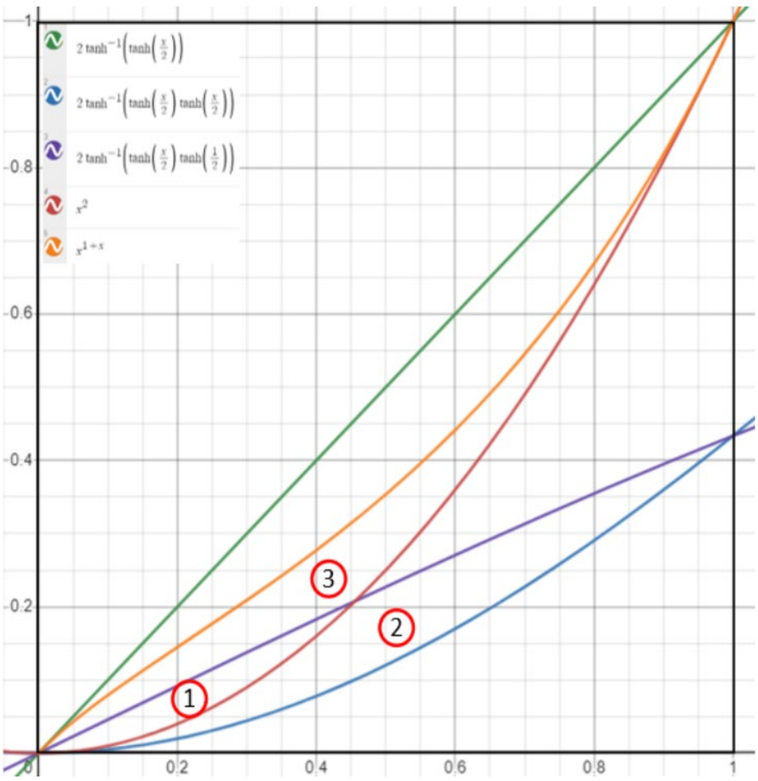


Figure 3 – AEMS related different tanh function curves
Рисунок 3 – Связанные с AEMS различные кривые функции тангенса

Substituting λ as an exponential factor in (6), we get AEMS algorithm CNs information update formula as shown in (14). Here we have used EI instead of the above $L^{l-1}\left(q_{i'j}\right)$, please pay attention to the difference

$$L^l(r_{ji})_{AEMS} = \prod sgn(EI) {EI_{m1}}^{\lambda}, EI_{m2} \le 1 \quad (14)$$

The following is a mathematical proof of the rationality of the $L^l(r_{ji})_{AEMS}$ value. Combined with (9) the value range of EI_{m2}, it can be seen that ${EI_{m1}}^{\lambda}$ has the mathematical relationship shown in (15) below.

$$EI_{m1}{}^{2} \leq EI_{m1}{}^{\lambda} \leq EI_{m1}{}^{1+EI_{m1}} \quad (15)$$

Combined with the x^2 and x^{1+x} curves shown in Figure 3, it can be seen that $EI_{m1}{}^{\lambda}$ value will be located in the area 1,3. When $EI_{m1}{}^{\lambda}$ is located in area 1, it has the following mathematical relationship as shown in (16).

$$2\tanh^{-1}\left(\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right)\right) \approx EI_{m1}{}^{\lambda} < |L^l(r_{ji})_{MS}| \quad (16)$$

When $EI_{m1}{}^{\lambda}$ is located in area 3, it has the following mathematical relationship as shown in (17). Taking (16) (17) together, the CNs update formula $L^l(r_{ji})_{AEMS_}$of AEMS is more reasonable compared to MS.

$$2\tanh^{-1}\left(\tanh\left(\frac{EI_{m1}}{2}\right)\tanh\left(\frac{EI_{m2}}{2}\right)\right) < EI_{m1}{}^{\lambda} < |L^l(r_{ji})_{MS}| \quad (17)$$

Here we have discussed the case of $EI_{m2} \leq 1$ and prove the rationality of the value from mathematical logic and function graph. When $EI_{m2} > 1$, EI_{m2} has no effect on reducing $L^l(r_{ji})_{MS}$, then the original CNs update formula of MS algorithm is used without changing as shown in (18).

