A New Finite Word-length Optimization Method Design for LDPC Decoder

Jinlei Chen, Yan Zhang and Xu Wang
Key Laboratory of Network Oriented Intelligent Computation
Shenzhen Graduate School, Harbin Institute of Technology
HIT Campus of Shenzhen University Town, Shenzhen, 518055
China
ianzh@foxmail.com http://www.hitsz.edu.cn

Abstract: -A new word-length optimization method based on Monte Carlo simulation is proposed. The word-length of the check node extrinsic message is also further optimized in this paper. In the proposed optimization method, and in the process of optimizing the word-length of the channel data, the statistical distribution results of variable node's posterior probability data and check node's extrinsic message are also obtained. The optimized word-length of variable node's posterior probability data and check node's extrinsic message is concluded by the statistical distribution result and the BER (Bit Error Rate) curves. Compared to the pure Monte Carlo simulation, the proposed method could reduce the amount of simulation work by more than 50%, and have the same word-length optimization results.

Key-Words: - LDPC decoder, word-length optimization, Monte Carlo simulation, min-sum.

1 Introduction

Low density parity check (LDPC) code was first proposed by Gallager in 1960 [1], and it was rediscovered by MacKay and Neal in 1996 [2]. Due to the excellent decoding performance, LDPC code has been widely used in many communication systems, such as wireless local area network (802.11n) [3], digital video broadcasting second generation (DVB-S2) [4] and world interoperability for microwave access (802.16e) [5-6]. Punctured LDPC codes used in coherent optical OFDM systems is also studied in [7].

Belief-propagation (BP), or called sum-product [2], is one of the best LDPC decoding algorithms, but it is not suitable for hardware implementation because of the exponent computation. Min-sum algorithm [8], which decodes LDPC only by comparisons and additions, simplifies the decoding process greatly with acceptable decoding performance loss. Modified min-sum algorithms were proposed in [9-11], normalized factor and offset factor are used to get a better performance in decoding.

Layered decoding method was proposed in [12]. Check node and variable node are both updated simultaneously in this method to reduce the decoding latency half without performance loss. In [13], a fast-convergence algorithm using layered decoding was proposed and about 1/6 iteration numbers decreased for LDPC codes used in DVB-S2.

The finite word-length (or the quantization scheme) of the data deeply affects the decoding performance and the total area of the LDPC decoder in hardware implementation. So the finite word-length optimization method should be able to balance between the decoding performance and the complexity of the LDPC decoder.

Monte Carlo simulation method is a widely used method in finite word-length optimization [14]. In LDPC decoding simulation, random numbers are used as the message bits of the code word in LDPC codes simulation. These random numbers, generated by a pseudo random number algorithm, are independent and equally likely to be 0 or 1. BER (Bit Error Rate) is used to measure the LDPC decoding performance over various finite word-lengths.

The expected BER is always below 10^{-6} in LDPC decoding, so it costs a long time to obtain the finite word-length of all the terms by using Monte Carlo simulation method. In this paper, we proposed a new finite word-length optimization method. In this method, the finite word-length of channel message is obtained by Monte Carlo simulation, and the finite word-length of other messages are obtained by the statistical distribution results in Monte Carlo simulation. The proposed method saves the simulation work more than 50% to obtain the same results compared to the original Monte Carlo simulation method.

The rest of this paper is organized as follows. Section 2 describes the LDPC decoding algorithm, normalized min-sum algorithm and off-set min-sum algorithm. Layered decoding scheme, word-length of fix-point data and Monte Carlo Simulation method are also discussed in this section. Section 3 proposes a new word-length optimization method. Section 4 proposes the method to further optimize the word-length of the check node extrinsic message. Simulations results and discussions are given in section 5. Section 6 concludes the paper.

2 Background

2.1 Min-Sum Algorithm

Min-sum algorithm is one of the most popular approaches of BP [9], and it reduces the hardware complexity greatly. Both normalized min-sum and offset min-sum algorithm are based on the min-sum algorithm. Min-sum algorithm is expressed as follows:

$$r_{ij} = \prod_{i' \in V_{j\backslash i}} a_{ji'} \cdot \min_{i' \in V_{j\backslash i}} |q_{ji'}| \quad (1)$$

$$q_{ji} = LLR_j + \sum_{j' \in C_{i\backslash j}} r_{j'i} \quad (2)$$

$$Q_j = LLR_j + \sum_{j' \in C_i} r_{j'i} \quad (3)$$

$$LLR_j = \log \frac{P(x=0\,|\,y)}{P(x=1\,|\,y)} \quad (4)$$

In (1), r_{ij} is the check-to-variable message passed from check node i to variable node j, $q_{ji'}$ is the variable-to-check message passed from variable node j to check node i', $a_{ji'}$ is the sign of $q_{ji'}$, $V_{j\backslash i}$ is the set of the variable nodes which connect to check node j without node i, in (2), $C_{i\backslash j}$ is the set of the check nodes which connect to variable node i without check node j, in (3), Q_j is the log likelihood ratio (LLR) for variable node j, C_i is the set of all the check nodes which connect to variable node i, in (4), x is the transmitted bit and y is the message received from channel.

