Deep Learning Algorithms for BCH Decoding in Satellite Communication

Xin Ruan *

Department of Electronics and Information Engineering, Hong Kong Polytechnic University, Hong Kong, 999077, China

* Corresponding Author Email: 21109434g@connect.polyu.hk

Abstract. Deep learning is widely used in various fields due to the advancement of algorithms, the enrichment of high-efficiency databases, and the increase in computing power. Especially in the satellite communication, the learning and parallel computing capabilities of neural networks make them ideal for decoding. Many researchers have recently applied deep learning neural networks to decode high-density parity check (HDPC) codes (such as BCH and RS code), improving the decoder's performance. This review aims to provide general insights on applying neural network decoders to satellite communications. Due to the neural network's learning ability, the neural network-based decoder can be trained to change the weights, thereby reducing the influence of non-white noise in satellite communications, such as the influence between the satellite and the terrestrial network and the mutual interference within the satellites. To compensate for non-white noise, shortest circles in Tanner graph and unreliable information, a decoder system model for satellite communication constructed by three neural networks is presented.

Keywords: Channel coding, BCH code, Satellite communication, Error-correcting Coding, Neural network.

1. Introduction

Internet speed and network security requirements have increased as autonomous driving, and the Internet of Things (IoT) have developed in recent years. As one of the significant Error Correction Codes, BCH codes can correct many random errors, making it the most substantial and well-studied High-Density Parity Check codes (HDPC). A multi-base BCH code that can also fix multiple symbol problems is the Reed-Solomon code (RS code). At the same time, deep learning has also become a research hotspot and has been applied to data analysis, speech recognition, image recognition, and other fields. The fitting and expressive abilities of neural networks are potent, and deep learning is suitable for HDPC decoding [1,2].

In 1948, through the seminal publication "A Mathematical Theory of Communication," Shannon initially put forward a strategy to ensure dependable communication in disrupted channels [4]. This famous disturbed channel coding theorem created the groundwork for error-correcting codes. Shannon found that the maximum speed at which data can be transmitted without error within any communication channel is related to noise and bandwidth. This maximum bit rate represents the channel capacity, now known as the Shannon limit. In order to improve the reliability and effectiveness of information transmission in satellite communication, researchers have been seeking to implement better coding and decoding methods with appropriate complexity to approach the ideal bounds of Shannon's theory.

In 1950, Hamming proposed an error-correcting linear block code, the Hamming code. In 1954, Muller and Reed proposed an RM (Reed-Muller) code with more vital error correction ability than Hamming code. In 1960, Hocquenghem, Bose, and Ray-Chaudhuri proposed a cyclic code with robust error correction ability in their paper, which was named after the initials of the three, namely BCH (Bose–Chaudhuri–Hocquenghem) code [5,6]. It's a type of cyclic code that can remedy random faults. Around 1960, Peterson theoretically solved the decoding algorithm of binary BCH code. He theoretically realized encoding, decoding, and error correction by constructing polynomials in Galois fields [7]. In 1965, Berlekamp developed an innovative decoding technique that could nearly 8-times improve the decoder's speed and reduce its storage needs by roughly 4 times [8]. Massey simplified the decoding process in 1969 by applying it to the linear feedback shift register. The combination of the two great works is the so-called Berlekamp-Massey algorithm nowadays (BM algorithm). For low-density parity check code (LDPC) decoding, Tanner [9] proposed a Tanner graph in 1981. Parallel decoding can significantly reduce decoding complexity. Pearl [10] suggested a belief propagation (BP) algorithm in 1982. According to the BP method, each node in a Markov random field has a probability distribution state, which is transferred to neighbouring nodes through message propagation and changes their probability distribution state. Each node's probability distribution will reach a stable state after a predetermined number of rounds. After that, the researchers found that combining BP and Tanner graphs can further enhance the efficiency of LDPC decoding. Belief propagation decoding algorithm became a hot research topic at that time. The belief propagation decoding algorithm is a Tanner graph-based iterative decoding technique. Iteratively, dependability information is transferred back and forth between variable nodes and check nodes via edges of the Tanner graph, which eventually converges to a stable value after many iterations, and the optimal option is determined. In LDPC decoding, the BP decoding method performs admirably. Many researchers have attempted to improve the BP algorithm by applying it to HDPC decoding.

Due to the learning capability of neural networks, the new decoder built with neural networks can attenuate the effects of non-white noise in the environment. Due to the high integration of satellites and shared channels [3], there will be much non-white noise in the channel. And these non-white noises can be analyzed and learned by neural networks to reduce their impact. Compared with traditional BCH decoding, based on neural networks, this paper presents a decoding system model with stronger decoding and anti-interference ability. This article's innovation is combining three neural networks with different functions to build a better decoder than traditional decoders in satellite communications from three aspects: compensating for non-white noise, classifying variable nodes and reducing shortest circles. This decoding system model is mainly used to solve the increasing signal interference between satellite and terrestrial users. This paper will first introduce the principles of BCH codes and RS codes and their traditional encoding and decoding methods. Three new decoders that introduce neural networks over traditional decoding methods will be introduced later. Finally, a comparative analysis of the three neural network decoders will be carried out, and the three will be combined to form a decoder system.

2. Principles of BCH coding and RS coding

2.1. The coding principle of BCH

In order to understand the working principle of the BCH code, it is first necessary to understand the coding principle of the BCH code. Like other error correction codes, BCH encoding converts the information to be sent by the source into codewords with redundant bits. BCH decoding is to convert the received codeword into the information sent by the source and perform error correction. The BCH coding and decoding procedures usually use Galois filed $GF(q)$ for convenient byte-oriented processing on computers. For binary BCH codes, $GF(2^m)$ is often used, where m is a positive integer. Let α be the generator of $GF(2^m)$. The generator must be such that the values $\{\alpha^0, \alpha^1, \alpha^2, \ldots, \alpha^{F-1}\}$ are all unique and non-zero, where F is the size of the Galois field. Add zero element 0 to form the extended field $GF(2^m)$.

