Low Density Parity Check (LDPC) Codes and the Need for Stronger ECC

August 2011

Ravi Motwani, Zion Kwok, Scott Nelson

Agenda

- NAND ECC History
- Soft Information
  - What is soft information
  - How do we obtain soft information?
  - How do we represent soft information?
- LDPC Decoding
  - Min sum decoder
  - How do we use soft information for decoding

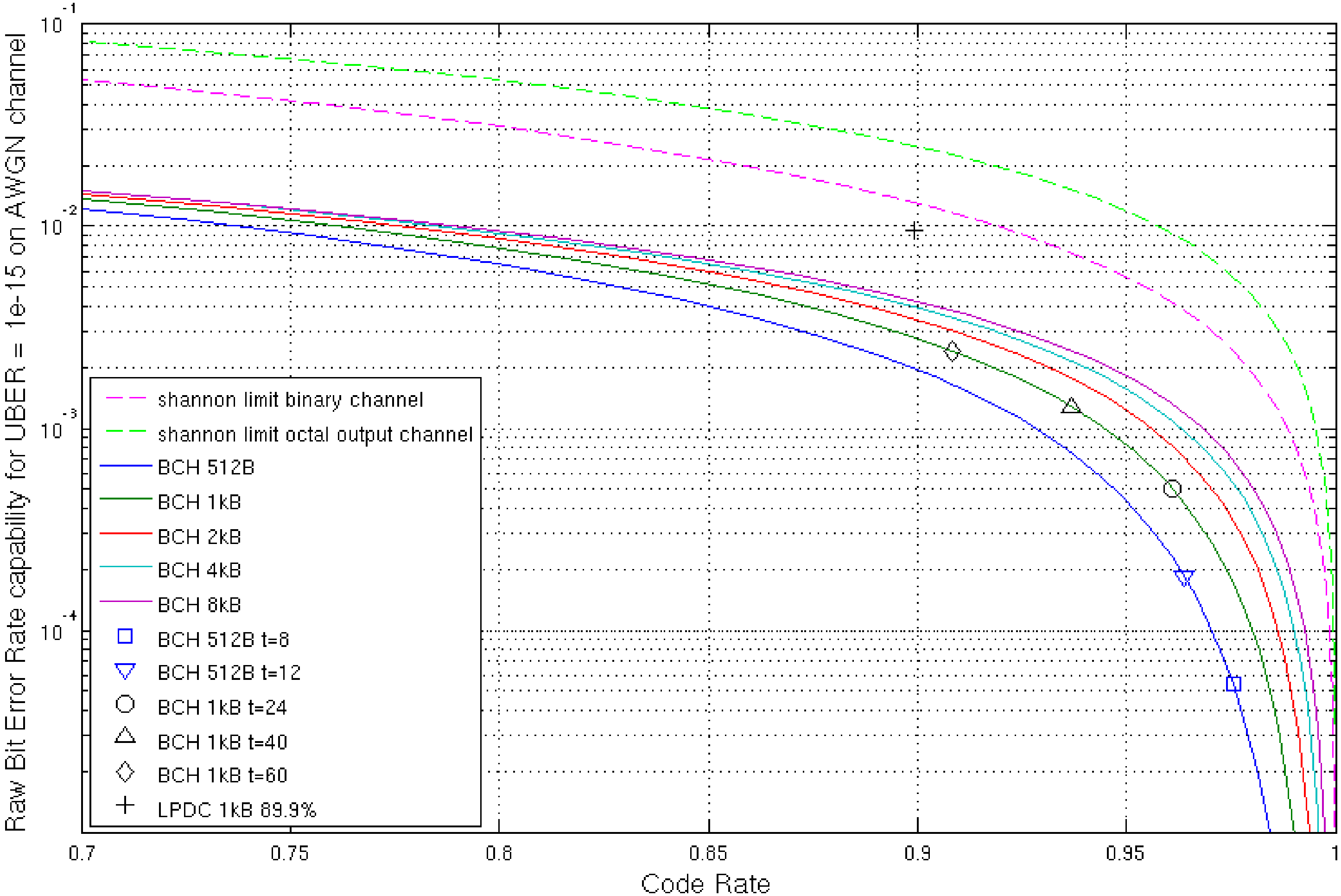
NAND ECC History

Intel / Micron NAND ECC

- 50nm MLC : 512B t=8
- 34nm MLC : 512B t=12
- 25nm MLC : 1024B t=24
- 25nm 3bit/cell : 1024B t=60
- 20nm MLC : 1024B t=40

- BCH codes are about to run into a brick wall . . .

