

An Optimization Method for Designing LDPC Analog Decoders Based on Frequent Subgraph Mining Algorithm

Yuan Gao, Yujie Lin and Jibo Dai

*School of Information and Electronics, Beijing Institute of Technology, Beijing
100081, China
DsForM@126.com*

Abstract

To lower the mapping complexity of designing analog decoders, a method to optimize the design of low-density parity-check (LDPC) analog decoders is proposed in this paper. Based on factor graphs and the sum-product algorithm, the LDPC decoding process on the factor graph and the construction of analog decoders are exploited. Then the frequent subgraph mining algorithm is introduced to search the isomorphic subgraphs in factor graphs. According to the output of the frequent subgraph mining algorithm which enumerates all the subgraphs in factor graphs, the mapping complexity of a LDPC analog decoder can be significantly reduced. Finally, a (40, 16) LDPC analog decoder is constructed using the proposed method. Simulation results show that the need to place gates and connections can be reduced 90% and 23%, respectively, and the ideal performance is obtained by carefully choosing unit currents and decoding time.

Keywords: *low-complexity; LDPC; analog decoders; frequent subgraph mining*

1. Introduction

Low-density parity-check (LDPC) codes were first proposed by Gallager [1] in 1962, and have attracted much attention after rediscovery by Mackay [2] in 1996. Because LDPC codes are applied to a wide range of applications, such as digital video broadcasting (DVB) and wireless sensor networks (WSNs) [3], there is enormous interest in implementing low-cost iterative decoders with high throughput and low power consumption. Analog continuous-time iterative decoding was proposed several years ago to improve the power-to-speed ratio of iterative decoder chips [4]. Analog decoders are also more compact and generate lower switching noise compared with the digital counterparts [5].

In [6], analog decoders for turbo-style and tail-biting trellis codes were proposed, which provided a different way for LDPC decoder implementation. Several analog decoders for the (32, 8) LDPC code have been proposed [5-8] using different decoding algorithms, such as the min-sum (MS) algorithm and the belief propagation (BP) algorithm. All of the analog decoders gained lower power consumption and smaller core area compared with digital decoders. A (120, 75) LDPC analog decoder with the MS algorithm was proposed in [9], which was one of the longest length codes implemented to date using analog techniques [10]. However, due to the complexity of mapping during the analog decoder design, the application of LDPC analog decoders is limited by the code lengths.

To lower mapping complexity, an optimization method based on factor graphs and the subgraph mining algorithm is proposed. By forming the algorithm of LDPC decoding on a undirected bipartite factor graph, a frequent subgraph mining algorithm is applied to the graph to determine the set of subgraphs which repeatedly appear in the factor graph. With these subgraphs, the mapping complexity of a LDPC analog decoder can be significantly reduced.

The remainder of the paper is organized as follows. Section 2 describes the factor graph and analog decoding process over memoryless channels. In Section 3, we propose an optimization method for analog decoder mapping based on the frequent subgraph mining algorithm. An approach based on the proposed method is introduced to analyze performance of the analog decoder in Section 4. In Section 5, an implementation of LDPC decoders is proposed, and a simulation analysis is also undertaken.

2. The Graphic Model of Analog Decoding over Memoryless Channels

A (N, K) LDPC code is defined as a binary linear code over $GF(2)$. An $M \times N$ check matrix is represented by H with $rank(H) = N - K \leq M$. Let the code words set be C , which is a $2K-1$ vectors subset of $GF(2^n)$, then $cH^T = 0, \forall c \in C$. The analog decoders of the aforementioned LDPC codes can be defined over an undirected bipartite factor graph, $G = (V, E)$, which consists of a set of vertices, including N variable nodes (VNs: $X_i, \{i = 1, 2, \dots, N\}$) and M check nodes (CNs: $C_j, \{j = 1, 2, \dots, M\}$). Each node is connected by a set of edges, E , which represents the check matrix, H , if the corresponding element in H is 1. $X_i \in \{0, 1\}$ and $Y_i \in R$ (i is a integer ranging from 1 to N) are denoted as variables that correspond to the i -th transmitted symbol and the i -th received symbol of a codeword, respectively. Then a memoryless binary-input output-symmetric channel can be defined by a conditional probability density function as follows:

$$f_{Y_i/X_i}(y_i/x_i) = f(Y_i = y_i / X_i = x_i) \quad (1)$$

from which the priori probability, $p(y_i/x_i)$, could be obtained.

