2012 2nd International Conference on Information Communication and Management (ICICM 2012)
IPCSIT vol. 55 (2012) © (2012) IACSIT Press, Singapore
DOI: 10.7763/IPCSIT.2012.V55.14

Layered Decoding With A Early Stopping Criterion For LDPC Codes

Xiongfei Tao [1 +], Yan Zhang [1], Deyu Feng [1] and Pan Liu [1]

[1] School of Optical and Electronic Information, Huazhong University of Science and Technology, Wuhan, Hubei, China

Abstract: This paper describes a Low Density Parity Check (LDPC) decoder which is based on the improved Layered belief propagation (LBP) algorithm. Including an early stopping criterion, the improved LBP algorithm not only has a high decoding speed, and a significant reduction on the memory required, but also greatly reduces the number of iterations in the low SNR. The decoder, which has a serial architecture, is a peripheral connected to the Nios II CPU of an Altera Cyclone III FPGA.

Keywords: LDPC codes, LBP, early stopping criterion, FPGA

1. Introduction

The introduction of Turbo-codes in the early 90's and the more generally iterative principle has deeply modified the methods for the design of communication systems. This breakthrough has also resurrected the Low Density Parity Check (LDPC) codes invented by R. Gallager in 1963 [1]. LDPC codes have simple iterative decoding algorithm and excellent performance which is close to the Shannon limit.

There are kinds of decoding methods for the LDPC codes. In this paper we apply layered belief propagation decoding (LBP). Compared with the traditional BP algorithm, the LBP algorithm has a significant reduction in the number of memory bits and memory instances required, in the range of 45-50%. And the LBP algorithm is decoding faster which means the decoder logic can also be reduced by nearly 50% to achieve the same throughput and error performance [2].

The common mehod is to verify if the computed codeword satisfies all parity check constraints, at the end of each iteration. Once convergence has been verfied, the decoding process is terminated. However at low SNR ranges it happens frequently that a block can not converge to a valid codeword even after a maximum number of decoding iterations. Thus the early termination of this block can avoid unnecessary iterations, and reduce power consumption and decoding latency [3]. Early termination of the iterative methods can be divided into two categories. One is monitoring the convergence of mean magnitude of variable nodes or check nodes [4]; the other is observing the numbers of satisfied parity check constraints or unsatisfied ones [3, 5].

There are many hardware LDPC decoders designed by using FPGA or VLSI [6-9], where [8, 9] use the LBP decoding algorithm and min-sum approximation. In this paper, we will propose a division method for sub-matrix of the LBP algorithm, and add the early termination of the iterative method into the LBP algorithm, and then use the FPGA to implement the improved LBP algorithm.

This paper is organized as follows. Section 2 presents LBP decoding algorithm and early stopping criterion and gives details about the improved LBP decoding algorithm. Section 3 the architecture model and how it has been implemented in FPGA. Section 4 gives some performance results, and a conclusion is given in Section 5.

+ Corresponding author. Tel.: +8613808698015.
E-mail address: taoxiongfei@mail.hust.edu.cn.

2. Proposed LBP Decoding

2.1. LBP and division method for check matrix

In the LBP decoding algorithm, the node data can be updated in time for the next level sub-matrix of the iterative computation, so that it effectively speed up the convergence of the LDPC decoding and thereby reducing the maximum number of decoding iterations required. At present, two kinds of layered decoding approaches, that is, row-layered decoding and column-layered decoding have been proposed. In this paper, we use the row-layered decoding method [2, 10].

The rows of the parity check matrix are grouped into non-overlapping sub-matrices in the LBP algorithm. Each column in this sub-matrix has at most weight of one. Each sub-matrix is processed as a unit, one sub-matrix after another. In [10], it proposed a method for the H division, which considered check node degree, variable node degree, and the edge connection between the each node. But this method had little effect on the bit error rate (BER) performance. This paper presents a simplified division method, that is each row is a sub-matrix. After completing the all rows computation, do the judgment and verification. The method simplifies the division, and retains the advantages of the LBP algorithm.

An LDPC code is defined by a $M \times N$ parity check matrix H, which encapsulates important matrix parameters: The number of rows, M, is the number of check nodes; the number of columns, N, is the number of variable nodes. Let R_{mj} denote the check node LLR (Log-likelihood ratios) messages sent from the check node m to the variable node j. Let $L(q_{mj})$ denote the variable node LLR messages sent from the variable node j to the check node m. Let $L(q_j)$ represent the APP (a posteriori probability) messages for all the variable nodes which are initialized with the channel messages (assuming BPSK on AWGN channel) for each code bit j by $2r_j/\sigma^2$, where σ^2 is the noise variance and r_j is the received value. S_{check} denotes the numbers of unsatisfied parity check constraints, L denotes the maximum iterations.