Let the BCH codeword has n bits with k information bits. Then the length of the redundant bits is $n-k$. Record the unencoded k-bit data as a polynomial:

$$m(x) = m_0 + m_1 \cdot x^1 + m_2 \cdot x^2 + \cdots + m_{k-1} \cdot x^{k-1} \tag{1}$$

Where $\{m_0, \cdots m_{k-1}\}$ belongs to $\{0,1\}$. Then, the generator polynomial can be written as:

$$g(x) = \prod_{i=0}^{n-k-1}\left(x - \alpha^i\right) = (x - \alpha^0)\cdots(x - \alpha^{n-k-1}) \tag{2}$$

To get the check digit polynomial, divide the generator polynomial g(x) by $x^r \cdot m(x)$ as:

$$r(x) = x^r \cdot m(x) mod\ g(x) \tag{3}$$

Finally, the encoded BCH codeword polynomial, which is the message to be sent, can be expressed as follows:

$$C(x) = x^r \cdot m(x) + x^r \cdot m(x) mod\ g(x) \tag{4}$$

RS code is similar to BCH code. The code space of the BCH code is on $GF(2)$. And the check space is on $GF(2^m)$. The code space and check space of the RS code are both on GF(2^m). The decoding algorithm of RS code and BCH code are the same.

2.2. The BP decoding algorithm of RS codes and BCH code

After the BCH codeword is transmitted in the channel. Noise $E(x)$ will be added to the signal. Then the message becomes:

$$R(x) = C(x) + E(x) \tag{5}$$

The traditional decoding method calculates the syndrome from the BCH codewords message R(x). The error patterns can be found in the estimated syndrome. The received codeword information is subtracted from the error pattern. Then the codeword most similar to the one without noise is obtained. However, the traditional decoding method is complex and inefficient. Jing Jiang et al.[11] proposed a stochastic shifting-based iterative decoding (SSID) algorithm, which utilized the BP algorithm to perform HDPC decoding. Jing Jiang et al. utilized a log-likelihood ratio (LLR) to indicate the reliability of the receiving bits. Suppose the coded bit is transmitted with BPSK modulation in the channel with additive white Gaussian noise (AWGN). The signal can be expressed as $y = x + n$, where y is corresponding to received bits, x is corresponding to the sent bits and n is the Gaussian noise. Then the posterior LLR of each received bit can be expressed as:

$$L(x_i) = log\frac{P(c_i=0|y)}{P(c_i=1|y)} \tag{6}$$

To mitigate deterministic errors, Jing Jiang et al. utilized a sum-product algorithm (SPA). Let L^j denote the sum of the received bit's LLR and the extrinsic LLR produced in the jth iteration. L^j can be expressed as:

$$L^{j+1} = L^j + \alpha L_{ext}^j \tag{7}$$

Where $\alpha \in (0, 1]$ is a damping coefficient. Then define L_θ to be the LLR cyclically shifted by θ, which is an integer belonging to $(0, n-1)$. $\psi(L^j)$ is defined as the one SPA iteration with input LLR L. Then the stochastic shifting-based iterative decoding (SSID) Proposed by Jing Jiang et al. can be described by the following flow chart:

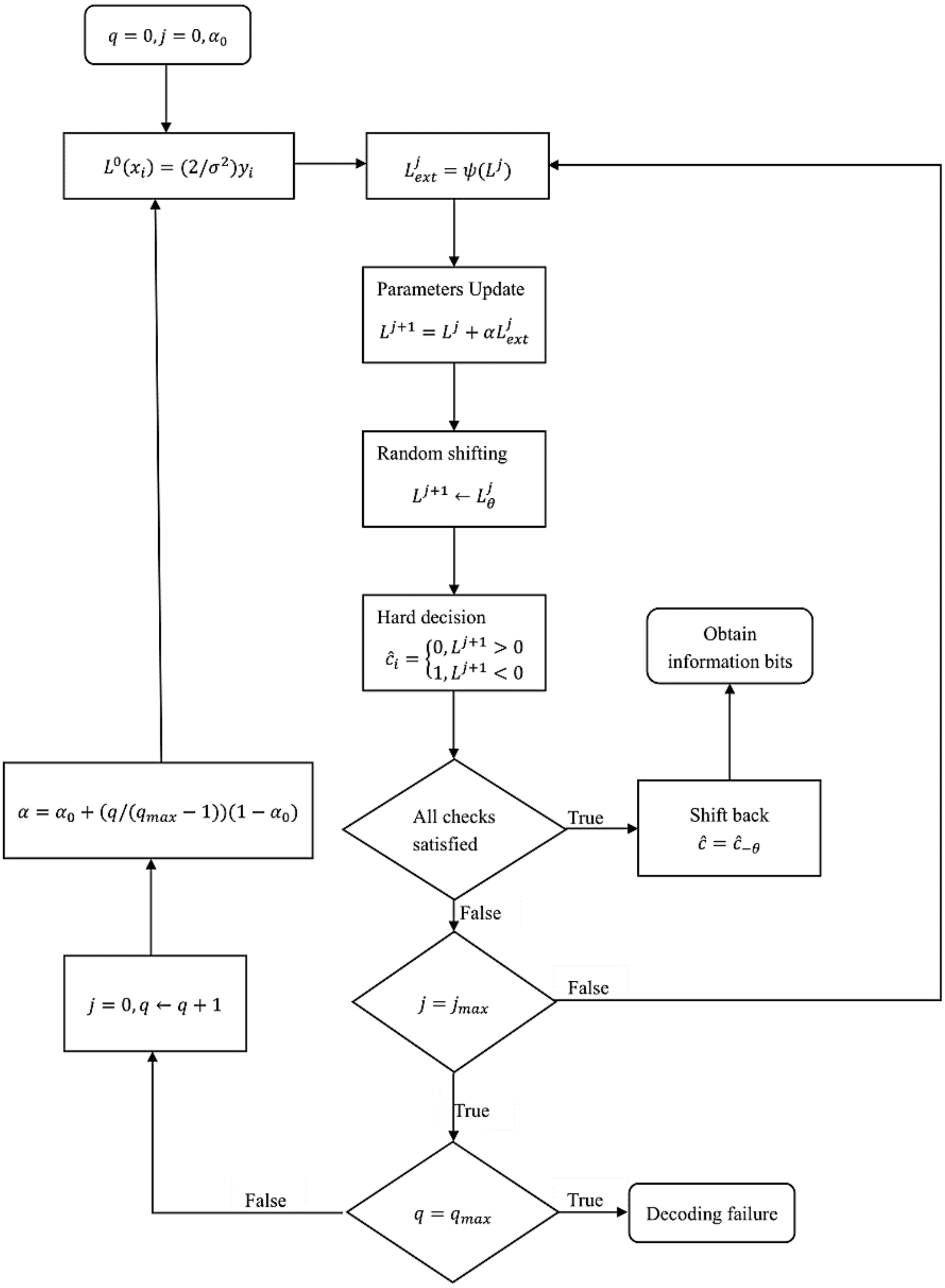

Figure 1. Flowchart of SSID algorithm

SSID algorithm performs BP algorithm to decode RS codes. Halford et al.[12] proposed a random redundant iterative decoding algorithm (RRD) for the BCH codes decoding based on the redundant Tanner graph. RRD algorithm is an extended version of SSID. RRD algorithm is a soft-input soft-output (SISO) algorithm. The RRD algorithm is mainly constructed with three loops. An inner loop called e loop is to realize BP iterative algorithm. A middle loop named h-loop is to update LLR value and bits permutation. An outer loop called q-loop is to update the damping ration α for improving convergence behaviour [13]. The inner BP iterative e-loop performs BP iteration between variable node (VN) and check node (CN). Let v_i represents the ith variable node and c_j for the jth check node. Let $L^{q,h,e}_{v_i \to c_j}$ represent the LLR information conveyed from the ith variable node to the jth check node (V2C) in the (q, h, e) iteration. In the same way $L^{q,h,0}_{c_j \to v_i}$ represents the LLR information conveyed from the jth check node to the ith variable node (C2V) in the $(q, h, 0)$ iteration. Thus, the structure of RRD algorithm can be introduced in figure 2:

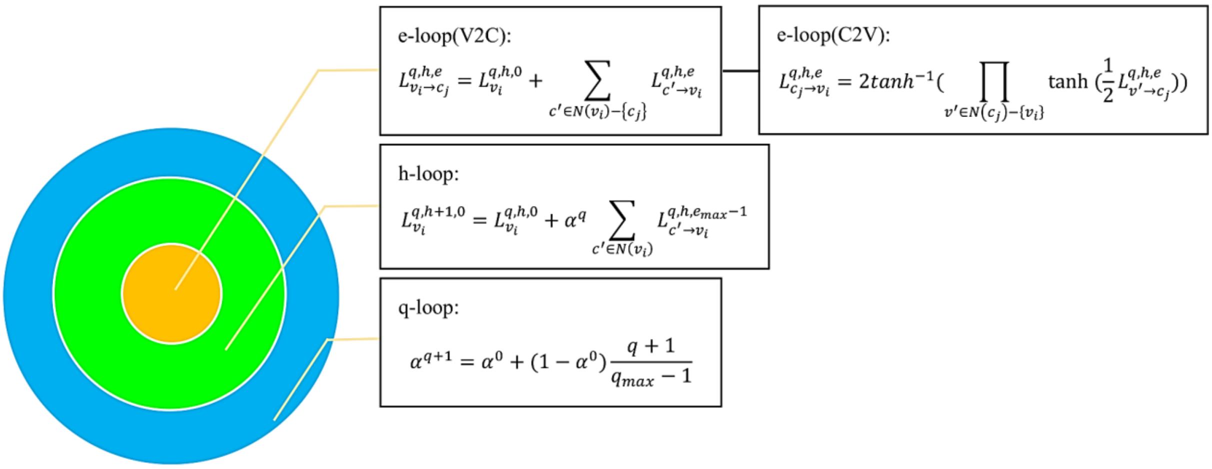

Figure 2. Structure of RRD algorithm [13]

Due to their excellent flexibility and learning ability, neural networks are very suitable for decoding error-correcting codes [2]. One of the most significant advantages of neural networks is that they can adjust the weights to reduce the impact of channel defects. For example, there may be some noise with specific characteristics in the signal's channel. Neural network decoders can specifically lessen the effects of these noises by adjusting the weights. For the BP algorithm, neural networks can also be used to reduce the influence of some low-confidence nodes on the decoding process, thereby improving the decoding performance. Next, three algorithms that apply the neural network to BCH decoding will be introduced to illustrate the role of neural networks in BCH decoding.

3. Neural Networks Decoder for Channels with Non-white Noise

Ortuno et al. [2]presented a neural network for decoding BCH (7,4) code over AWGN channel. They built two neural networks to decode BCH (7,4) code with different noise conditions. The rest of this section will review and discuss their process and discussion.

3.1. Neural Networks for BCH (7,4) code with white noise

There exist 4 information bits with BCH (7,4) code. And it will be sent as 7 bits sequence. There exist $2^4 = 16$ codewords. Ortuno et al. [2] built a neural network decoder for BCH (7,4) code, which is shown in figure 3