With the factor graph of (N, K) LDPC codes and the priori probability, $p(x_i/y_i)$, the decoding process can be achieved by a “sum-product algorithm” to the graphical model [12]. The priori probability, $p(x_i/y_i)$, is used to initialize the procedure of the sum-product algorithm, which can be expressed as follows:

CNs update: CN c_j calculates the message, $\mu_{c_j \rightarrow x_i}(x)$, sent to $x_i \in X_j$ using message, $\mu_{y \rightarrow c_j}(x)$ ($y \in X_j$):

$$\mu_{c_j \rightarrow x_i}(x) = \sum_{\sim \{x_i\}} f(X) \prod_{y \in X_j \setminus \{x_i\}} \mu_{y \rightarrow c_j}(x) \quad (2)$$

where $f(X)$ is the constraint function of CN, X_j denotes the set of neighbor VNs connected to CN c_j , and $X_j \setminus \{x_i\}$ represents the set, X_j , except x_i .

VNs update: VN x_i calculates the message, $\mu_{x_i \rightarrow c_j}(x)$, sent to $c_j \in C_i$ using the message, $\mu_{h \rightarrow c_j}(x)$ ($h \in C_i$):

$$\mu_{x_i \rightarrow c_j}(x) = \prod_{h \in C_i \setminus \{c_j\}} \mu_{h \rightarrow x_i}(x) \quad (3)$$

where C_n denotes the set of neighbor CNs connected to VN, x_i , and $C_n \setminus \{c_m\}$ represents the set, C_n , except c_m .

Generally, the graph, G , of LDPC codes in this paper is a complete bipartite graph and also a simple graph, in which self-loops are exclusive. From (2) and (3), we can note that the process of message passing refers to the edge in graph, G , which corresponds to the mapping of analog decoders. In graph G , the node with degree, D ($D > 3$), is implied by a sequence of 3 ports bi-directional Soft-XOR gates or Equal gates. The problem of analog decoders mapping optimization is equal to the problem of frequent subgraph mining,

which determines whether or not a given graph occurs in another graph and enumerates all the frequent subgraphs.

3. Frequent Subgraph Mining Algorithm

To optimize the mapping of graph model, G , we have to extend the graph, G , to label graph GL . A labeled graph, GL , is a five-element tuple, $GL = (V, E, \Sigma V, \Sigma E, l)$, where V is a set of vertices, $E \subseteq V \times V$ is a set of edges, ΣV is the set of vertex labels, and ΣE is the set edge labels. In the graph model of LDPC codes, $\Sigma E = \{1,0\}$ and l is a function assign label to vertices. Without loss of generality, we assume that there is a total order \geq on each label set ΣV and ΣE . The mapping complexity of an analog decoder is determined by the edges, which are connected by Soft-XOR gates and Equal gates or blocks. The optimization of analog decoder mapping is to reduce the edges of the graph model or to enhance the multiplex of gates-blocks. Thus, we introduce the subgraph isomorphic to indicate the block multiplexing. A labeled graph, GL , is a subgraph isomorphic to a labeled graph, GL' , denoted by $GL \subseteq GL'$, if and only if there is a subgraph GL'' of GL' such that GL is isomorphic to GL'' . The reusability of a given graph of a block is equivalent to the "support" of a graph. The support of a graph, G , denoted by sup_G , is defined as the fraction of graphs in GS to which GL is subgraph isomorphic.

$$\text{sup}_G = \frac{|\{G' \in GS | G \subseteq G'\}|}{GS} \quad (4)$$

where GS is a set of graphs and $\sigma (0 < \sigma \leq 1)$ is the threshold. GL is frequent if and only if $\text{sup}_G \geq \sigma$.

The optimizing problem of mapping an analog decoder represented by labeled graph can be equivalent to the problem of frequent subgraph mining, which is to enumerate all the frequent subgraphs in GS for the given threshold and graph database.

The adjacency matrix, $A(G)$, of graph G is translated as follows:

$$A(G) = \begin{bmatrix} 0 & H^T \\ H & 0 \end{bmatrix} \quad (5)$$

where H is the check matrix. The vertex labels are given by $V_k = N \cdot i + j, k \in \{0,1,\dots,M \times N - 1\}$. The adjacency matrix, $A(GL)$, we used here, can be represented by filling the diagonal entry of $A(G)$ with the label, V_k , of the corresponding node. Then, we define the code of the adjacency matrix $A(GL)$, shortened as A , as the sequence of lower triangular entries of A in the following order: $\{a_{1,1}; a_{2,1}; a_{2,2}; \dots; a_{n,1}; a_{n,2}; \dots; a_{n,n-1}; a_{n,n}\}$, where $a_{i,j}$ represents the entry at the i -th row and j -th column in A . To find out the unique pattern of a given adjacency matrix in all its possible forms, we use the $C_{\max}(A)$ to denote the maximal code among all its possible codes, which are in standard lexicographic order. The adjacency matrix, A' , which produces the canonical form is defined as GL 's canonical adjacency matrix (CAM).