At the beginning of decoding, all variable node messages q_{ji} are installed by (4). In each iteration, (1), (2) and (3) are processed serially, and a guess of the codeword is obtained by the sign of Q_j (0 for $Q_j > 0$, 1 for $Q_j < 0$) in (3), if the codeword fits all the parity check or the iteration exceeds the predefined maximum iteration time, the decoding stops.

2.2 Modified min-sum Decoding Algorithm

Although it is easy to implement the min-sum algorithm, it results in degradation in decoding performance. Both normalized min-sum and offset min-sum are modified versions of the min-sum algorithm, the first one with a normalized factor and the second with an additive correction factor, and these algorithms achieve almost the same performance as that of the BP algorithm.

In the normalized min-sum algorithm, check node updating operation uses normalization constant λ smaller than 1, and (1) is changed to:

$$r_{ij} = \prod_{i' \in V_{j\backslash i}} a_{ji'} \cdot \lambda \cdot \min_{i' \in V_{j\backslash i}} |q_{ji'}| \quad (5)$$

In offset min-sum algorithm, the check node updating operation is given as follows:

$$r_{ij} = \prod_{i' \in V_{j\backslash i}} a_{ji'} \cdot \max\left\{ \min_{i' \in V_{j\backslash i}} |q_{ji'}| - \varepsilon, 0 \right\} \quad (6)$$

In (5) and (6), both correction factors are used to decrease the magnitude of r_{ij}, and their performance is analyzed in [12]. Compared to offset min-sum, normalized min-sum algorithm has better decoding performance but the multiplication increases the implementation complexity. In this paper, normalized min-sum algorithm is used in the proposed LDPC decoder.

2.3 Layered Decoding Scheme

In layered decoding scheme, the parity check matrix can be viewed as horizontal layers, and each layer can represent a component code [13]. The code is composed of all layers and their intersections. In QC-LDPC code, the rows and columns are naturally blocked by the permutation matrix, so each block row can be indicated as a layer. As each layer starts decoding, the inputs of variable node contain channel inputs and the extrinsic messages of the check nodes from the previous layers. Iterations within a layer are called sub-iterations and the overall process is labelled as super-iterations.

In each sub-iteration, the variable-to-check message q_{ji}' is firstly computed by:

$$q_{ji}' = Q_j' - r_{ij}' \quad (7)$$

Where r_{ij}' is the check-to-variable message in the previous super-iteration of this layer, and Q_j' is the LLR result of variable node j from the previous sub iteration. The r_{ij}' is computed by (1) or (5) or (6), and Q_j is computed by:

$$Q_j = q_{ji}' + r_{ij} \quad (8)$$

When super-iteration is finished, The codeword is obtained by the sign of Q_j.

In general, the decoding convergence speed of the layered decoding scheme is two times faster than that of the two-phase scheme, and in layered decoding, only Q_j and r_{ij} are stored in memories because the variable-to-check message q_{ji} are computed by (7).

2.4 Word-length of Fix-point Data

The difference of fix-point data and floating-point data is that for fix-point data, the position of the decimal point is fixed. The word-length of integer part and the fraction part are constant. For fix-point data, we have:

$$WL = IWL + FWL$$

WL is the word-length of the fix-point data, IWL is the word-length of the integer part, FWL is the word-length of the fraction part.

A signed number is always expressed by two's complement form, in this case, IWL contains a sign bit: 0 for positive number, 1 for negative number, and in this case:

$$WL = IWL + FWL + 1$$

So the range of the signed fix-point data is $[-2^{IWL}, 2^{IWL} - 2^{-FWL}]$, the precision of the data is 2^{-FWL}.

In LDPC decoder hardware implementation, we use the notation $(WL : FWL)$ to represent a quantization scheme. As introduced previously, WL bits are used for total bit size and FWL bits are used for fractional values.

2.5 Monte Carlo Simulation Method

The quantization scheme of LLR_j significantly affects the decoding performance and the total decoder complexity, and the word-length of other terms in the decoding algorithm are also depended on it, so the quantization scheme of LLR_j should be determined firstly. Though large word-length has good decoding performance, it causes hardware overhead for the buffers and a large number of hardware for the iterative decoding computation. A small word length may result in very poor performance. Hence, the quantization scheme should balance the decoding performance and the hardware complexity.