NAND ECC Evolution



Soft Information

What is soft information?

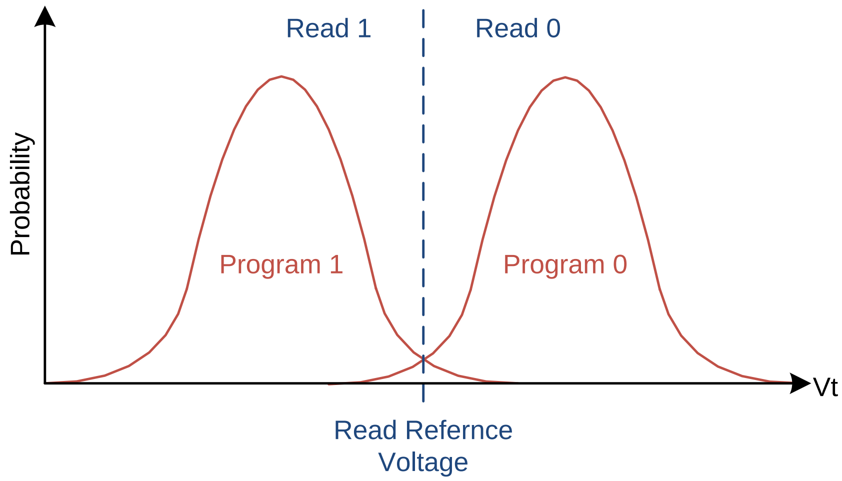
- Hard information: 1 or 0
  - channel output is best guess of original bit
- Soft information: Probability of a bit begin 1 or 0
  - Measure the reliability of each bit from channel
  - Decoder can give proper weight to the input information depending on its reliability

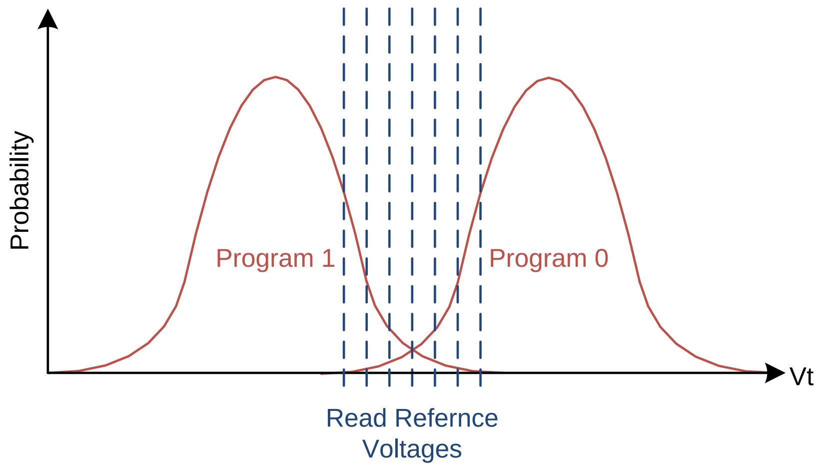
1	1	0	1	1	0	Hard input decoding
0.9	0.54	0.2	0.9	0.6	0.1	Soft input decoding

Soft Information

How do we obtain soft information?