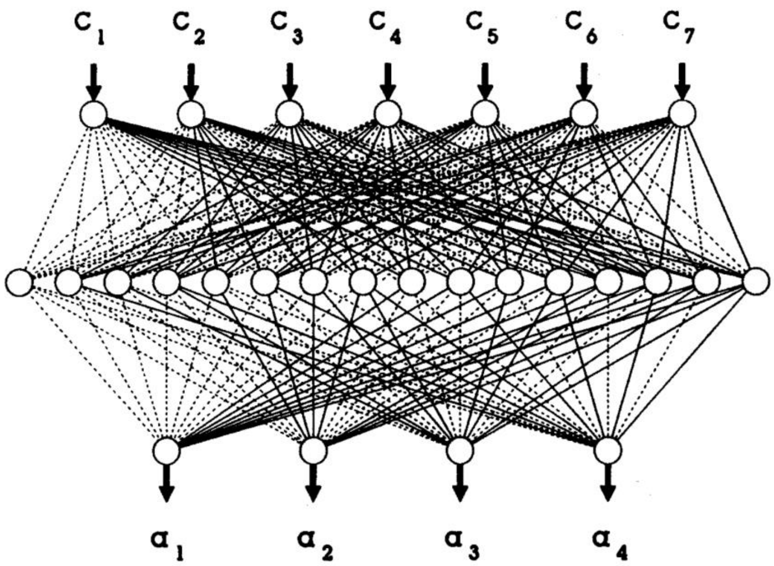

Figure 3. Neural network decoder for BCH (7,4) code [2]

In figure 3, the received 7 bits message are put into the corresponding 7 input nodes $(c_1, c_2 \cdots c_7)$. These input nodes are connected to the $16(2^4)$ intermediate neurons. And the final 4 nodes α_1 to α_4 denotes the 4 information bits. The first use the neural networks with fixed weight to decode the receiving message with white noise, which implies the noisy bits are uncorrelated and their neural networks do not have learning capability. They set the solid lines in figure 3 to represent $+1$ and the dashed lines to represent -1. Their neural networks are equivalent to a maximum likelihood decoding (MLD) device in this case.

3.2. Neural Networks for the BCH (7,4) code with non-white noise

Then, they change the noise signals to have a non-uniform power spectrum, which implies the noisy bits are correlated. At this time, they use a neural learning network to do the decoding, implying the neural network has learning capability. Due to the learning capability of neural networks, the decoding performance of their model is better than their original neural networks mentioned in section 3.1.

3.3. Experiment Result

Ortuno et al. [2] test the bit error probability of the neural networks with different noise correlation coefficients. The result is shown in figure 4.

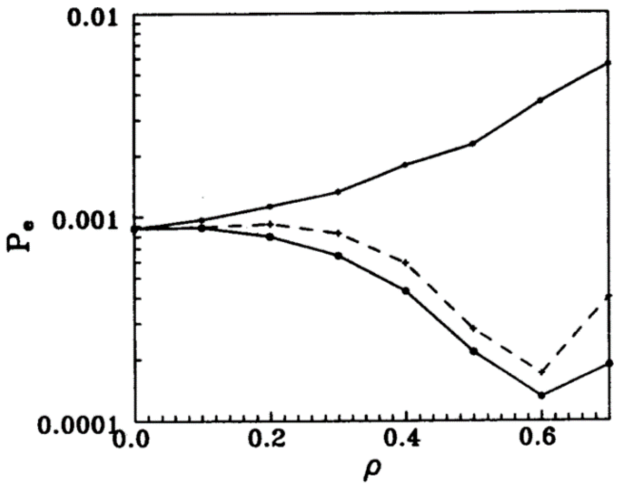

Figure 4. Bit error probability results for BCH (7,4) with different noise correlation coefficients [2]

In figure 4, the solid upper curve denotes the result of applying the original neural network mentioned in section 3.1 to decode the non-white noise with different noise correlation coefficients. And the lower solid curve refers to the neural net mentioned in section 3.2. Ortuno et al. [2] also utilized a standard error back propagation learning algorithm to train the original neural network. The result is the dashed curve in figure 4. The result shows that the neural networks can change their weights to adjust channels with a different noise due to the learning capability. The decoding performance of neural networks will become much better after a time of learning. The learning capability of this neural network decoder can be applied to satellite communications to solve non-white noise interference. This decoder can be used as the first stage of the satellite communication decoding system to lessen the effect of non-white noise generated by the other systems in the satellite and the shared channel. In the next sections, two neural network decoders will be introduced. The second decoding algorithms utilize neural networks to reduce the shortest circles of the Tanner graph [14]. The third algorithm utilized neural networks to do the variable node classification to limit the transmission of unreliable information. The second and third decoders can be used as the second stage of the decoding systems to improve the decoder performance further.

4. Neural Network Decoder Based on BP Decoding Algorithm for Compensating Shortest Circles

Eliya Nachmani et al. [15] proposed a deep learning method to enhance the performance of the BP decoding algorithm for HDPC decoding. They built their neural networks based on the Tanner graph tunable weights on edges. The neural network's hidden layers correspond to the edges of the Tanner graph. Then the weights in the hidden layers are trained by stochastic gradient descent to compensate for the shortest circles in the Tanner graph, which adversely affects the decoding performance [14]. Sigmoid neurons are used in the last layer to make the final output in the range of [0,1]. The architecture of the deep neural network decoder for BCH (15,11) is shown in figure 5.