There are two steps in Monte Carlo simulation method. First, the quantization scheme of channel message is obtained. Second, in two-phase decoding scheme, the finite word-length of extrinsic message of check node and variable node are obtained. In layered decoding scheme, the finite word-lengths of variable node message and the extrinsic message of check node are obtained sequentially. In [15-17], the word-lengths of variable node message and the extrinsic message of check node are equal.

3 An Improved Word-length Optimization Method

In LDPC decoding, the expected BER is always below 10^{-6}, and that means 10^8 message bits should be transmitted in Monte Carlo simulation. It will cost several days to obtain the quantization scheme of all the terms using Monte Carlo method. In the proposed method, days of time will be saved because of more than half Monte Carlo simulations are omitted.

The proposed finite word-length optimization method is expressed as follows:

Step 1, the quantization scheme of channel message is obtained using Monte Carlo simulation. Meanwhile, the statistical distribution result of variable node's posterior probability message Q_j is also obtained.

Step 2, by analyzing the statistical distribution result, the quantization scheme of Q_j is achieved.

At last, the word-length of r_{ij} equals to the word-length of Q_j.

This method could be used in any min-sum based decoding algorithm.

4 Further Word-length Optimization of the Check Node Extrinsic Message

In layered decoding scheme, only r_{ij} is stored in the extrinsic memory. So the word-length of r_{ij} deeply affects the area of the decoder. In [15-17], the word-lengths of r_{ij} and Q_j are equal. But in min-sum based LDPC decoding algorithm, as function (1) shows, the check node extrinsic message r_{ij} is the minimum value of the input messages $q_{ji'}$. So there is a chance that the word-length of r_{ij} could be further optimized. In this section, we propose methods to further optimize the word-length of r_{ij} both in Monte Carlo method and the proposed method.

4.1 In Monte Carlo Method

Let LLR_j and Q_j use the quantization scheme obtained in Monte Carlo simulation method as introduced in section 2.5. The appropriate word-length of r_{ij} is obtained by analyzing BER curves of r_{ij} with various word-length.

More Monte Carlo simulations are needed in the word-length optimization process of r_{ij}, and that means more time is needed in Monte Carlo simulation method to obtain the quantization scheme of all the terms.

4.2 In the Improved Optimization Method

In the improved method, the process of optimizing the word-length of r_{ij} is similar as the process of Q_j. The statistical distribution result of r_{ij} is obtained in step 1, and the result is used to choose the appropriate word-length of r_{ij}.

In the proposed method, the process of optimizing the word-length of r_{ij} doesn't bring in any extra Monte Carlo simulation, so the time of optimizing the word-length of all the terms is nearly the same as in section 3.

5 Simulations and Discussions

5.1 Monte Carlo Method

In this section, we use layered offset min-sum algorithm as the decoding algorithm, all the BER curves are obtained by transmitted 100,000 code words, the max iteration number is 50, the modulation method is BPSK and the channel module is AWGN. BER results over Monte Carlo Simulation are used to compare the different performance of various quantization scheme of LLR_j, Q_j and r_{ij}.

The performances of the (1944, 972) LDPC code in IEEE 802.11n with floating point, (7:4), (6:3) and (5:2) quantization schemes of LLR_j are shown in Fig. 1. It shows that (7:4) quantization scheme has the best performance of the three fix-point quantization schemes, and the difference of decoding performance between (7:4) and (6:3) quantization scheme is less than 0.05dB. The (5:2) quantization scheme has the worst decoding performance. Thus it turns out that using the (6:3) scheme of LLR_j seems to be the optimal tradeoff between hardware complexity and decoding performance. This quantization scheme has a precision of $2^{-3} = 0.125$ with a maximum value of $2^2 - 2^{-3} = 3.875$ and a minimum value of $-2^2 = -4$.

Let LLR_j use the (6:3) quantization scheme, the decoding performances of the (1944, 972) LDPC code in IEEE 802.11n with various word-length of Q_j (which is denote by W_Q) are shown in Fig. 2. As shown in Fig. 2, $W_Q = 9$ has nearly the same performance as $W_Q = \infty$, and when $W_Q = 8$, the BER curve has an error floor at 10^{-5}. So at last, we choose $W_Q = 9$. The precision of Q_j is the same as that of LLR_j.

Let LLR_j use (6:3) quantization scheme, Q_j uses (9:3) quantization scheme, the decoding performances of the (1944, 972) LDPC code in IEEE 802.11n with various word-length of r_{ij} are shown in Fig. 3. When $W_r \geq 7$, the BER curves is coincide with the curve of $W_r = \infty$, and there is only little difference between $W_Q = 6$ and $W_r \geq 7$. For example, when SNR=1.9dB, BER of $W_r \geq 7$ is 3.0×10^{-6}, and BER of $W_r = 6$ is 3.44×10^{-6}. But the difference between $W_r = 5$ and $W_r \geq 6$ is quite significant. So at last, we choose $W_r = 6$. And the precision of r_{ij} is also the same as that of LLR_j.