$$L^l(r_{ji})_{AEMS} = \prod sgn(EI)EI_{m1}, EI_{m2} > 1. \quad (18)$$

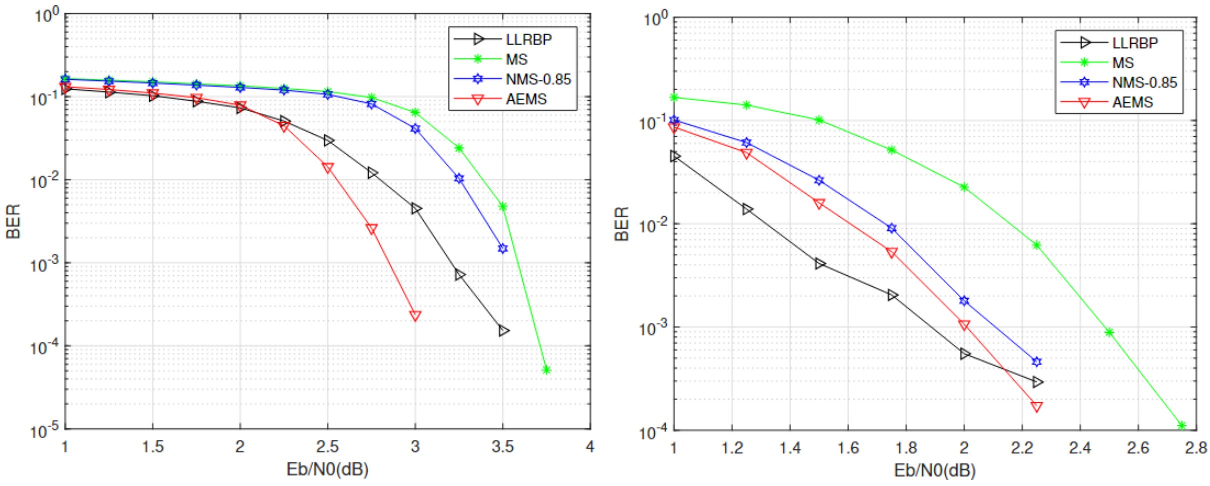
Finally, the CNs update formula of the AEMS algorithm is determined as shown in (19).

$$\begin{cases} L^l(r_{ji})_{AEMS} = \prod sgn(EI)EI_{m1}{}^{\lambda}, EI_{m2} \leq 1, \\ L^l(r_{ji})_{AEMS} = \prod sgn(EI)EI_{m1}, EI_{m2} > 1. \end{cases} \quad (19)$$

Simulation and comparison with others decoding algorithms

In this section, we compare the performance of the AEMS algorithm introduced in above with the LLR-BP, MS, and NMS algorithms in different LDPC codes, including regular LDPC codes, irregular LDPC codes, and three short block length CCSDS LDPC codes recommended by CCSDS 231.1-O-1 for TC synchronization and channel coding [15]. In the simulation, we applied additive white Gaussian noise (AWGN) channel and transmitted the coded bits through binary phase shift keying (BPSK) modulation, with the SNR range set to 1.0dB to 4.5dB, and frames set to 1000, and then calculated BER of different algorithms.

We use (2018, 1009) regular LDPC codes and (2018, 1009) irregular LDPC codes to test the decoding performance of AEMS algorithm, and the maximum number of iterations is set to 20. Since the structures of irregular LDPC codes and regular LDPC codes are quite different, this experiment is mainly to test the applicability of the AEMS algorithm in regular LDPC codes and irregular LDPC codes. The simulation results, displayed in Figure 4 (a) and Figure 4 (b), clearly demonstrate that the proposed AEMS algorithm significantly outperforms the MS and NMS algorithms, achieving superior BER performance at moderate to high SNR values. In Figure 4 (a), it is evident that when the SNR exceeds 2.0 dB, the BER performance of the AEMS algorithm begins to surpass that of the LLR-BP algorithm. The AEMS algorithm is the first to complete decoding process and has about a 0.5dB coding gain compared to the LLR-BP algorithm. In Figure 4 (b), it is shown that when the SNR exceeds 2.1 dB, the BER performance of the AEMS algorithm gradually improves, surpassing the LLR-BP algorithm. When the two complete decoding, the AEMS algorithm has about 0.1dB coding gain compared to the LLR-BP algorithm. The above results prove that the AEMS algorithm has excellent performance in both regular and irregular LDPC codes, and the structure of LDPC codes has limited impact on it.