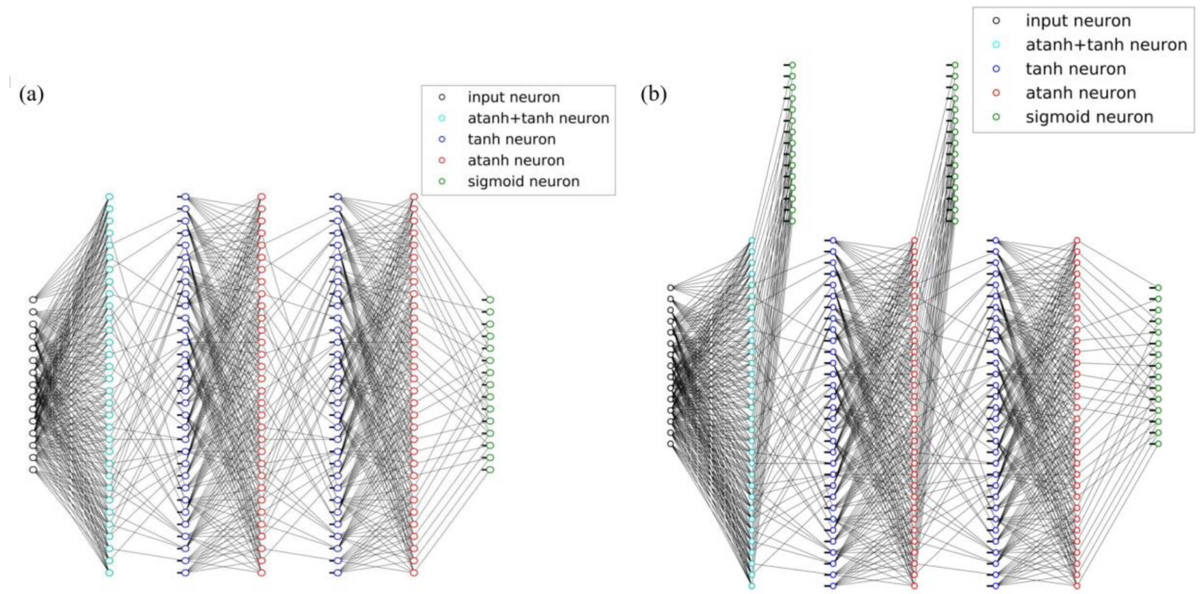

Figure 5. (a) Architecture of deep neural network decoder for BCH (15,11), (b) Architecture of deep neural network decoder for BCH(15,11) with training multiloss[15].

4.1. Experiment Result

The transmitting data is the zero codewords transmitted with an SNR of 1dB to 6dB in the AWGN channel. The bit error rate of BP algorithms and the proposed deep neural decoder is compared as a function of the signal-to-noise ratio (SNR). The comparison results are shown in Figure 6.

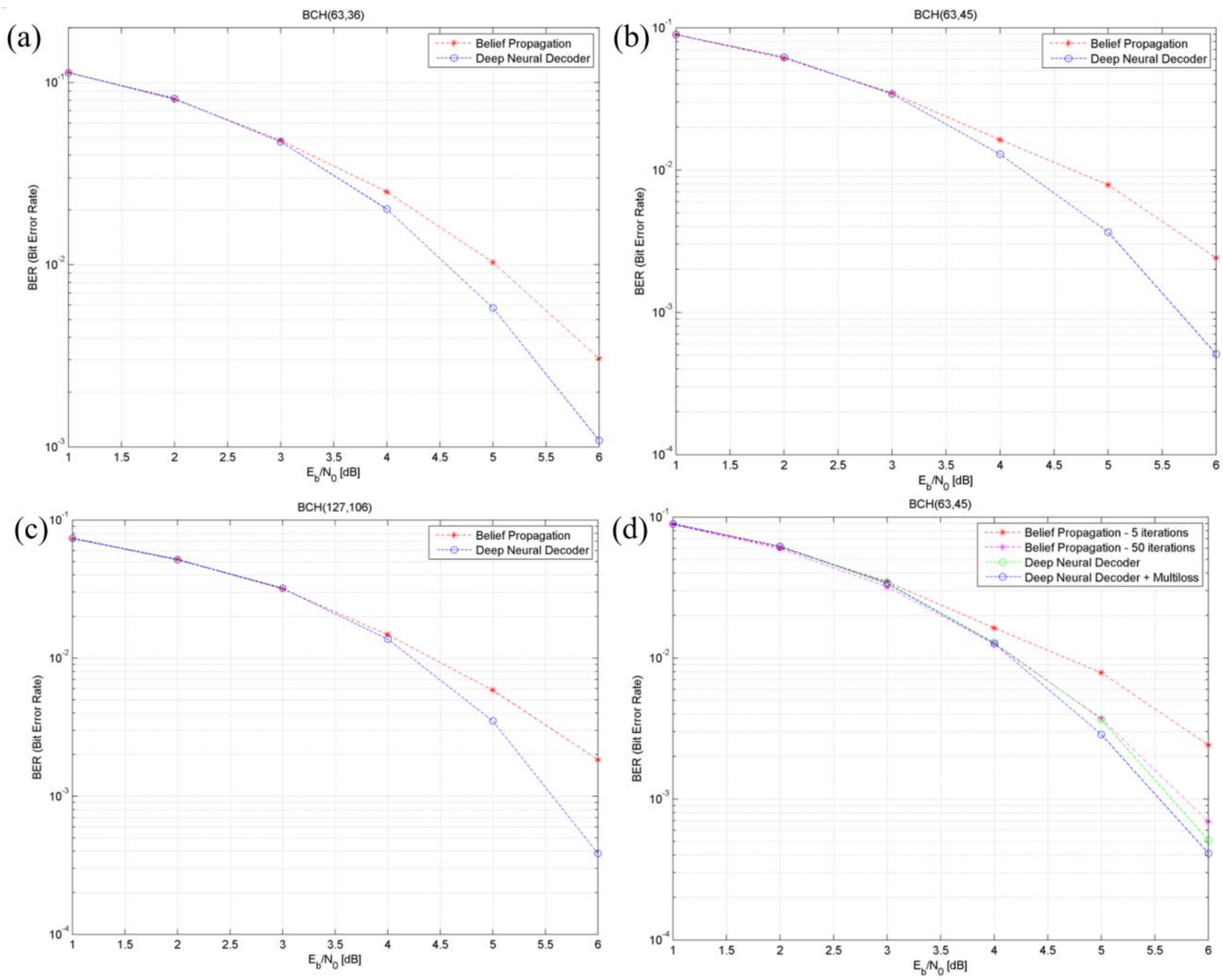

Figure 6. (a) BER results for BCH (63,36) code, (b) BER results for BCH (63,45) code, (c) BER results for BCH (127,106) code (d) BER results for BCH (63,45) code trained with multiloss[15].

In figure 6(a-c), the deep neural decoder has a 0.75 dB improvement compared with the BP algorithm in the high SNR region. Figure 6(d) shows that the deep neural decoder with multiloss has a 0.9 dB improvement compared with the BP algorithm in the high SNR region. And in figure 6, the BER curve of the BP algorithm with 50 iterations is similar to the curve of a deep neural decoder without multiloss. And the author mentioned that the curve of the deep neural decoder without multiloss in figure 6(d) is the result of the deep neural decoder with 5 iterations. The proposed neural decoder can reduce training time tenfold to achieve almost the same performance as the BP algorithm.