- Read oversampling
- Binary input x_i (Program 0 or 1)
- Octal output y_j





Hard vs Soft Channel

- NAND Programming step is common to hard and soft channels
  - no change in the write path
  - can choose between hard or soft information at read time
- Hard Read
  - Faster read performance
  - Shorter NAND I/O time
- Soft Read
  - Higher Information rate - Decoder can handle higher RBER

Log Likelyhood Ratio (LLR)

- Symbol x_i is transmitted
- Symbol y_j is received

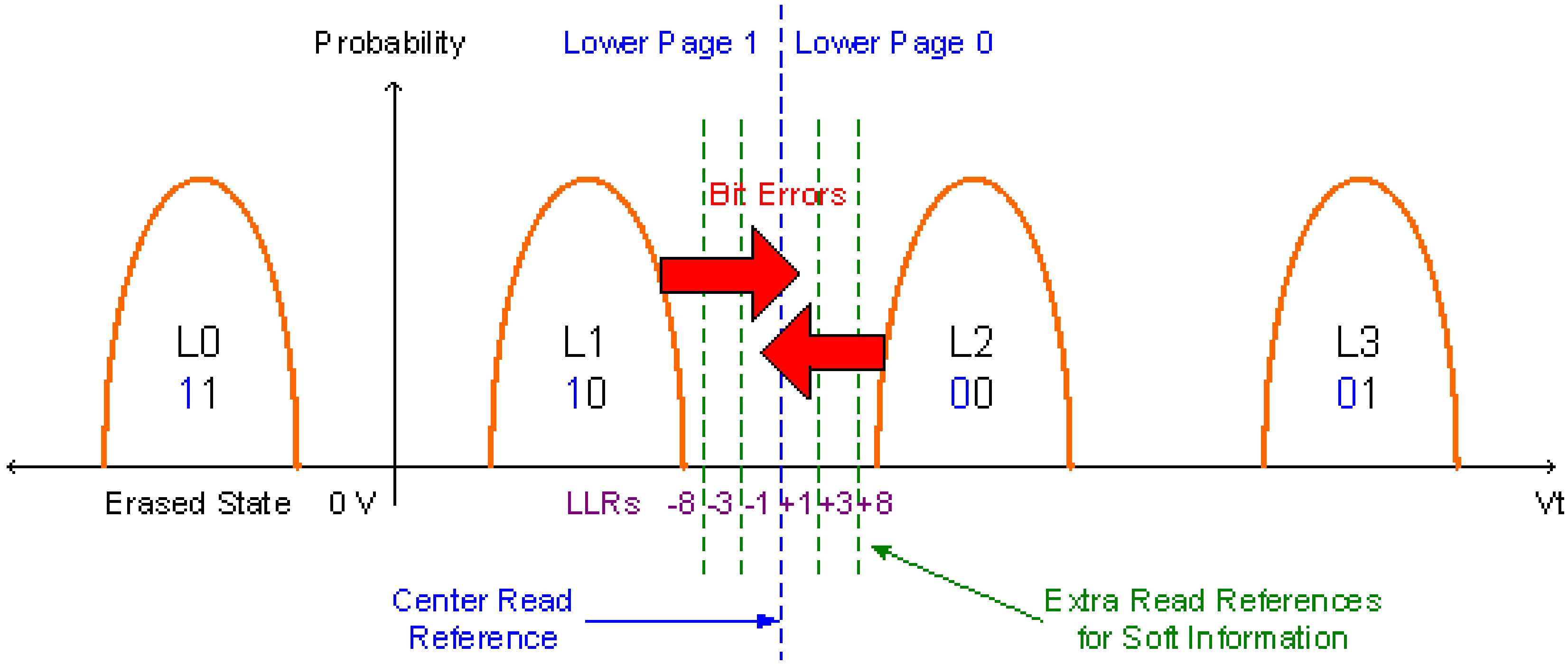
$$LLR(y_j) = \ln \frac{p(x=0 \mid y_j)}{p(x=1 \mid y_j)}$$

$$= \ln \frac{p(y_j \mid x=0)}{p(y_j \mid x=1)}$$

Integer LLR representation

- Simple hardware implementation
- Good Dynamic Range
- Good precision near p=0 and p=1
- Easy to add LLRs
  - Addition is multiplication of probabilities
  - combine two independent LLRs that give independent information about the same source variable x

Soft information readout with LLRs



LDPC codes

LDPC = Low Density Parity Check

- H matrix is sparse (less than 1% of matrix is 1s, remainder is 0s)
- Many ways to construct H matrix

