효율적인 LDPC 디코딩을 위한 노드 모니터링 알고리듬

*주경삼, 양석, 서희종
전남대학교 전자통신공학과(전공)
e-mail : *gongcheng0905@hotmail.com*

A Node Monitoring Algorithm for Efficient LDPC Decoding

*Qingsen Zhou, Shuo Yang, Heejong Suh
Division of Electronic Communication, Computer and Electrical Engineering
Chonnam National University

Abstract

In this paper, we propose an efficient algorithm for reducing the complexity of LDPC code decoding by using node monitoring (NM). This NM algorithm is based on a new node-threshold method, and the message passing algorithm. This algorithm was simulated in order to verify its efficiency. Simulation results show that the complexity of our NM algorithm is improved to about 10%, compared with well-known methods.

I. NODE MONITORING ALGORITHM

We introduce a decoding method based on the node monitoring algorithm. It updates messages only that instable nodes send at some iteration. In order to describe the degree of some nodes that have achieved a certain stable state, we define the "aggregate messages" B_i for each variable node as follows:

$$B_i = |V_n^{(i)}| = |U_{ch,n} + \sum_{m \in M(n)} U_{mn}^{(i)}| \quad (1)$$

Checking B_i against the node-thresholds t_v, will find the nodes that are stable. B_i is the confidence of the variable node to be in state 0 or 1, and the bigger B_i is, the more stable variable node is. We give two vectors to store the state of check nodes and variable nodes, respectively. Deactivated vectors are filled with zeros. Variable nodes get the information bits, and $V(0)_{nm} = U_{ch,n}$ .They send values as messages to their neighbor check nodes. When check nodes receive these messages, they send their messages $U(i)_{mn}$ to variable nodes. Then the variable nodes compute B_i to compare with the node-threshold t_v. If it is bigger than the t_v, the element in deactivated-v to this node will change to 1 from 0 representing a stable variable node. If a check node's neighbor variable nodes are all in stable state, the element in deactivated-c to this node will change to 1 from 0 representing a stable check node. They will never send the message to its neighbors. It reduces the complexity of the decoding process, but barely brings out degradation of the bit error rate (BER) and the frame error rate (FER) performance.

II. SIMULATION RESULTS

We simulated two kinds of LDPC code. First is the (3, 6) regular LDPC code with a block-length of 256 bits and rate R=0.5. And second is also the (3, 6) regular LDPC code, but with a block-length of 10^3 bits.

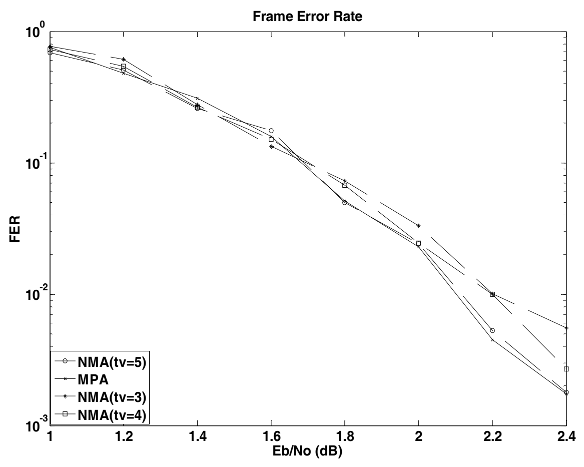

Fig. 1. Frame error performance of the LDPC code.

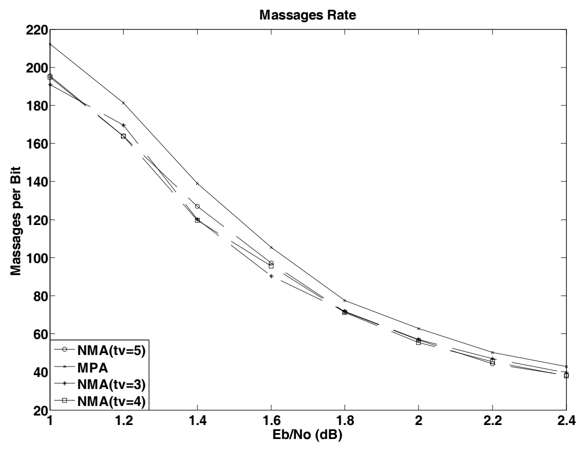

Fig. 2. Messages per bit of the LDPC code.

Fig. 1. show the performance of NM algorithm in conjunction with MP algorithm. It is easy to see that under the node-threshold of 3, 4, 5, the performance of the NM algorithm and MP algorithm.

Fig. 2. shows the messages per bit. It is easy to see that under the node-threshold of 3, 4, 5, the massages of the NM algorithm are less than MP algorithm by 10%.

Ⅲ. CONCLUSION

We have proposed a node monitoring algorithm to reduce the complexity of the LDPC decoding. With the simulation, we could see its better performance than existing well-known methods.

So, we can conclude that this method is the best way to reduce efficiently the complexity of the decoder. But, we must carefully choose node-threshold, because a node-threshold selection can result in serious performance degradation, and wrong decoding. And we must endeavor in order to achieve a practical node monitoring decoder for LDPC codes. And we must try to make more improvement to get better performance.

REFERENCES

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

[2] J. Zhang, M. Fossorier, D. Gu, and J. Zhang, "Two-dimensional correction for min-sum decoding of irregular LDPC codes,"*IEEE Commun. Lett.*, vol. 10, pp. 180-182, Mar. 2006.

[3] D. Levin, E. Sharon, and S. Litsyn, "Lazy Scheduling for LDPC Decoding," *IEEE Commun. Lett.*, vol. 11, pp. 70-72, Jan. 2007.

[4] D. H. Kim and S. W. Kim, "Bit-level stopping of turbo decoding," *IEEE Commun. Lett.*, vol. 10, pp. 183 - 185, 2006.