5. Node-Classified Redundant Decoding Neural Network (NC-RDNN) Algorithm

5.1. Principles of NC-RD algorithm

NC-RDNN classifies variable nodes according to the shortest circle contained in variable nodes to limit the spread of unreliable information. This neural network can be applied before the neural network decoder mentioned in Section 5 to classify the reliability of variable nodes, thereby further improving performance. NC-RDNN applies deep learning on top of the node-classified redundant decoding (NC-RD) algorithm. Therefore, it is necessary to introduce the NC-RD algorithm first.

The RRD algorithm mentioned in section 1.3 successfully performs the BP decoding algorithm on BCH decoding. On top of the RRD algorithm, Bryan Liu et al. proposed the NC-RD algorithm [13]. In the RRD algorithm, the bits positions are randomly permutated to reduce the growing correlations between the decoding messages. NC-RD algorithm performs bit permutation according to an ordered permutation list. Before doing the RRD algorithm, the variable nodes will be classified by K-means to get an ordered list of permutations through the centroid in the variable node cluster. Then the obtained list is used in the RRD algorithm to limit the transmission of unreliable information in decoding iterations. The process of the NC-RD algorithm could be summarized as follows:

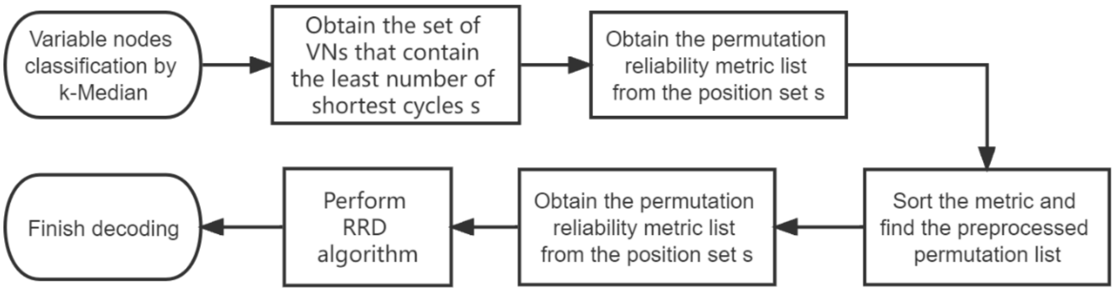

Figure 7. NC-RD algorithm

Based on NC-RD algorithm, Bryan Liu et al. [13] proposed mNC-RD algorithm. In mNC-RD algorithm, generated permutation list is generated multiple times and permutate bits according to the permutation lists in different orders. mNC-RD algorithm performs better, since the bits converge to the true codewords faster.

5.2. NC-RDNN algorithm

Based on NC-RD algorithm, Bryan Liu et al. [13] proposed the NC-RDNN algorithm, which constructs the neural network decoder based on the Tanner graph. Part of the neural network layers is shown in figure 6.

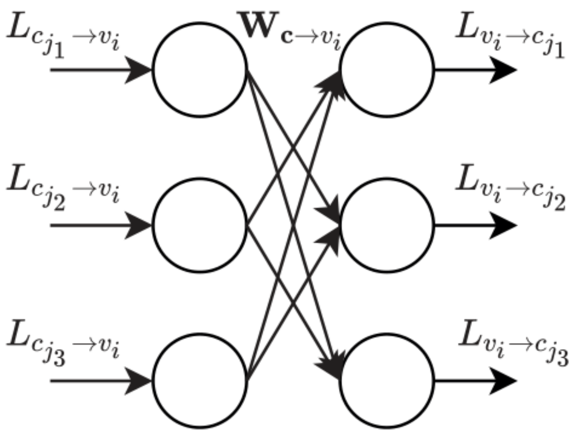

Figure 8. Two layers of NC-RDNN [13]

In figure 6, same as in the RRD algorithm, $L_{c\to v_i}$ and $L_{v_i\to c}$ represent the LLR information transform between check nodes and variable nodes. After introducing the weight variable, $L_{v_i\to c}$ can be represented as $L_{v_i\to c} = w_{c\to v_i}L_{c\to v_i}$, where $w_{c\to v_i} \in F_2^{(|N(v_i)|\times|N(v_i)|)}$ and $F_2 \in \{0,1\}$. $w_{c\to v_i} = 0$ means the neurons do not have connections. The structure of NC-RDNN for BCH (7,4) is shown in figure 7.