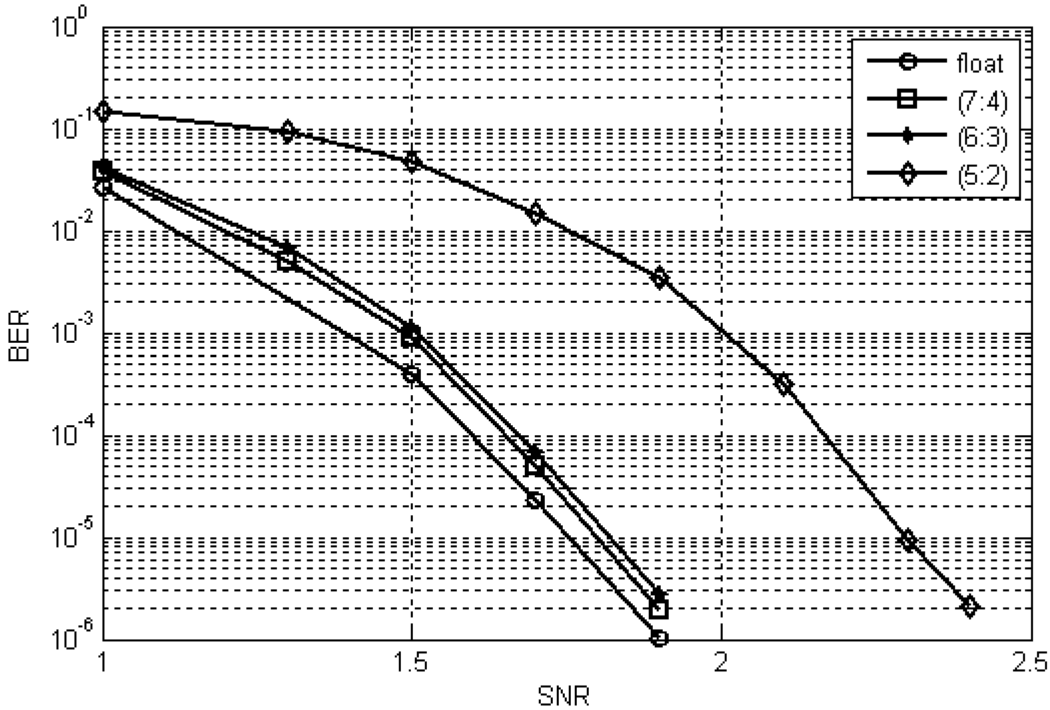

Figure 1. BER Curves of LLR_j under Layered Offset Min-Sum Algorithm with Different Quantization scheme.

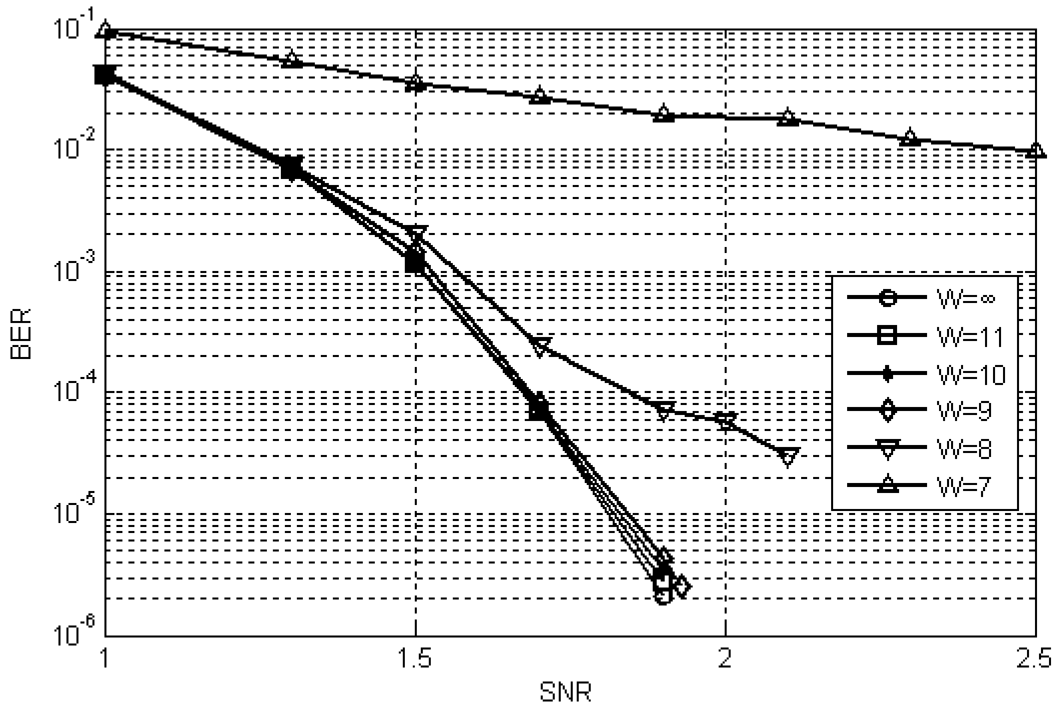

Figure 2. BER Curves of Q_j under Layered Offset Min-Sum Algorithm with Different Quantization