LDPC Terminology

- Column Weight = # of 1s in each column of the H matrix
- Row Weight = # of 1s in each row of the H matrix
- Regular LDPC Code = All columns / rows have the same weight
- Tanner graph = a bipartite graph representing the H matrix

H matrix

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

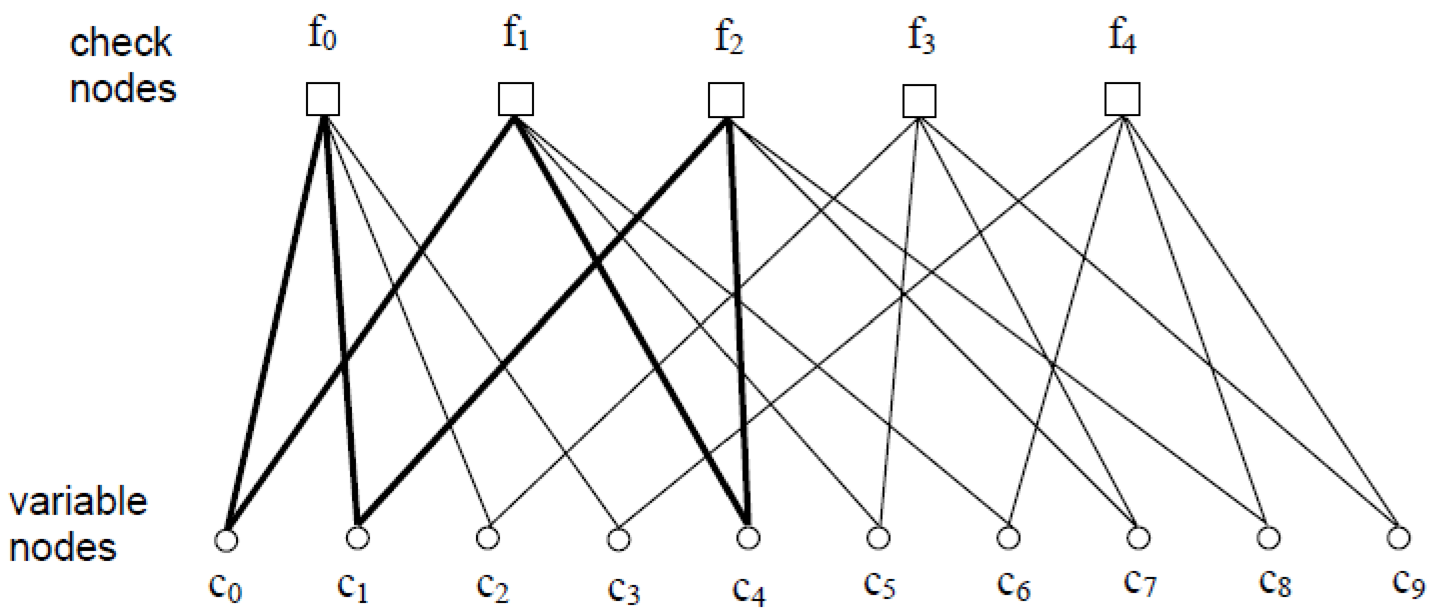
Codeword Size	10
Parity Check Equations	5
Row Weight	4
Column Weight	2

Tanner Graph

$$H = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}$$

Syndrome

$$cH^T = 0$$



LDPC information exchange

Parity check equation:

$$c_0 + c_1 + c_2 + c_3 = 0$$

Extrinsic information:

$$c_3 = c_0 + c_1 + c_2$$

Check Node Update

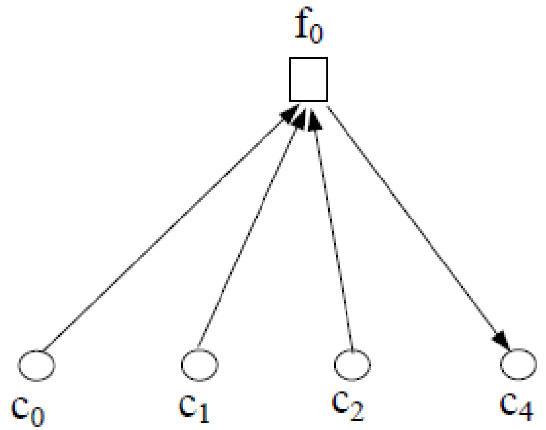
$$e(c_3) = \Psi(LLR(c_0), LLR(c_1), LLR(c_2))$$