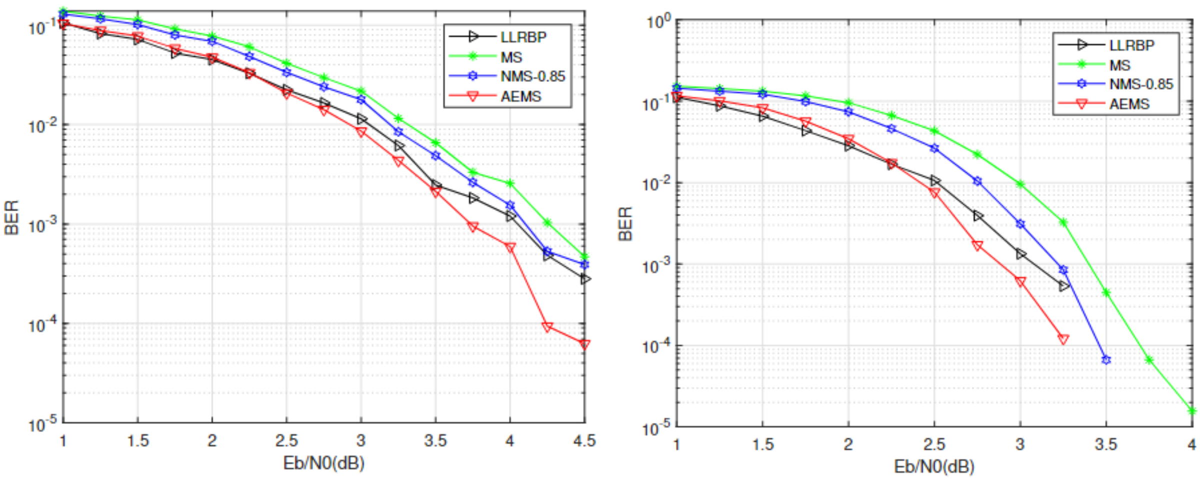


a b

Figure 4 – Simulation results: a – BER in regular LDPC codes; b – BER in irregular LDPC codes

Рисунок 4 – Результаты моделирования: a – BER в обычных кодах LDPC; b – BER в нерегулярных кодах LDPC

Two short block lengths recommended by CCSDS 231.1-O-1 for TC synchronization and channel coding are suitable for remote control instructions with short code lengths, and have good error correction capabilities and low decoding complexity [15]. We use (64, 128) and (256, 512) CCSDS LDPC codes and set the maximum number of iterations to 20 to test the practicality of the AEMS algorithm.



a b

Figure 5 – Simulation results: a – BER of (64, 128) CCSDS; b – BER of (256, 512) CCSDS

Рисунок 5 – Результаты моделирования: a – BER в (64, 128) CCSDS; b – BER в (256, 512) CCSDS

From Figure 5, it can be seen that the AEMS algorithm has excellent decoding performance in two short block CCSDS LDPC codes. Under medium and high SNR conditions, the AEMS algorithm decoding performance is optimal, basically when the SNR is greater than 2.5dB, the BER performance of AEMS begins to surpass LLR-BP algorithm. Since the code length of (64, 128) CCSDS LDPC codes is too short, the simulated BER curve is not very smooth. The BER curve of the (256, 512) CCSDS LDPC code is relatively smooth. As can be seen from Figure 5 (b), taking the BER performance of the AEMS algorithm when the SNR is 3 dB as the standard, AEMS has a 0.25 dB coding gain compared to the LLR-BP algorithm. The above results prove that the AEMS algorithm has high application value and is very suitable for short code length and high code rate control instruction type LDPC codes and its decoding performance is better than the LLR-BP algorithm in practical applications.

Compared to the MS algorithm, the AEMS algorithm adds one comparison, two addition operations, and one exponentiation operation in each calculation of the CN information value. This enhancement results in excellent decoding performance with only a modest increase in computational complexity and it is very suitable for the IoT devices that are mainly used for the short code-length and high-rate control commands.