Repeat the following for each sub-matrix of rows and for each bit node j:

$$L\left(q_{mj}\right) = L\left(q_j\right) - R_{mj} \tag{1}$$

$$R_{mj} = \prod_{\substack{n \in N(m) \\ n \neq j}} sign\left(L\left(q_{mn}\right)\right) \times f\left(\sum_{\substack{n \in N(m) \\ n \neq j}} f\left(\left|L\left(q_{mn}\right)\right|\right)\right) \tag{2}$$

$$L\left(q_j\right) = L\left(q_{mj}\right) + R_{mj} \tag{3}$$

where $N(m)$ is the set of all variable nodes from parity-check equation m, and

$$f(x) = -\ln\left[\tanh\left(\frac{x}{2}\right)\right] = \ln\frac{e^x + 1}{e^x - 1} \tag{4}$$

After complete l-th iteration:

$$\hat{c}_j = \begin{cases} 0, L\left(q_j\right) \geq 0 \\ 1, L\left(q_j\right) < 0 \end{cases} \tag{5}$$

$$S_{check} = \sum_{i=1}^{M} \bigoplus_{n \in N(m)} \hat{c}_n \tag{6}$$

If $S_{check} = 0$, then decoding is successful and terminate the decoding. Or, if $l < L$, then $l = 1 + 1$; if $l = L$, then reach the maximum number of iterations and terminate the decoding.

2.2. Early stopping criterion

Early termination of the iterative method used in this paper is to observe the numbers of satisfied parity check constraints or unsatisfied ones [3, 5]. Let S_{check} denotes the numbers of unsatisfied parity check

constraints. The behaviors of the LBP decoding can be categorized into three different types in terms of the S_{check} value with different SNR: convergence, stuck and oscillation. We demonstrate the S_{check} value during decoding a (894,447) LDPC code in the Fig. 1.

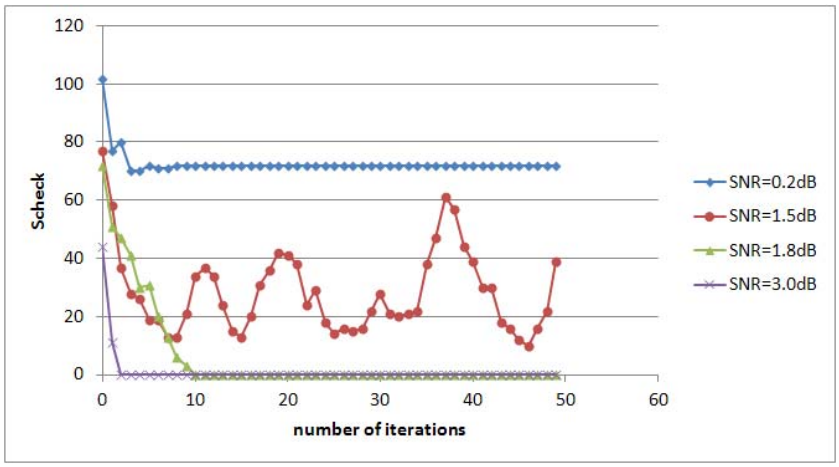

Fig. 1: S_{check} value during decoding a (894,447) LDPC code.

The early stopping criterion proposed in this paper is to record the value of S_{check} in the iteration process. When the S_{check} value didn't reduce at X-th iteration, terminate the iteration. This criterion is mainly used for the types of stuck and oscillation.

The improved LBP algorithm in the paper takes the early stopping criterion into account. Including the early stopping criterion, the improved LBP algorithm not only has a high decoding speed, and a significant reduction on the memory required, but also greatly reduces the number of iterations in the low SNR.

3. LBP decoder structure

There are many decoder structures based on BP algorithm, and [2, 6, 7] give the corresponding decoder structure. According to these decoder structures, this paper designs a decoder structure which is suitable for the improved LBP decoding algorithm.

The design is a SOPC design that takes advantage of the Nios II CPU embedded in the Cyclone III FPGAs. This methodology provides us more flexibility since the additional functionality can be added to the system simply by writing C code for the Nios II CPU. The overall system structure is shown in Fig. 2. The Nios II CPU and other peripheral controllers are connected to the Avalone-MM bus. The communication between the FPGA and the host PC is done through a JTAG UART interface. The CPU is used to generate the additive white Gaussian noise(AWGN), initialize $L(q_j)$, syndrome check and early stopping criterion.