Bit (variable) Node Update:

$$LLR(c_3) = LLR(c_3) + e(c_3)$$

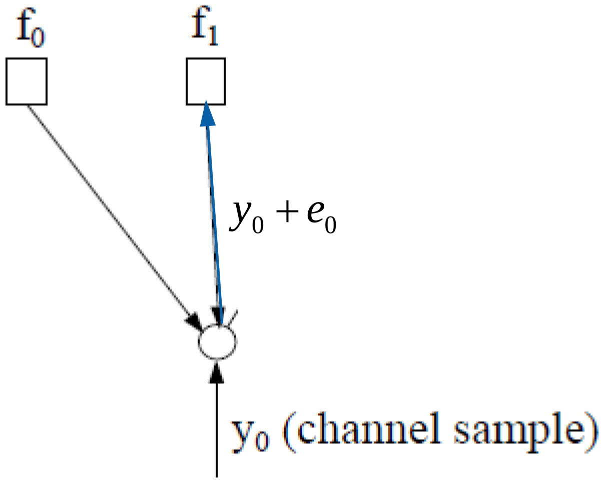
Min-sum decoder

$$\Pi_{i=0,1,2}\mathrm{sgn}(y_i)\min(|y_0|,|y_1|,|y_2|)$$



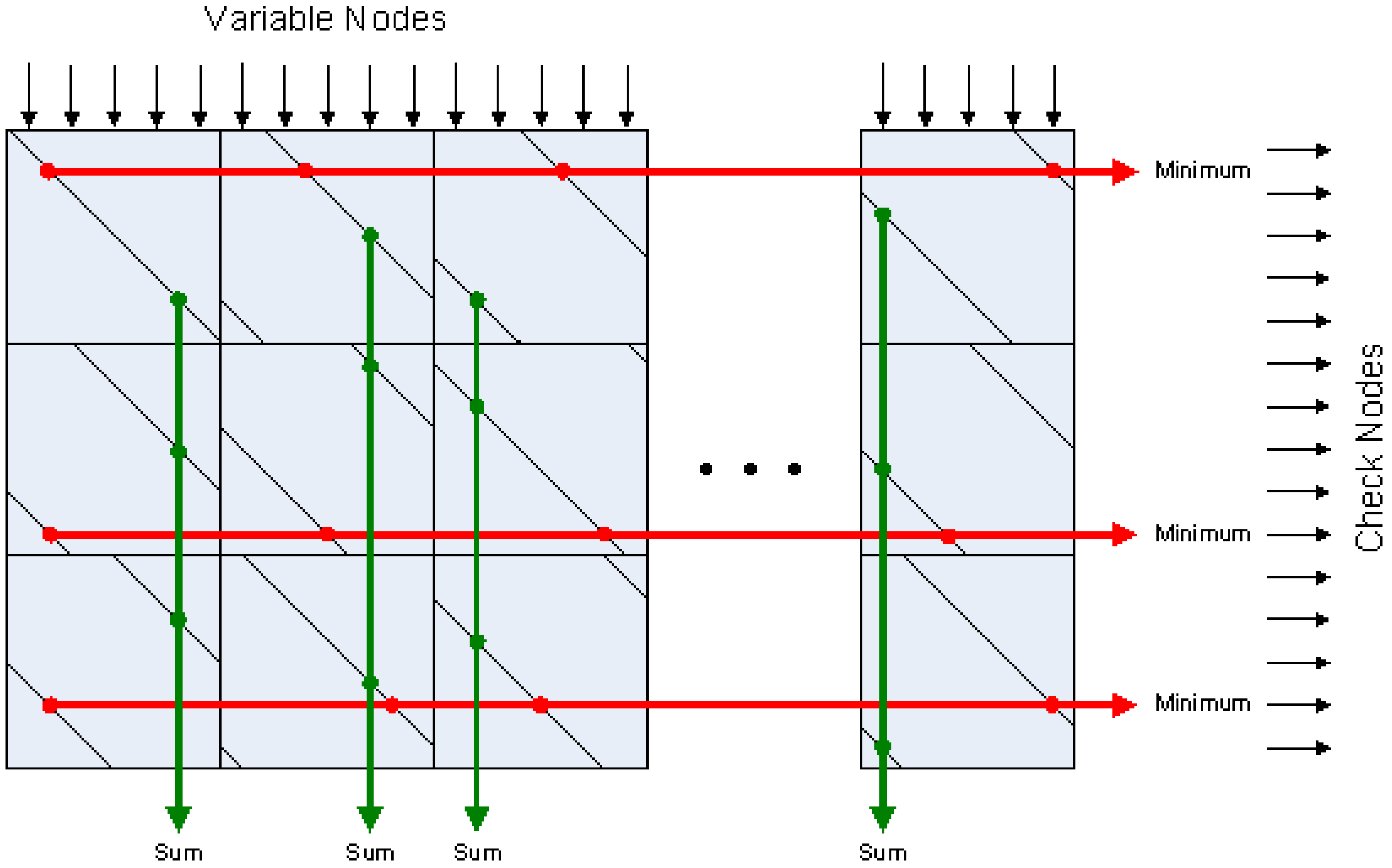
Check node update

Sum Operation



Bit node update

Quasi-cyclic H matrices



Drawbacks with LDPC

- Cannot mathematically characterize the performance
  - have to do simulation / emulation to measure code performance
  - 10^{16} or more bits to measure desired UBER
  - unlike BCH codes which are easy to predict
- Computationally intensive.
  - More information to process (soft input)
- Code-construction is challenging
  - Error floors may exist

Summary

- Soft information increases the channel capacity at the same RBER
  - Can choose between hard and soft information at readout time to tradeoff speed for decoding performance
  - Soft information represented using Log Likelihood Ratio (LLR)
- LPDC Min-Sum decoder can take advantage of soft information
  - Same decoder can use hard input or soft input
  - Higher channel capacity leads to better code performance at same RBER

References

[1] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, 27: 379-423, 623-656, July, October 1948.

[2] I. S. Reed and G. Solomon, “Polynomial Codes over Certain Fields,” *J. Soc. Ind. Appl. Math.*, 8: 300-304, June 1960.