Given a graph GL , a suboptimal CAM of GL is an adjacency matrix A of G such that its maximal proper submatrix N is the CAM that graph N represents.

All the suboptimal CAMs of graph GL can be organized in a tree as follows by using the operation of FFSM-Join and FFSM-Extension proposed in [11]:

- (i) The root of the tree is empty matrix;
- (ii) Each node in the tree is a distinct connected subgraph of G , represented by $C_{\max}(A)$;
- (iii) For a given non-root node, its parent is the graph represented by A 's submatrix.

Clearly, the suboptimal CAM tree is "complete" in the sense that all vertices in a suboptimal CAM tree can be enumerated using join and extension operations.

To find out the optimization scheme of analog decoder mapping, we define the mapping cost function, $f_{map}(s, w)$, as follows:

$$f_{map}(s, w) = s \sum |A(w)|_0 \quad (6)$$

where s is the support of a graph, GL , w is the frequent subgraph of GL , and $|A(w)|_0$ denotes the 0-norm of adjacency matrix, w , which can be obtained by counting the non-zero elements of w . Generally, the optimization of analog decoder mapping scheme (s', w') is:

$$(s', w') = \arg \max(f_{map}(s, w)) \quad (7)$$

We use the frequent subgraph mining algorithm proposed in [11] to enumerate the subgraphs, and the output of the algorithm is the set W which contains CAMs of all the searched frequent subgraphs.

4. Constructing the LDPC Analog Decoder with Kernel Blocks

4.1. Simplification of Analog Decoder Mapping for CCSDS LDPC

To examine our method of analog decoder mapping optimization, we perform the LDPC codes presented in CCSDS 131.1-O-1 standard (shortened as CCSDS-LDPC).

The check matrices of CCSDS-LDPC for rate-1/2 can be specified as follows:

$$\mathbf{H} = \begin{bmatrix} \mathbf{0}_M & \mathbf{0}_M & \mathbf{0}_M & \mathbf{I}_M & \mathbf{I}_M \oplus \mathbf{\Pi}_1 \\ \mathbf{I}_M \oplus \mathbf{\Pi}_8 & \mathbf{I}_M \oplus \mathbf{\Pi}_7 \oplus \mathbf{\Pi}_6 & \mathbf{0}_M & \mathbf{0}_M & \mathbf{I}_M \\ \mathbf{0}_M & \mathbf{I}_M & \mathbf{I}_M \oplus \mathbf{\Pi}_5 & \mathbf{0}_M & \mathbf{\Pi}_4 \oplus \mathbf{\Pi}_3 \oplus \mathbf{\Pi}_2 \end{bmatrix} \quad (8)$$

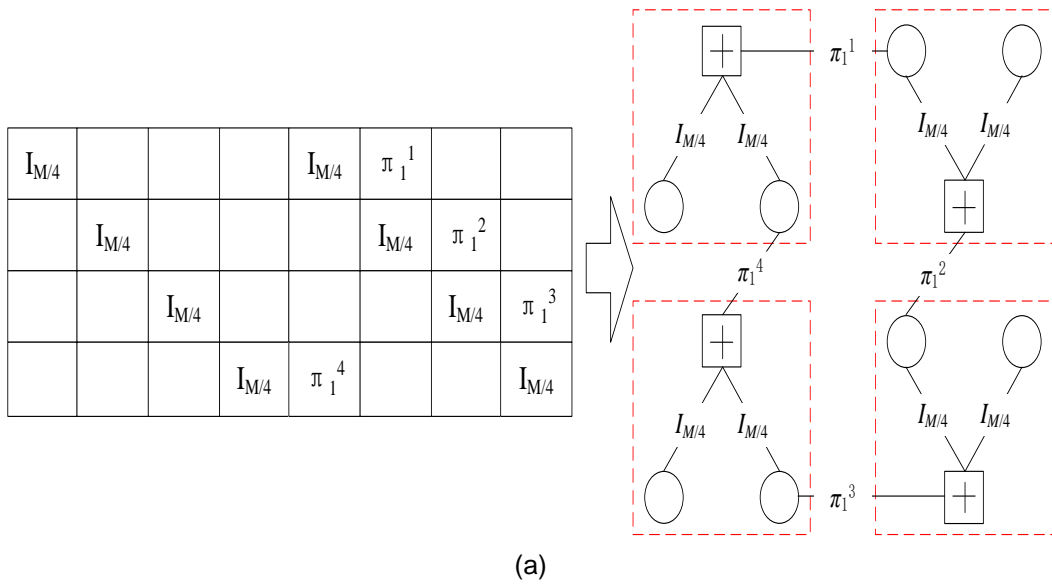
where \mathbf{I}_M and $\mathbf{0}_M$ are $M \times M$ identity and zero matrices, respectively, and through $\mathbf{\Pi}_1$ to $\mathbf{\Pi}_8$ are $M \times M$ permutation matrices. The distribution of row and column weight of check matrix is shown in Table 1.