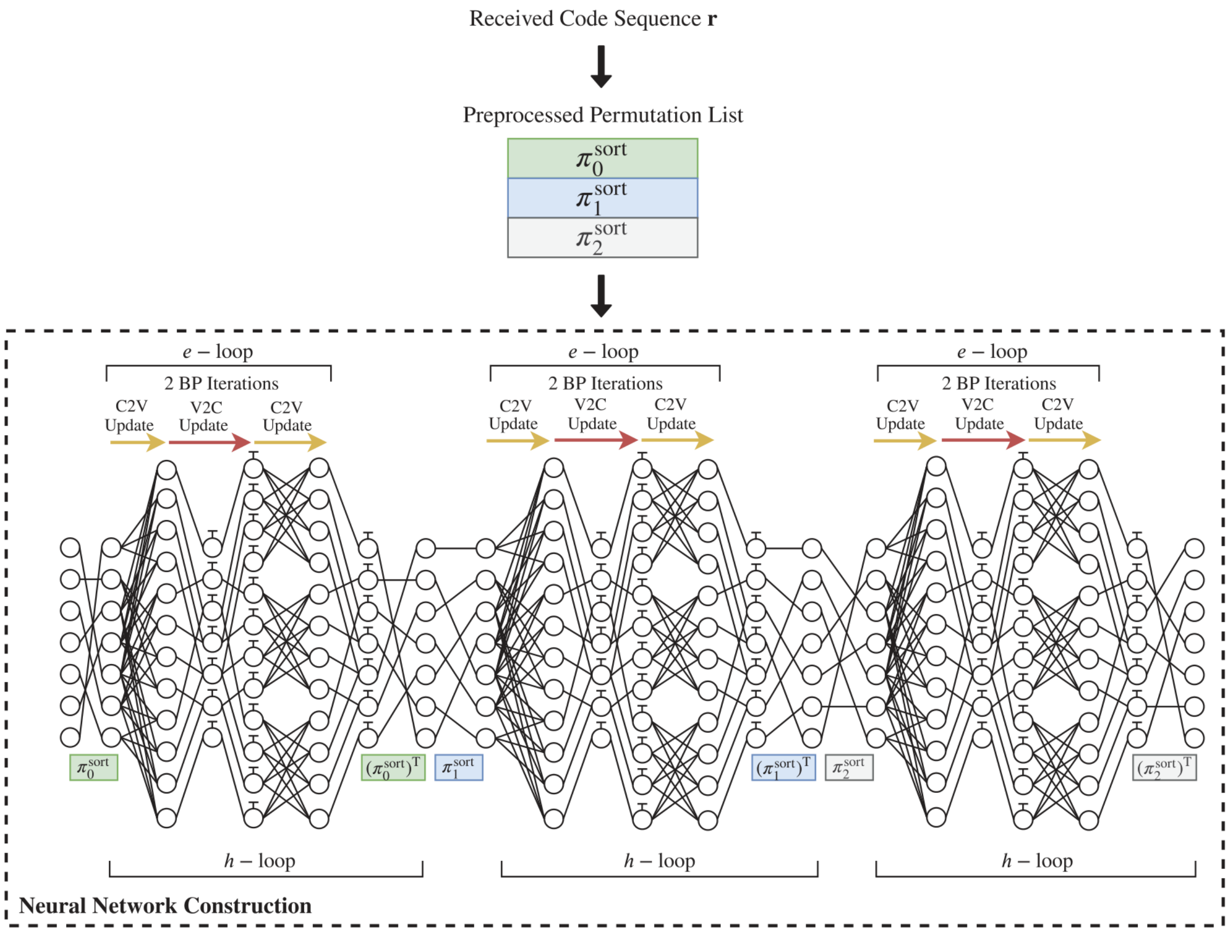

Figure 9. Structure of NC-RDNN for BCH (7,4) [13]

5.3. Performance of NC-RDNN algorithm

Bryan Liu et al. compared the bit error rate of different decoding algorithms as a function of SNR. The signal in the experiment is transmitted in the AWGN channel. The result of the comparison is shown in figure 8:

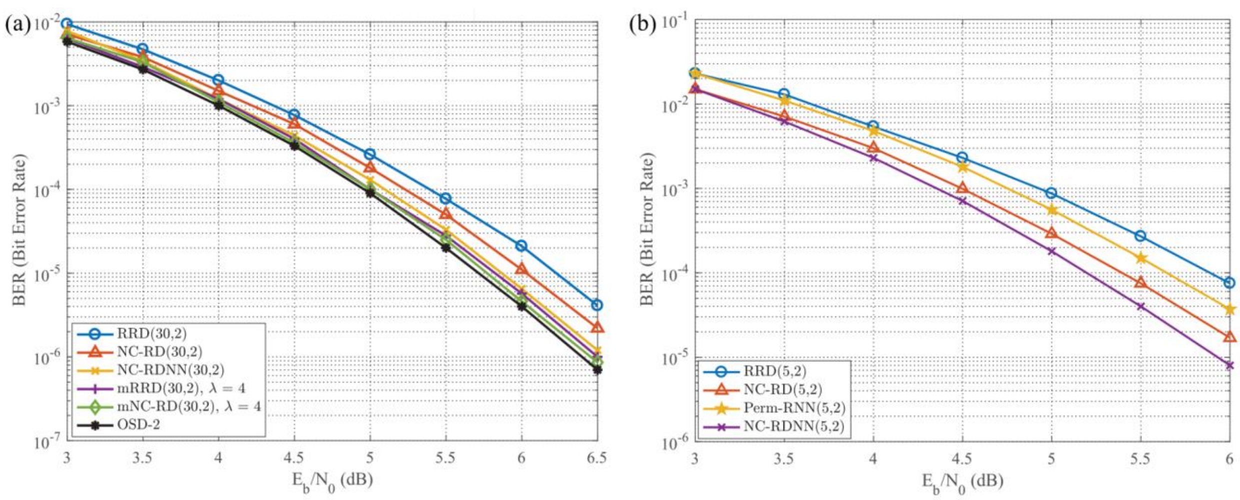

Figure 10. (a) Bit error rate of BCH (31,21). (b) Bit error rate of BCH (63,36) [13].

Figure 8(a) shows that the performance of NC-RD algorithm has 0.2dB gain over that of RRD, and the decoding performance of NC-RDNN has 0.18dB gain compared with NC-RD for decoding BCH (31,21). Figure 8(b) demonstrates that NC-RD is 0.5 dB better than RRD for decoding BCH

(63,36), and NC-RDNN has a further 0.2 dB gain than NC-RD. As can be seen from the figure 8, as the SNR increases, the performance difference between the decoders also increases.

6. Decoder System for Satellite Communication

The satellite-ground integrated system can improve spectrum utilization efficiency and optimize the scarce low-frequency spectrum resources [3]. The disadvantage is that the ground and satellite networks use the same frequency band, which will inevitably cause interference. The primary interference scope in the satellite-ground integrated system is shown in figure 9.