[3] R. C. Bose and D. K. Ray-Chaudhuri, “On a Class of Error Correcting Binary Group Codes,” *Information and Control*, 3: 68-79, March 1960.

[4] A. J. Viterbi, “Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm,” *IEEE Trans. Inform. Theory*, 13: 260-269, April 1967.

[5] C. Berrou, A. Glavieux, and P. Thitimajshima, “Near Shannon Limit Error-Correcting Coding and Decoding: Turbo Codes,” *Proc. IEEE Intl. Conf. Communications (ICC 93)*, pp. 1064-1070, May 1993.

[6] R. G. Gallager, *Low-Density Parity-Check Codes*, M.I.T. Press, Cambridge, Mass., 1963.

[7] D. J. C. MacKay and R. M. Neal, “Near Shannon limit performance of low density parity check codes,” *IEE Electronics Letters*, 32: 1645-1655, Aug. 1996.

[8] M. Fossorier, M. Mihaljevic, and H. Imai, “Reduced complexity iterative decoding of low-density parity check codes based on belief propagation,” *IEEE Trans. Communications*, 47: 673-680, May 1999.

[9] X. Y. Hu, E. Eleftheriou, and D. M. Arnold, “Progressive edge-growth Tanner Graphs”, *IEEE GLOBECOM*, pp. 995-1001, Nov. 2001.

[10] Juntan Zhang, M. Fossorier, D. Gu, and Jinyun Zhang, “Improved Min-Sum Decoding of LDPC Codes Using 2-Dimensional Normalization,” *IEEE GLOBECOM*, pp. 1187-1192, Nov. 2005.

[11] E. R. Berlekamp, *Algebraic Coding Theory*, McGraw-Hill, New York, 1968.