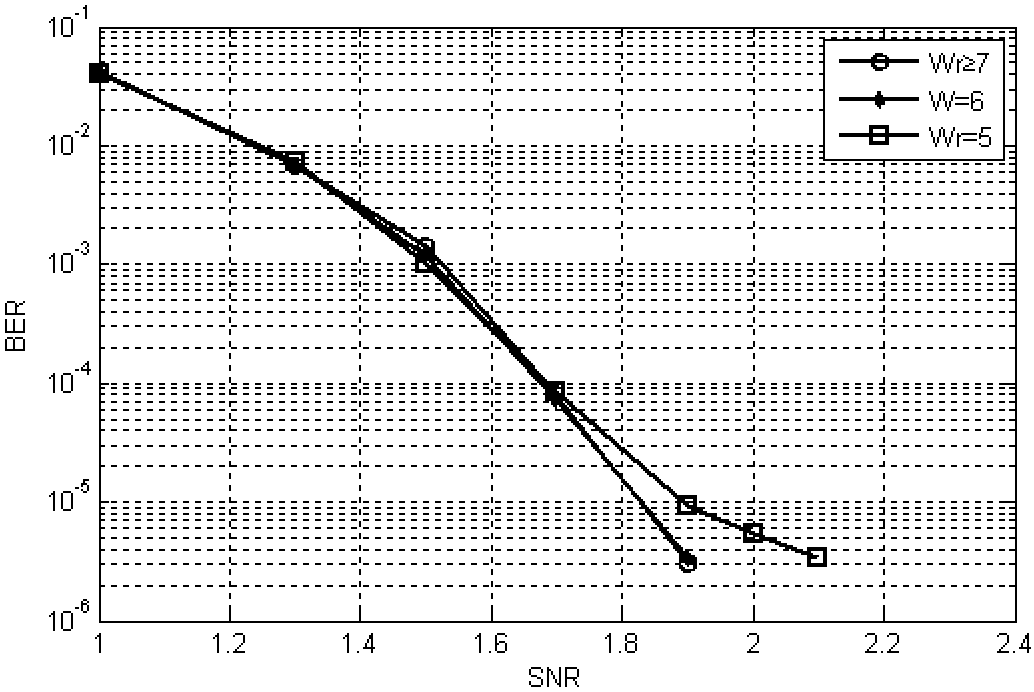

Figure 3. BER Curves of r_{ij} under Layered Offset Min-Sum Algorithm with Different Quantization

Table 1: Quantization Scheme of Offset Min-Sum Decoding Algorithm

Message	Quantization scheme	Range	Precision
LLR_j	(6:3)	(-4, 3.875)	0.125
Q_j	(9:3)	(-32, 31.875)	0.125
r_{ij}	(6:3)	(-4, 3.875)	0.125

The quantization scheme of all the terms in offset min-sum algorithm is summarized in table 1. In [15-17], the word-length of r_{ij} would be the same as Q_j, so in the proposed method, the word-length is further optimized by 1/3.

5.2 The Improved Optimization Method

We also use the example in 5.1. The statistical distribution result of Q_j and r_{ij} in the simulation of Fig. 1 is shown in Fig. 4 and Fig. 5. The assumed SNR of AWGN channel is 1.9dB, and in this case, the average iteration is 6.08. In Fig. 4 and Fig. 5, the x-axis data is quantized. For example, the number ‘5’ in x-axis equals to 0.625.

As shown in Fig. 4, the distribution range of Q_j increases as the iterations increase, and most values of Q_j is in the range of (-256, 256), so the quantization scheme of Q_j is (9:3).

From Fig. 4, we can see that the area of the curves decrease as the iterations increase. It means the amount of Q_j decreases with the iterations, and that because in each iteration, many code words are decoded, and the number of un-decoded codes decreases with the increasing of iterations.