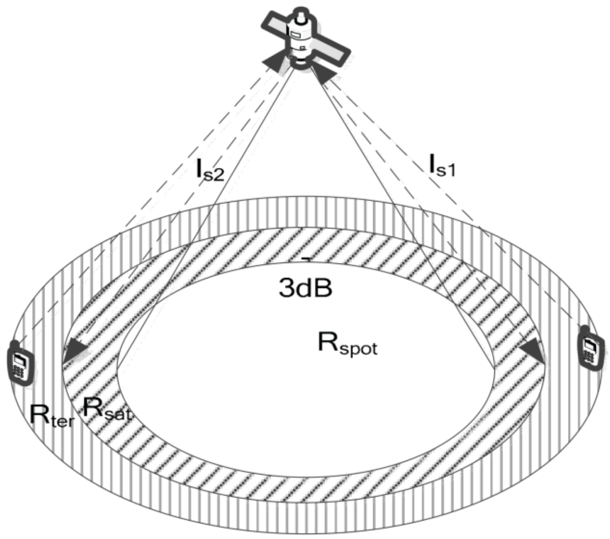

Figure 11. The main interference scope in the system[3]

Due to the high level of integration within the satellite, noise effects exist between different systems. The neural network decoder mentioned in section three is suitable for reducing this interference. After training the neural network decoder to compensate for the non-white noise, the neural network BP decoder is used to decode the messages. And NC-RDNN is used to classify the variable nodes to improve decoding performance further. The decoding system is shown in figure 10.

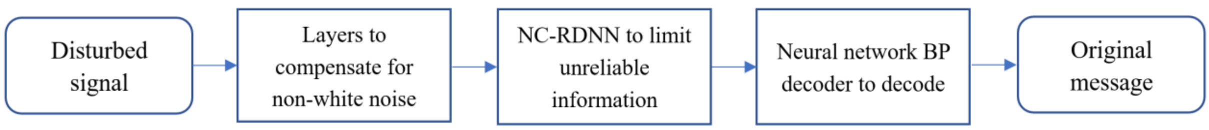

Figure 12. decoder system for satellite communication

7. Conclusion

This paper introduces three neural network decoders for different problems. Then these three kinds of neural networks are combined to propose a neural network decoding system. The algorithm in section 4 employs neural networks to lessen the influence of non-white noise in the system. The algorithm in section 5 uses the neural network to adjust for the shortest cycles in the Tanner graph. The NC-RDNN algorithm mentioned in section 6 establishes the variable nodes classification method and uses neural networks to enhance performance further. As the BCH code length increases, the complexity of sample training and quantization accuracy increases dramatically, which is not conducive to the performance improvement of neural networks. Therefore, classifying BCH codes with different code lengths and selectively using neural network decoders to decode BCH codes with a specific code length range is also a direction worthy of further study.

This system improves the encoder's performance by reducing the impact of non-white noise, reducing the shortest circles in the Tanner graph, and classifying variable nodes. The system consists entirely of neural networks, and the training of the weights can be done on the ground and then uploaded to the satellite decoding system to prevent unnecessary energy waste. Since the system is wholly based on the neural network and can compensate for noise by adjusting the weights, the system is very suitable for satellite systems in complex space environments where various noises exist and need energy saving.

In the high SNR value range, the improvement of using the neural network is larger. Many researchers have proposed optimized decoding methods for RS codes [12,16–18]. Since RS codes have many similarities with BCH codes, the optimized decoding algorithms of RS codes may be implemented into BCH codes in the future.

References

[1] Liang F, Shen C and Wu F 2018 An Iterative BP-CNN Architecture for Channel Decoding IEEE Journal of Selected Topics in Signal Processing 12 144 – 59.

[2] Ortuno I, Ortuno M and Delgado J A 1992 Error correcting neural networks for channels with Gaussian noise [Proceedings 1992] IJCNN International Joint Conference on Neural Networks vol 4 pp 295–300 vol.4.

[3] Gao Y, Ou Q, Zeng Q, Zhang X and Wu J 2013 Analysis of mutual interferences between two components in satellite-terrestrial integrated system 2013 3rd International Conference on Consumer Electronics, Communications and Networks pp 145 – 8.

[4] Shannon C E 1948 A mathematical theory of communication The Bell System Technical Journal 27 379 – 423.

[5] Bose R C and Ray-Chaudhuri D K 1960 On a class of error correcting binary group codes Information and Control 3 68 – 79.

[6] Bose R C and Ray-Chaudhuri D K 1960 Further results on error correcting binary group codes Information and Control 3 279 – 90.

[7] Peterson W 1960 Encoding and error-correction procedures for the Bose-Chaudhuri codes IRE Transactions on Information Theory 6 459 – 70.

[8] Berlekamp E R 1968 Algebraic coding theory (New York: McGraw-Hill).

[9] Tanner R 1981 A recursive approach to low complexity codes IEEE Transactions on Information Theory 27 533 – 47.

[10] Pearl J 2022 Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach Probabilistic and Causal Inference ed H Geffner, R Dechter and J Y Halpern (New York, NY, USA: ACM) pp 129 – 38.

[11] Jiang J and Narayanan K R 2004 Iterative soft decoding of Reed-Solomon codes IEEE Communications Letters 8 244–6.

[12] Halford T R and Chugg K M 2006 Random Redundant Soft-In Soft-Out Decoding of Linear Block Codes 2006 IEEE International Symposium on Information Theory pp 2230 – 4.

[13] Liu B, Xie Y and Yuan J 2020 A Deep Learning Assisted Node-Classified Redundant Decoding Algorithm for BCH Codes IEEE Transactions on Communications 68 5338 – 49.

[14] Halford T R and Chugg K M 2006 An algorithm for counting short cycles in bipartite graphs IEEE Transactions on Information Theory 52 287 – 92.

[15] Nachmani E, Be'ery Y and Burshtein D 2016 Learning to decode linear codes using deep learning 2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton) pp 341 – 6.

[16] Dimnik I and Be'ery Y 2009 Improved random redundant iterative HDPC decoding IEEE Transactions on Communications 57 1982 – 5.

[17] Liu B, Xie Y, Yang L and Yuan J 2018 An Iterative Soft-Decision Decoding Algorithm with Dynamic Saturation for Short Reed-Solomon Codes 2018 IEEE Information Theory Workshop (ITW) pp 1 – 5.

[18] Tang S, Chen L and Ma X 2012 Progressive List-Enlarged Algebraic Soft Decoding of Reed-Solomon Codes IEEE Communications Letters 16 901 – 4.