Table 1. The Distribution of Row Weight and Column Weight of Check Matrix

Columns/Rows	1~M	M+1~2M	2M+1~3M	3M+1~4M	4M+1~5M
Columns Weight	2(+1)	3(+1)	2(+1)	1(+1)	6(+1)
Rows Weight	3	6	6	--	--

In order to output the posterior probability of each bit, an edge to output extrinsic information is added to each VN in the factor graph, which is denoted by (+1) in Table 1.

Thus, we can obtain the modified check matrix of LDPC codes. By following the steps aforementioned in this paper, the subgraphs of LDPC codes can be sorted into two kinds, which are shown on the left side of Figure 1. Each nonblank block is a $M/4 \times M/4$ matrix, and blank blocks are zero matrices, while $I_{M/4}$ and π_x^y are $M/4 \times M/4$ identity and permutation matrices representing connections between $M/4$ CNs and $M/4$ VNs.



$I_{M/4}$				$I_{M/4}$	π_a^1			$I_{M/4}$	π_b^1	π_c^1	
	$I_{M/4}$				$I_{M/4}$	π_a^2			$I_{M/4}$	π_b^2	π_c^2
		$I_{M/4}$				$I_{M/4}$	π_a^3	π_c^3		$I_{M/4}$	π_b^3
			$I_{M/4}$	π_a^4			$I_{M/4}$	π_b^4	π_c^4		$I_{M/4}$

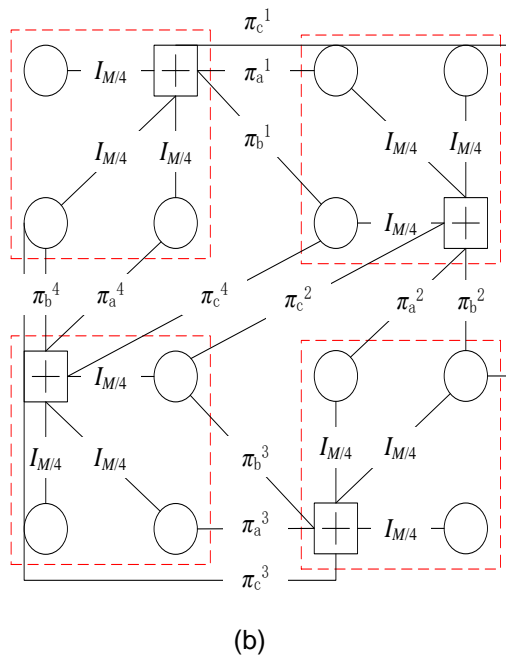


Figure 1. Structures of Subgraphs (Present by Adjacency Matrix): (a) Subgraph for 1~M Rows; (b) Subgraph for M+1~2M and 2M+1~3M Rows

Frameworks of two kinds of subgraphs are shown on the right/lower side of Figure 1, each \boxplus (\circ) represents $M/4$ CN (VN), with the connection between the frameworks if and only if the corresponding block is not blank. On the right side of Figure 1(a), each subgraph inside the dashed box owns the same structure, which can be viewed as a basic kernel block, as well as subgraphs on the lower side of Figure 1(b).

Hence, a factor graph of LDPC codes according to the CCSDS standard can be composed by two kinds of basic kernel blocks presented on the right/lower side of Figure 1.

4.2. Structure of the Corresponding Analog Decoder

There are two sorts of basic kernel blocks of CCSDS-LDPC analog decoders according to Figure 1. The first basic kernel block, which realizes subgraph on the right side of Figure 1 (a), is composed by $M/4$ Soft-XOR gates with degree 3 and $M/4$ Equal gates with degree 4, while the second basic kernel block realizes the subgraph on the lower side of Figure 1 (b), and can be composed by $M/4$ Soft-XOR gates with degree 6, $M/4$ Equal gates with degree 4 and $2M/4$ Equal gates with degree 3. Making use of these kernel blocks, the framework of the proposed LDPC analog decoders is shown as Figure 2.