From Fig. 4, we can also see that the statistical distribution curve of the channel input data LLR_j is the superposition of two Gauss curves, and the symmetry axis of the two curves are $x=-8$ and $x=8$. That because in the simulation, the modulation system is BPSK, and in BPSK, bit ‘0’ is changed to ‘-1’, and bit ‘1’ is unchanged, and the channel module in the simulation is AWGN, the data received at each time is equal to the sent data plus Gaussian noise, so the statistical distribution curve of LLR_j is the superposition of two Gauss

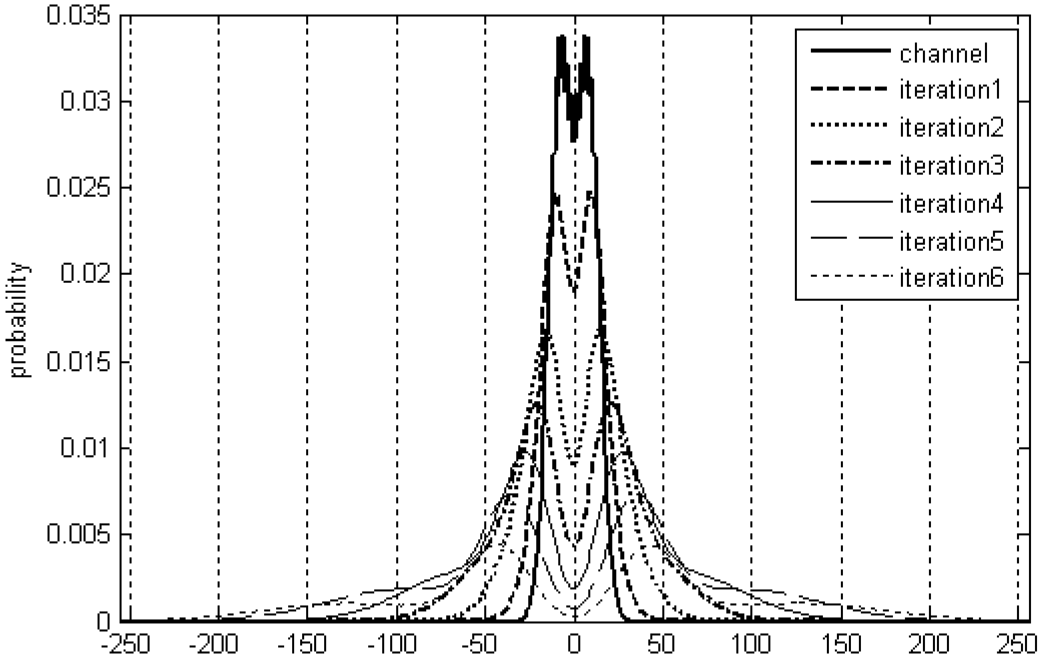

Figure 4 Statistical Distribution of Q_j

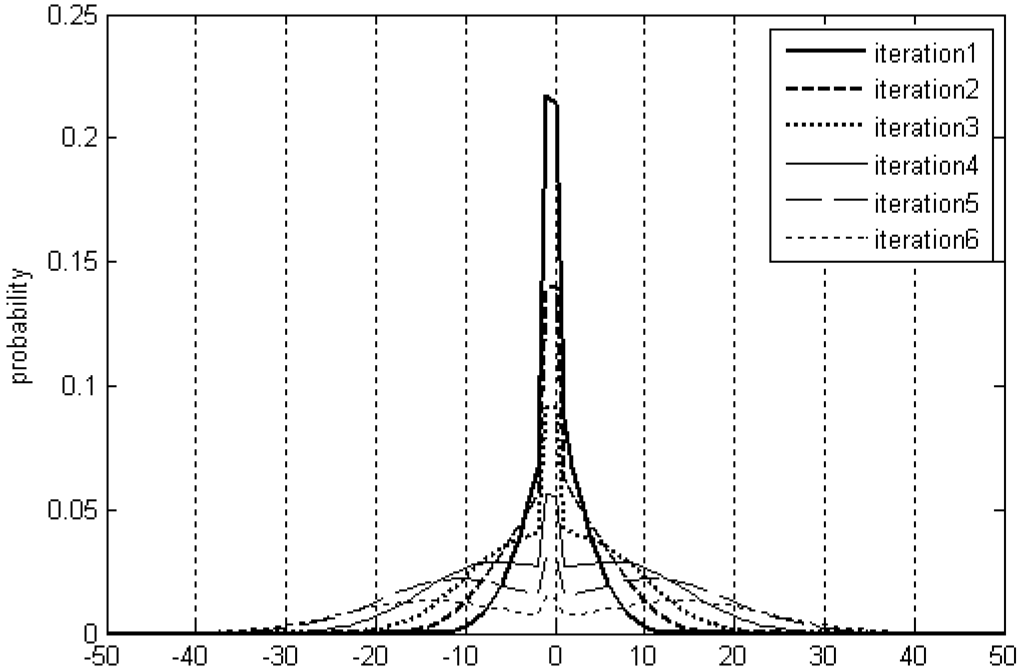

Figure 5 Statistical Distribution of r_{ij}

Figure 6. BER Curves of Q_j under Layered Offset Min-Sum Algorithm with Different Quantization