The main process in LDPC Decoder is shown in Fig. 3, in which sub-matrix operations are sequentially. The process is designed according to (1)-(3). Since the decoder is intended for any kind of LDPC codes, the serial structure is chosen. The serial structure also consumes fewer resources.

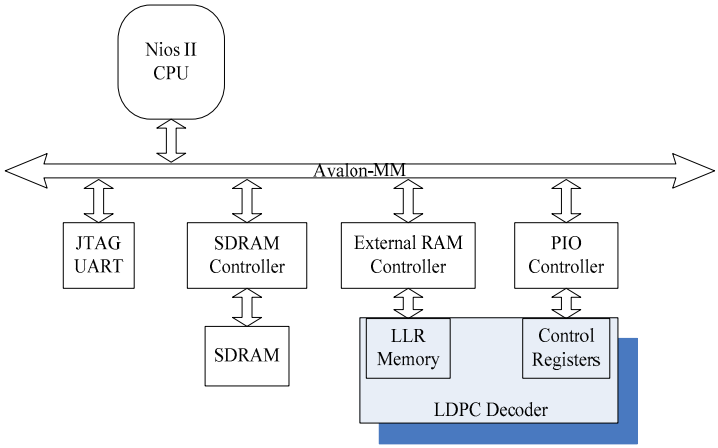

Fig. 2: The system structure.

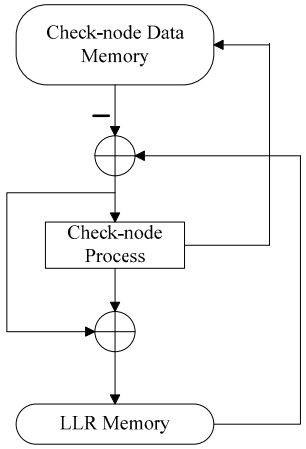

Fig. 3: LBP decoding processes.

Check-node Process (CNP) is a core part of the decoder. Structure of the CNP unit is shown in Fig. 4. Absolute values and signs (produced by the ABS module) are treated separately according (2). Let F denote the LUT of the *f(x)*, and let D denote the delayer. Two D delayers are used to balance latencies between the signs (upper) and absolute values (lower) data paths.

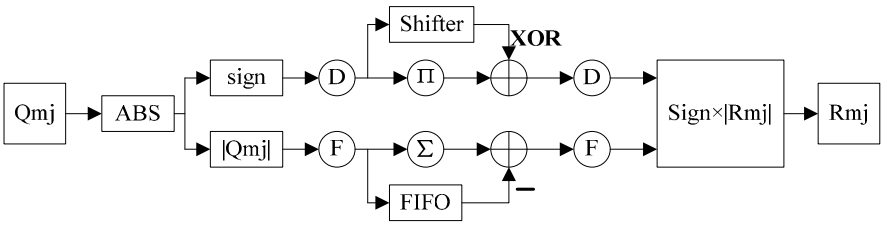

Fig. 4: The structure of CNP unit.

4. Implementation and Results

The described structure of the serial LDPC decoder is implemented on a Cyclone III FPGA. The LDPC Decoder occupies 307 logic elements (LEs) which is less than 2% of available LEs on the FPGA. The total combinational functions use 290 LEs and the total registers use 201 LEs. This low area implemented is a result of the serial structure. Table 1 shows resource usage summary of the LDPC Decoder and the entire system (including CPU).

Table 1: Resource usage summary.

	LDPC Decoder	Overall System
Estimated Total logic elements	307 / 15,408 (2%)	3,945 / 15,408 (26%)
Total combinational functions	290 / 15,408 (2%)	3,552 / 15,408 (23%)
Total registers	201 / 15,408 (1%)	2,300 / 15,408 (15%)
Total memory bits	183,366 / 516,096 (36%)	248,262 / 516,096 (48%)
Embedded Multiplier 9-bit elements	0 / 112 (0%)	4 / 112 (4%)

Take a (894,447) LDPC code for example. In Fig. 5 and Fig. 6, it is a compare of the BER and the number of iteration through the decoder system simulation. The simulation is made in different early stopping threshold X (X = 5, 7, 9). If the X value is set larger, the BER on the loss will be reduced, of course, the number of iterations will increase accordingly.