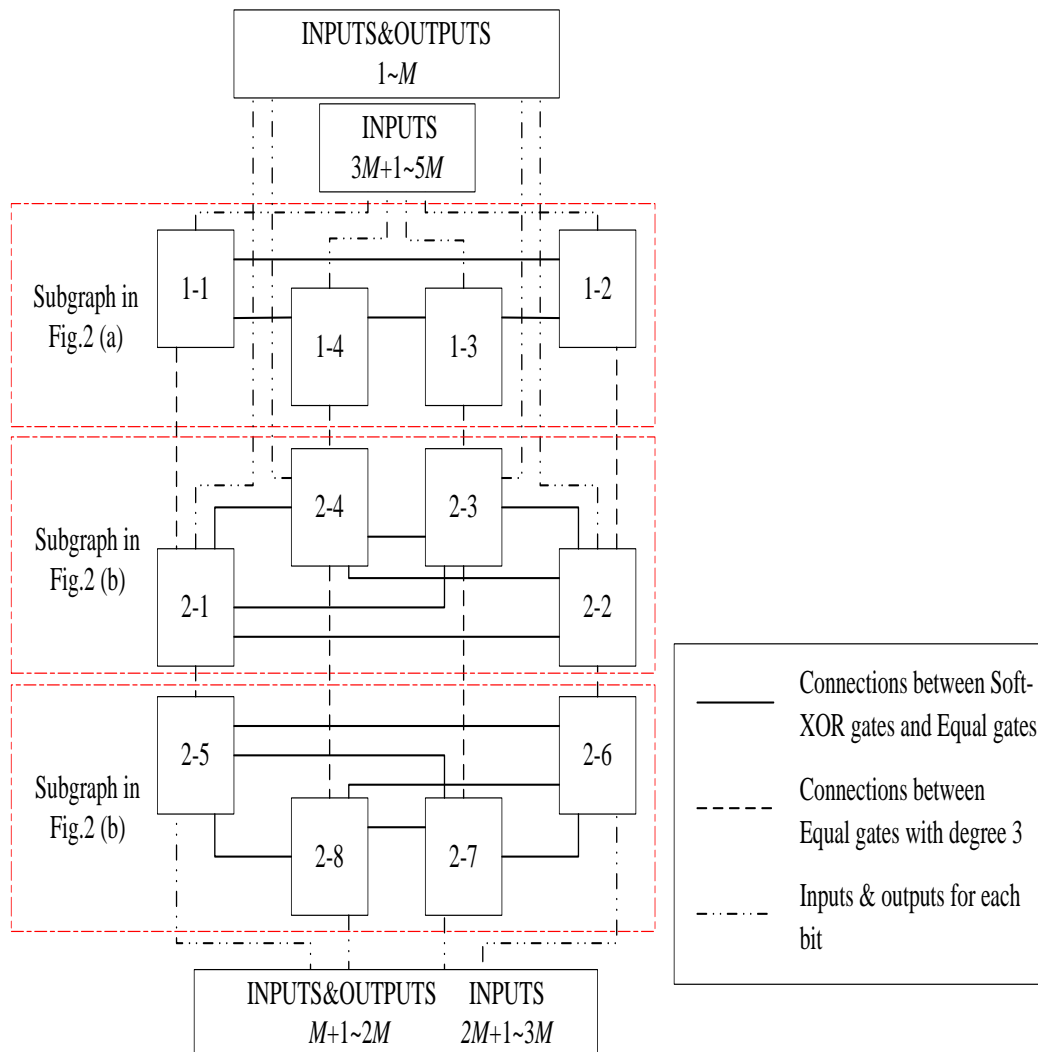


Figure 2. Framework of CCSDS-LDPC Analog Decoders

In Figure 2, squares with “x-y” inside are kernel blocks corresponding to subgraphs on the right/lower side of Figure 1, while “x” and “y” indicate which sort of kernel block and what order a kernel block is, respectively. The solid lines between kernel blocks are connecting relationships between kernel blocks corresponding to π_x^y in Figure 1, which exist in the original factor graph, while dashed lines and dotted lines are links that connect Equal gates in the chain, as is illustrated in Figure 1, and inputs (or outputs) of bits for decoders, respectively.

Figure 2, shows the relationship between different kernel blocks and reveals that the CCSDS-LDPC analog decoders can be composed by two sorts of circuits corresponding to subgraphs on the right/lower side of Figure 1, which eventually lower the mapping complexity from the whole decoder to two sorts of kernel blocks and connections between them.

5. The (40, 16) LDPC Analog Decoder and Simulation Results

In this section, an example of (40, 16) LDPC analog decoder from CCSDS standard is implemented from two sorts of kernel blocks, and the mapping complexity and performance are both analyzed.