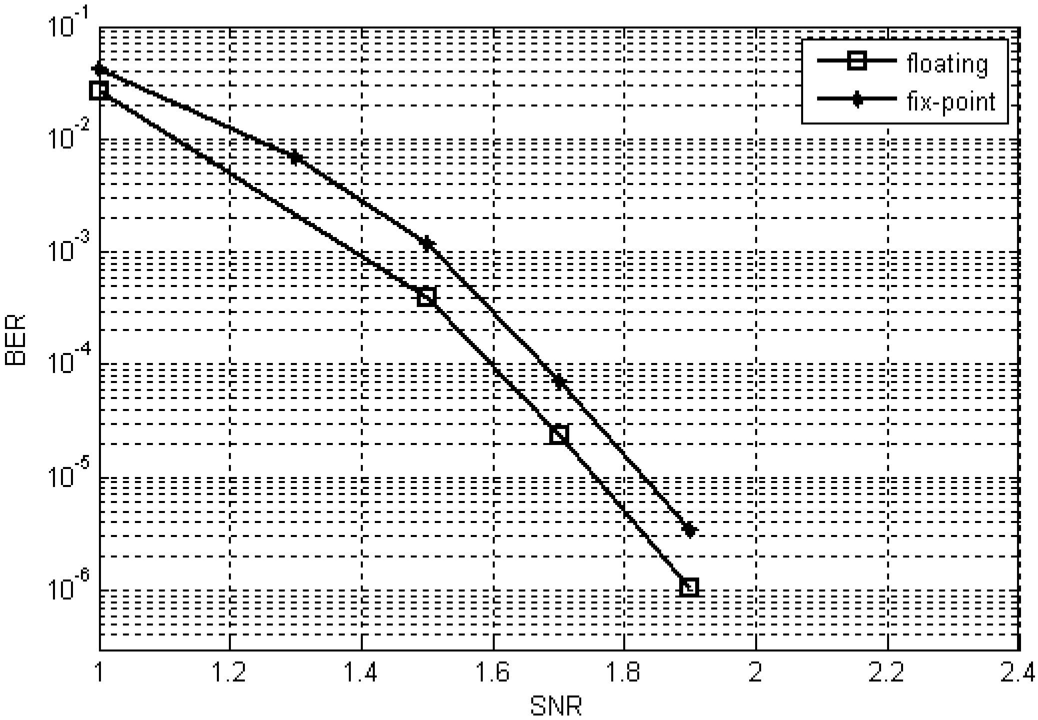

Figure 6. BER Curves of Q_j under Layered Offset Min-Sum Algorithm with Different Quantization

curves. The quantization scheme of LLR_j is (6:3), so the symmetry axis of the two curves are $x=-8$ and $x=8$.

As shown in Fig. 5, the distribution range of r_{ij} increases with the iterations, and most values of r_{ij} is in the range of (-32, 32), so the quantization scheme of r_{ij} is (6:3).

In Fig. 5, there is a wave crest around 0 in the statistical distribution curve of the first iteration, and the crest disappears as the iterations increase. That because in the first few iterations, the signs of many Q_j are uncertain or the reliabilities of Q_j are small, and from function (1), we can conclude that in the first few iterations, most values of r_{ij} are around 0. When the number of iterations increases, more codeword is decoded, and for the un-decoded codeword, the reliability of Q_j and the range of r_{ij} would increases. So the wave crest around 0 disappears with iterations.

Fig. 6 shows the decoding performance of the (1944, 972) LDPC code in IEEE 802.11n with floating-point and the final quantization scheme. It is shown that the decoding performance loss using the proposed quantization scheme compared with floating point is less than 0.1dB.

In this example, by using Monte Carlo method, 12 BER curves are needed to obtain the final quantization scheme. In the improved finite word-length optimization method, only 3 BER curves and 2 statistical distribution curves are needed. So in this example, the improved finite word-length optimization method reduces the simulation work by 75%.

Generally speaking, in Monte Carlo simulation method, for layered decoding scheme, at least 9 BER curves are needed to obtain the final quantization scheme (3 curves for LLR_j, 3 curves for Q_j and 3 curves for r_{ij}), for two-phase decoding scheme, at least 6 BER curves are needed to obtain the final quantization scheme(3 curves for LLR_j and 3 curves for the extrinsic messages of variable nodes). In the proposed finite word-length optimization method, only 3 BER curves are needed to obtain the final quantization scheme for both layered decoding scheme and two-phase decoding scheme. So the proposed method can reduce the simulation work by more than 50%

6 Conclusion

In this paper, we proposed a new word-length optimization method and further optimized the word-length of the check node extrinsic message. In the proposed method, the word-length of variable node's posterior probability data and check node's extrinsic message is concluded by the statistical distribution result and the BER (Bit Error Rate) curves. Compared to the pure Monte Carlo simulation, the proposed method could reduce the amount of simulation work by at least 50%, and has the same results.