Conclusion

In this paper, we propose the AEMS algorithm, which has excellent decoding performance with a slight increase in computational complexity. Through extensive simulations, our results have demonstrated that the AEMS algorithm consistently outperforms both the conventional MS and NMS algorithms. Notably, the AEMS algorithm exhibits superior decoding accuracy, surpassing the LLR-BP algorithm across various LDPC code scenarios, including regular, irregular, and short block CCSDS LDPC codes with low code rates. Moreover, considering the evolving landscape of IoT devices, the AEMS algorithms blend of low computational complexity, minimal SNR requirements, and exceptional decoding capabilities positions it as a compelling choice for future IoT device decoding algorithms.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Gallager R. Low-density parity-check codes. *IRE Transactions on Information Theory.* 1962;8(1):21–28. https://doi.org/10.1109/TIT.1962.1057683
2. Sarvaghad-Moghaddam M., Ullah W., Jayakody D.N.K., Affes S. A New Construction of High Performance LDPC Matrices for Mobile Networks. *Sensors.* 2020;20(8). https://doi.org/10.3390/s20082300
3. Weijia Z., Jayakody D.N.K. On the competitiveness of LDPC codes in wireless. In: *2nd International Research Conference, SLTC, 29–30 September 2022, Padukka, Sri Lanka.*
4. Nachmani E., Marciano E., Lugosch L., Gross W.J., Burshtein D., Be'ery Y. Deep Learning Methods for Improved Decoding of Linear Codes. *IEEE Journal of Selected Topics in Signal Processing.* 2018;12(1):119–131. https://doi.org/10.1109/JSTSP.2017.2788405
5. MacKay D.J.C., Neal R.M. Near Shannon Limit Performance of Low Density Parity Check Codes. *Electronics Letters.* 1997;33(6):457–458.
6. Fossorier M.P.C., Mihaljevic M., Imai H. Reduced complexity iterative decoding of low-density parity check codes based on belief propagation. *IEEE Transactions on Communications.* 1999;47(5):673–680. https://doi.org/10.1109/26.768759
7. Chen J., Fossorier M.P.C. Near optimum universal belief propagation based decoding of low-density parity check codes. *IEEE Transactions on Communications.* 2002;50(3):406–414. https://doi.org/10.1109/26.990903
8. Chen J., Fossorier M.P.C. Density evolution for two improved BP-Based decoding algorithms of LDPC codes. *IEEE Communications Letters.* 2002;6(5):208–210. https://doi.org/10.1109/4234.1001666

9. Zhao J., Zarkeshvari F., Banihashemi A.H. On implementation of min-sum algorithm and its modifications for decoding low-density Parity-check (LDPC) codes. *IEEE Transactions on Communications.* 2005;53(4):549–554. https://doi.org/10.1109/TCOMM.2004.836563
10. Roberts M.K., Mohanram S.S., Shanmugasundaram N. An Improved Low Complex Offset Min-Sum Based Decoding Algorithm for LDPC Codes. *Mobile Networks and Applications.* 2019;24(6):1848–1852. https://doi.org/10.1007/s11036-019-01392-7
11. Wu X., Jiang M., Zhao C. Decoding optimization for 5g ldpc codes by machine learning. *IEEE Access.* 2018;6:50179–50186. https://doi.org/10.1109/ACCESS.2018.2869374
12. Jadhav M.M., Pancholi A., Sapkal A.M. Analysis and implementation of soft decision decoding algorithm of ldpc. *International Journal of Engineering Trends and Technology.* 2013;4(6):2380–2384.
13. Lakshmi R., Tony T., Raju A.J. An analytical approach to the performance of Low Density Parity Check Codes. In: *2013 International Conference on Advanced Computing and Communication Systems, 19–21 December 2013, Coimbatore, India.* IEEE; 2013. pp. 1–4. https://doi.org/10.1109/ICACCS.2013.6938732
14. Jiang M., Zhao C., Zhang L., Xu E. Adaptive offset min-sum algorithm for low-density parity check codes. *IEEE Communications Letters.* 2006;10(6):483–485. https://doi.org/10.1109/LCOMM.2006.1638623
15. CCSDS Historical Document "Short Block Length LDPC Codes for TC Synchronization and Channel Coding" CCSDS 231.1-O-1 (2015). URL: https://public.ccsds.org/Pubs/231x1o1s.pdf [Accessed 12th September 2024].

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Чжан Вэйцзя, аспирант, ассистент, Национальный исследовательский Томский политехнический университет, Томск, Российская Федерация.
e-mail: victoryzh@tpu.ru
ORCID: 0000-0003-2252-2750

Zhang Weijia, Postgraduate Student, Assistant Lecturer, National Research Tomsk Polytechnic University, Tomsk, the Russian Federation.

Ибрагим Мухамад, аспирант, ассистент, Национальный исследовательский Томский политехнический университет, Томск, Российская Федерация.
e-mail: ibragim1@tpu.ru
ORCID: 0000-0003-1569-5493

Ibrahem Mouhamad, Postgraduate Student, Assistant Lecturer, National Research Tomsk Polytechnic University, Tomsk, the Russian Federation.

Саклаков Василий Михайлович, старший преподаватель, Отделение информационных технологий, Национальный исследовательский Томский политехнический университет, Томск, Российская Федерация.
e-mail: saklavas@tpu.ru
ORCID: 0000-0003-1716-4581

Vasiliy M. Saklakov, Senior Lecturer in Division for Information Technology, National Research Tomsk Polytechnic University, Tomsk, the Russian Federation.

Статья поступила в редакцию 21.10.2024; одобрена после рецензирования 12.11.2024; принята к публикации 20.11.2024.

The article was submitted 21.10.2024; approved after reviewing 12.11.2024; accepted for publication 20.11.2024.