5.1. (40, 16) LDPC Analog Decoder and Mapping Complexity Analysis

To construct the (40, 16) LDPC analog decoder, two sorts of kernel blocks of the (40, 16) LDPC analog decoder are implemented and exhibited in Figure 3, which realizes subgraphs on the right/lower side of Figure 1. The first sort of kernel blocks is composed by 2 degree-3 Soft-XOR gates and 2 degree-4 Equal gates, and the second kernel block is composed by 2 degree-6 Soft-XOR gates, and 2 degree-4 Equal gates with and 4 degree-3 Equal gates.

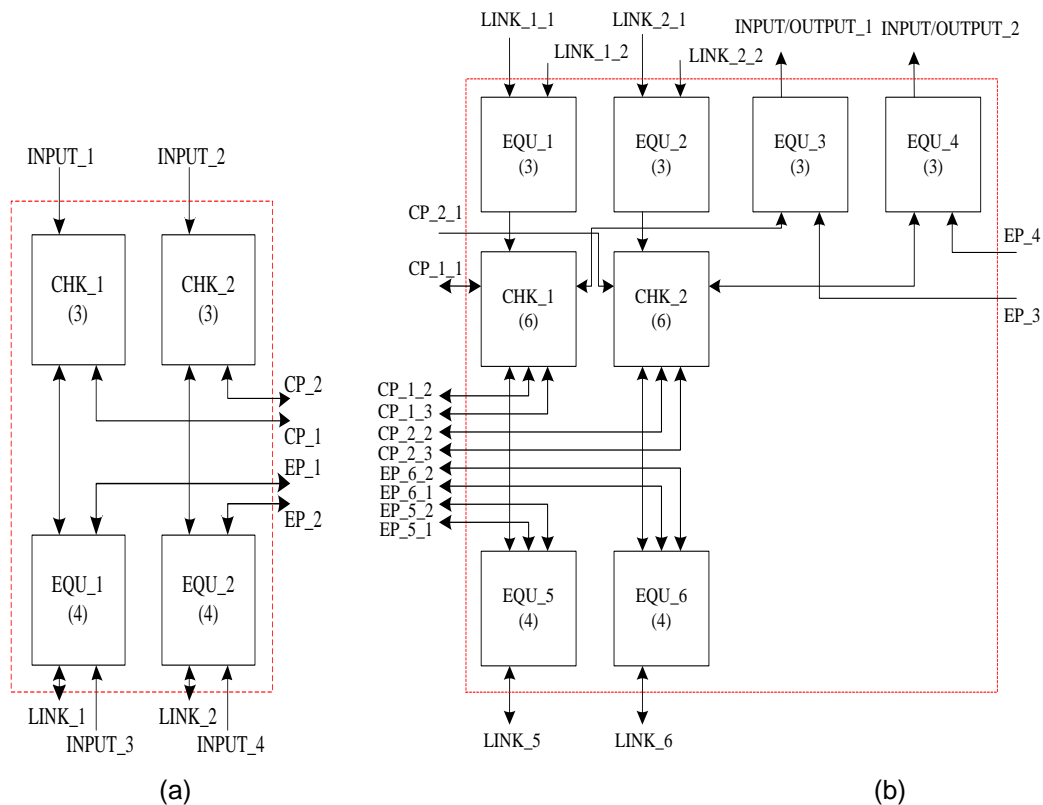


Figure 3. Two Kernel Blocks: (a) The First One; (b) The Second One

With four kernel blocks of the first sort and eight kernel blocks of the second sort, the analog decoder can be obtained following the framework in Figure 3. Table 2, gives the mapping complexity comparisons between the proposed method and the conventional way maps the whole decoder. In Table 2, “Conventional” denotes the analog decoder built in the conventional way, while “Degree-3 Gates” and “Average Connections per Input Bits” indicate the number of degree-3 gates that need to be placed and the number of average connections need to be mapped when the number of input bits is increased by one.

Table 2. Mapping Complexity Comparisons in Designing

	Degree-3 Gates	Average Connections per Input Bits
Conventional	216	11.2
Proposed	22	8.6

As the proposed method only maps two kernel blocks and wires connecting the blocks, the number of degree-3 gates placed to construct the analog decoders is reduced to 10% of the conventional method, while the number of connections mapped is decreased by 23%. The mapping complexity of designing analog decoders is lowered to 20% compared with the conventional way.

5.2. Simulation Results of the (40, 16) LDPC Analog Decoder

To testify the feasibility of the (40, 16) LDPC analog decoder, the simulation model from [13] is improved by taking mismatch of transistors, error brought by IC technics and unit currents (I_u) into account. In the simulation, Soft-XOR gates and Equal gates together implement the BP algorithm over AWGN channel with BPSK modulation. The mismatch of transistors and error brought by IC technics are transformed into parameters in each edge, which obey (0, 0.2) Gauss distribution ranging from -1 to 1.