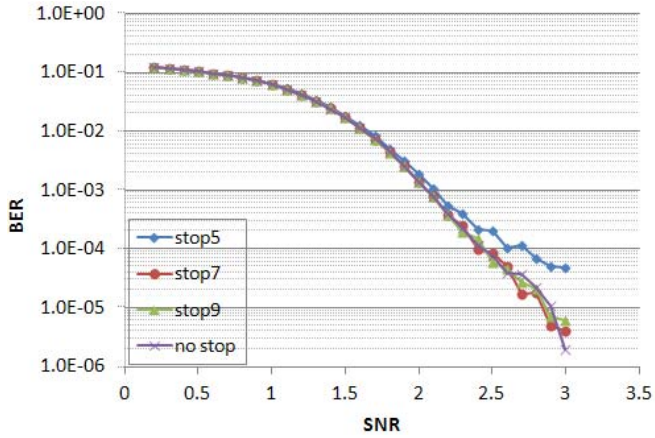

Fig. 5: BER simulation curve

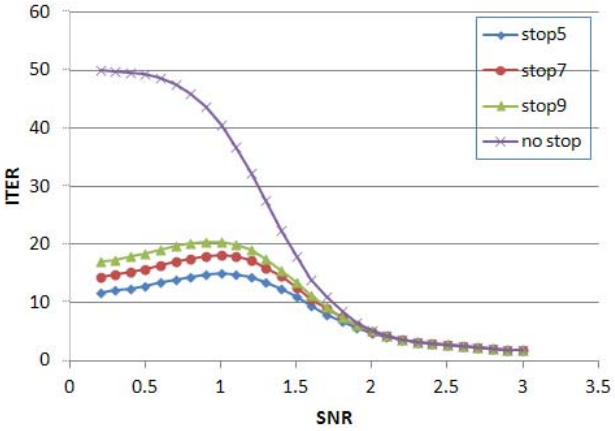

Fig. 6: ITER simulation curve

5. Conclusion

In this paper, we present an improved LBP decoding algorithm, and implement it on the Cyclone III FPGA. Including an early stopping criterion, the improved LBP algorithm not only has a high decoding speed, and a significant reduction on the memory required, but also greatly reduces the number of iterations in the low SNR. The entire system uses the SOPC design approach which provides the flexibility of software and offers high throughput of the hardware.

6. Acknowledgements

This study was supported by the National Natural Science Foundation (No.60902006), Huazhong University of Science and Technology School Fund (No.2012QN152).

7. References

[1] R. G. Gallager, Low-Density Parity-Check Codes. Cambridge, MA: MIT Press, 1963.

[2] Hocevar, D.E., A reduced complexity decoder architecture via layered decoding of LDPC codes. *Signal Processing Systems, 2004. SIPS 2004. IEEE Workshop on*, 2004, pp. 107- 112.

[3] Mohsenin, T., H. Shirani-mehr and B. Baas, Low power LDPC decoder with efficient stopping scheme for undecodable blocks. *Circuits and Systems (ISCAS), 2011 IEEE International Symposium on,* 2011, pp.1780-1783.

[4] Li, J., X. You and J. Li, Early stopping for LDPC decoding: convergence of mean magnitude (CMM). *Communications Letters, IEEE*, 2006, 9(10): pp. 667-669.

[5] Shin, D., et al., A Stopping Criterion for Low-Density Parity-Check Codes. *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, 2007, pp. 1529-1533.

[6] Hosseini, S.M.E., K.S. Chan and W.L. Goh, A reconfigurable FPGA implementation of an LDPC decoder for unstructured codes. *Signals, Circuits and Systems, 2008. SCS 2008. 2nd International Conference on*, 2008, pp.1-6.

[7] Verdier, F. and D. Declercq, A low-cost parallel scalable FPGA architecture for regular and irregular LDPC decoding. *Communications, IEEE Transactions on*, 2006, 54: pp. 1215 - 1223.

[8] Karkooti, M., P. Radosavljevic and J.R. Cavallaro, Configurable, High Throughput, Irregular LDPC Decoder Architecture: Tradeoff Analysis and Implementation. *Application-specific Systems, Architectures and Processors, 2006. ASAP '06. International Conference on*, 2006, pp. 360-367.

[9] Sun, Y., M. Karkooti and J.R. Cavallaro, VLSI Decoder Architecture for High Throughput, Variable Block-size and Multi-rate LDPC Codes. *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, 2007, pp. 2104-2107.

[10] Kim, D. and Y. Lee, Serial Scheduling Algorithm of LDPC Decoding, *in Convergence and Hybrid Information Technology*, G. Lee, D. Howard and D. Slezak, G. Lee, D. Howard and D. Slezak Editors.2011, Springer Berlin / Heidelberg. pp. 376.

Xiongfei Tao obtained his B.S. degree (1997) and M.S. degree (2001) in electric power system and automation both from Wuhan University, Wuhan, China. He received his Ph.D.degree in microelectronics and solid state electronics (2007) from Huazhong University of Science and Technology (HUST), Wuhan, China. Currently, he is working as a lecturer at Department of Electronic Science and Technology, HUST. His research interests include channel coding, digital television system, etc.