References:

[1] R.G.Gallager, Low-density parity-check codes, *IEEE Transactions on Information Theory*, Jan.1962, pp.21-28.

[2] D. J. C. MacKay and R. M. Neal, Near Shannon limit performance of low-density parity-check codes, *Electronic Letter*, vol. 32, Aug 1996, pp. 1645-1646

[3] IEEE Std. 802 11n-2009, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Enhancements for Higher Throughput, *IEEE P802.11n*, Sep. 2009.

[4] Digital video broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broad-band satellite applications, *ETSI*, EN 302 307 V1. 1.1, 2005.

[5] IEEE Std. 802 16e, Air Interface for Fixed and Mobile Broadband Wireless Access Systems, *IEEE P802.16e*, Feb. 2006.

[6] M. Khan, R. Carrasco and I. Wassell, application of high rate LDPC codes for IEEE 802.16d or WiMAX (fixed broadband) systems, *WSEAS Transactions on Communications*, vol. 6, iss. 1, Jan. 2007, pp. 92-97.

[7] J. Chang, K. Takeshi, F. Hisato and H. Kazuhisa, Influence of laser linewidth and modulation level for coherent optical OFDM with punctured LDPC codes, *WSEAS Transactions on Communications*, vol. 10, iss. 6, pp. 175-181.

[8] J. Chen and M. P. C. Fossorier, Near optimum universal belief propagation based decoding of low-density parity check codes, *IEEE Transactions on Communications*, vol. 50, Mar. 2002, pp. 406-414.

[9] J. Chen and M. P. C. Fossorier, Decoding low-density parity-check codes with normalized APP-based algorithm, *Proceedings of IEEE Globecom, San Antonio*, TX, Nov. 2001, pp. 1026–1030.

[10] E. Eleftheriou, T. Mittelholzer, A. Dholakia, Reduced-complexity decoding algorithm for low-density parity-check codes, *IEEE Electronic Letters*, vol. 37, Jan. 2001, pp. 102–104.

[11] M.P.C. Fossorier, M. Mihaljevic, H. Imai, Reduced complexity iterative decoding of low density parity check codes based on belief propagation, *IEEE Transactions on Communications*, vol. 47, no. 5, May 1999, pp. 673–680.

[12] D.E. Hocevar, A Reduced Complexity Decoder Architecture via Layered Decoding of LDPC Codes, *IEEE workshop on Signal Processing Systems*, Oct. 2004, pp: 107-112.

[13] J. Xie, L. Yin, N. Ge and J. Lu, Fast Convergence Algorithm for Decoding of Low Density Parity Check Codes, *WSEAS Transactions on Communications*, volume.8, issue7, July 2009, pp. 598-607.

[14] N. Metropolis, S. Ulam, The Monte Carlo Method, *Journal of the American Statistical Association*, vol. 44, no. 247, pp.335-341.

[15] V. K. K. Srinivasan, C. K. Singh, P. T. Balsara, A Generic Scalable Architecture for Min-Sum/Offset-Min-Sum Unit for Irregular/Regular LDPC Decoder, *IEEE Transactions on VLSI*, vol. 18, no. 9, Aug., 2010, pp. 1372-1376.

[16] T. C. Kuo, A. N. Willson, A flexible Decoder IC for WiMAX QC-LDPC Codes, *IEEE International Conference on Custom Integrated Circuits*, Sept. 2008, pp.527-530.

[17] C. H. Liu, S. W. Yen, C. L. Chen and H. C. Chang, An LDPC Decoder Chip Based on Self-Routing Network for IEEE 802.16e Applications, IEEE Journal of Solid-State Circuits, vol. 43, n. 3, Mar. 2008, pp.684-694.

Biographies

Jinlei Chen born in 1982. He received the B.S. degree from Jilin University in microelectronics in 2005 and M.S. degree in microelectronics from the Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China, in 2007. Since 2008, he has been a PhD candidate in microelectronics. His main research interests include LDPC codes and VLSI hardware design.

Yan Zhang born in 1969. He has been professor of the Shenzhen Graduate School, Harbin Institute of Technology since 2002. His main research interests are application specific instruction set processor design, including medical image processing chips and wireless communication baseband chip.

Wang Xu born in 1980. Received the M.A's. degrees in microelectronics from the Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen, China, in 2007. Since 2008, he has been a PhD candidate in microelectronics. His main research interests include image processing and embedded DSP processor design.