Figure 4, depicts the bit error rate (BER) performance of decoder versus E_b/N_0 for different decoding time (T_d) and I_u . It can be observed that the larger the decoding time, the better the BER performance. When the decoding time is too short, such as 20ns, the decoder is likely to stop decoding process before it converges. However, the BER performance of decoder approaches the ideal performance rapidly as decoding time increases. A decoder with $I_u=5\mu A$ needs less decoding time compared with decoders with $I_u=1\mu A$, which means that to strike balance of power consumption and decoding time one needs to manipulate I_u .

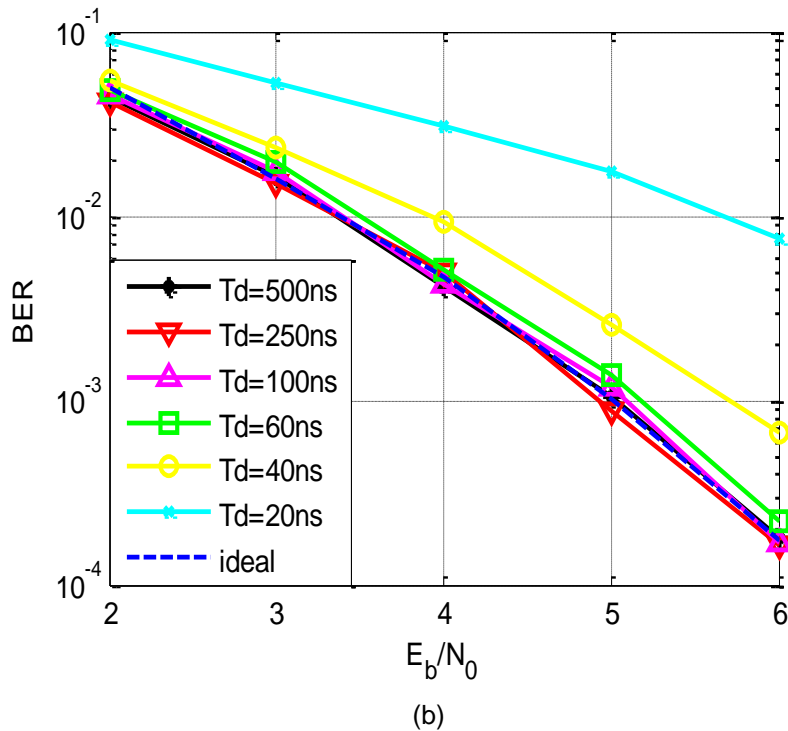
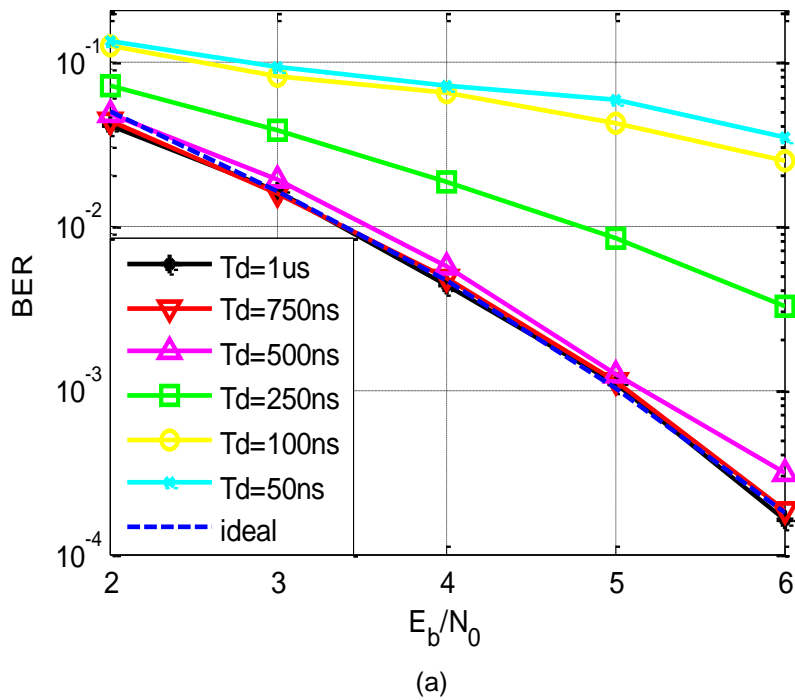


Figure 4. Simulation Results of the (40, 16) Analog Decoder: (a) $I_u = 1\mu A$; (b) $I_u = 5\mu A$

6. Conclusion

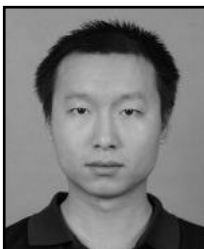
An optimization method for designing analog decoders is presented. By exploiting LDPC decoding process on factor graphs and applying the frequent subgraph mining algorithm on factor graphs, the analog decoder of LDPC can be simplified as connecting frequent subgraphs. This method enables system-level optimization of an analog decoder,

regardless of the physical realization. Moreover, this method can extend to optimize the design of other analog decoders for linear block codes, such as BCH and RS codes. According to the simulations of an example analog decoder, the average mapping complexity of decoders can be lowered 80%, and simulation results indicate that the decoder can achieve the BER performance as good as the ideal decoder.

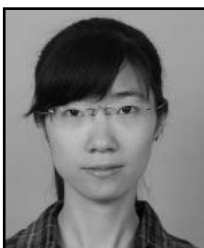
References

- [1] R. G. Gallager, "Low-Density Parity-Check Codes", IRE Transactions on Information Theory, vol. 8, no. 1, (1962), pp. 21-28.
- [2] D. J. C. MacKay and R. M. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes", Electronics Letters, vol. 32, no. 18, (1996), pp. 1645-1646.
- [3] J. Zhang, Y. Piao and F. Yin, "Research On Image Watermarking Applied to the Atmospheric Laser Communication", Journal of Harbin University of Science and Technology, vol. 15, no. 5, (2010), pp. 90-94.
- [4] Y. Cao, Z. Wang, J. Chen and L. Shi, "Design of an OTA in a Low Power A/D Converter", Journal of Harbin University of Science and Technology, vol. 15, no. 2, (2010), pp. 83-87.
- [5] S. Hemati, A. H. Banihashemi and C. Plett, "A 0.18- μm CMOS Analog Min-sum Iterative Decoder for A (32,8) Low-Density Parity-Check (LDPC) Code", IEEE Journal of Solid-State Circuits, vol. 41, no. 11 (2006), pp. 2531-2540.
- [6] F. Lustenberger, "On the Design of Analog VLSI Iterative Decoders", Zürich: Swiss Federal Institute of Technology, (2000).
- [7] M. Gu and S. Chakrabartty, "A 100 pJ/bit, (32,8) CMOS Analog Low-Density Parity-Check Decoder Based on Margin Propagation", IEEE Journal of Solid-State Circuit, vol. 46, no. 6, (2011), pp. 1433-1442.
- [8] D. Miyashita, R. Yamaki, K. Hashiyoshi, H. Kobayashi, S. Kousai, Y. Oowaki and Y. Unekawa, "A 10.4pJ/b (32,8) LDPC Decoder with Time-domain Analog and Digital Mixed-signal Processing", 2013 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, USA, (2013).
- [9] A. R. Abolfazli, Y. R. Shayan and G. E. R. Cowan, "750Mb/s 17pJ/b 90nm CMOS (120,75) TS-LDPC Min-Sum Based Analog Decoder", 2013 IEEE Asian Solid-State Circuits Conference (A-SSCC), Singapore, (2013).
- [10] K. S. Andrews, D. Divsalar, S. Dolinar, J. Hamkins, R. Jones and F. Pollara, "The Development of Turbo and LDPC Codes for Deep-Space Applications", Proceedings of the IEEE, vol. 95, no. 11, (2007), pp. 2142-2156.
- [11] J. Huan, W. Wang and J. Prins, "Efficient Mining of Frequent Subgraphs in the Presence of Isomorphism", 2003 IEEE International Conference on Data Mining (ICDM), Melbourne, Florida, USA, (2003).
- [12] S. M. Aji and R. J. McEliece, "The Generalized Distributive Law", IEEE Transactions on Information Theory, vol. 46, no. 2 (2000), pp. 325-343.
- [13] S. Hemati and A. H. Banihashemi, "Dynamics and Performance Analysis of Analog Iterative Decoding for Low-density Parity-check (LDPC) Codes", IEEE Transactions on Communications, vol. 54, no. 1, (2006), pp. 61-70.

Authors



Yuan Gao, he received his B.S. and M.S degrees from Beijing Institute of Technology in 2007 and 2010. Now he is a Ph.D. candidate in Beijing Institute of Technology in major of information and communication engineering. His research interests include information theory coding and satellite communications systems.



Yujie Lin, she received his B.S. degrees from Beijing Institute of Technology in 2011. She is a Ph.D. candidate in Beijing Institute of Technology in major of information and communication engineering from 2011. Her research interests include communication theory and array signal processing.



Jibo Dai, he received his B.S. degree in communication engineering from Beijing Institute of Technology in 2014. Currently, he is the M.S. candidate of Beijing Institute of Technology. His research interests include signal detection and estimation.

