International Journal of Innovative Computing, Information and Control
Volume **9**, Number **4**, April **2013**
ICIC International ©2013 ISSN 1349-4198
pp. 1617–1625

A GRAPHICAL MODEL AND SEARCH ALGORITHM BASED QUASI-CYCLIC LOW-DENSITY PARITY-CHECK CODES SCHEME

Jingli Lin[1,2], Peng Shi[3,4], Gongjun Yan[2,5], Jun Wang[1] and Yong Huang[1]

[1]School of Electrical and Information Engineering
Xihua University
Chengdu 610039, P. R. China
jllin@uestc.edu.cn; { wangjun; huangyong }@mail.xhu.edu.cn

[2]Indiana University Kokomo
Kokomo, IN 46904, USA
goyan@iuk.edu

[3]School of Engineering and Science
Victoria University
Melbourne, Vic 8001, Australia
Peng.Shi@vu.edu.au

[4]School of Electrical and Electronic Engineering
The University of Adelaide
Adelaide, SA 5005, Australia

[5]University of Southern Indiana
Evansville, IN 47712, USA

Received December 2011; revised July 2012

Abstract. *A constant challenge in Quasi-Cyclic Low-Density Parity-Check (QC-LDPC) codes lies on flexible code lengths and rates, conditioned by good error performance. This paper presents an optimal construction method of QC-LDPC codes on the basis of graphical model and search algorithm. Utilizing the proposed scheme, we can construct adaptable QC-LDPC codes whose girths are no less than 12, and column weights are 3. Comparing with any other QC-LDPC code construction approach, this method is much more flexible in block length and block rate. Furthermore, when construction parameters are selected, regular QC-LDPC codes can be constructed easily. Simulation results indicate that LDPC codes proposed in the method perform similarly to the classic Progressive Edge-Growth (PEG) based LDPC codes but outweigh in convenience of implementation and applications.*
Keywords: Quasi-cyclic low-density parity-check (QC-LDPC) codes, Tanner graph, Girth, Iterative decoding

1. **Introduction.** The past decades have witnessed the increasing research interests in Low-Density Parity-Check (LDPC) codes [1]. LDPC codes have two important merits: 1) the performance can approach the Shannon limit. The longer code length, the better performance; and 2) decoding complexity is linear, which is much better than that of most other channel codes. LDPC codes can be classified into two categories approximately: 1) random or pseudorandom codes; and 2) structured codes. Generally, random codes outperform structured codes. However, the code length in real applications normally is not long, limited by the algorithm complexity. In this way, some well-structured LDPC codes perform as well as random ones. In addition, random LDPC codes need to store their complete parity-check matrices, which result in large storage volume, especially when

their code lengths are large. The large storage demands of LDPC codes are also one of the reasons that prevent their applications after these codes were discovered.

As one class of structured LDPC codes, QC-LDPC codes have very good error performance over noisy channels [2-5] and have linear relationship with their code length in the encoding complexity. In addition, a QC-LDPC code can be encoded/decoded with storing only part of its generator/check matrix because of the quasi-cyclic feature of QC-LDPC codes. Hence, QC-LDPC codes have been applied to some telecommunication systems, such as digital television terrestrial broadcasting system of China (CDTTB) [3].

In the bipartite graph corresponding to an LDPC code check matrix, its girth is one of the important factors of determining the performance of the LDPC code. Therefore, LDPC codes are generally expected to meet the RC constraint conditions (i.e., the bipartite graph has no cycle of 4). LDPC codes often adopt the belief propagation (BP) algorithm to decode. Smaller cycles of LDPC codes will result in autocorrelation of message passing, which will reduce decoding performance. Therefore, constructing LDPC codes without small cycles is highly recommended in applications and has been discussed in many references [6-17]. [6,7] present a classic random LDPC code construction algorithm named as Progressive Edge-Growth (PEG) algorithm which makes girth as large as possible. The methods in [8-10] are based on the PEG algorithm. The codes constructed in [8] outperform the ones constructed by the original PEG algorithm dramatically in fading channel. However, these codes are still random codes. The method in [9] partly expands the length of the cycles, but whether to enlarge the girth or not is not sure. The algorithm in [10] has large amounts of calculation. Except the PEG algorithm and its evolved construction algorithms, there are some other construction methods without small girth [11-17]. Some of them are to construct QC-LDPC codes [11,12,16,17]. The method given in [11] cannot be directly used to construct low rate and large girth QC-LDPC codes. The LDPC codes construction methods for girth of 6, 8, 10 and 16 are respectively proposed in [12-15]. We proposed a QC-LDPC code construction method [16] which can construct QC-LDPC codes with girth of 12. The fundamental idea of this method is to construct QC-LDPC codes using linear congruence theorem and graph theory. However, the dimensions of sub-matrices of these QC-LDPC codes are limited to prime numbers. Also, the method provided in [17] requires that the dimensions of sub-matrices should be prime numbers or multiple of prime numbers. Thus, the rates and lengths of QC-LDPC codes are constrained as well.

In fact, LDPC codes in communication systems are often with small or median code lengths. The practical LDPC codes are more likely to have small cycles distributed inside, comparing with LDPC codes with long code length. Most QC-LDPC construction methods require the dimensions of sub-matrices be prime numbers or multiple of prime numbers, which result in inflexibility in code lengths and code rates. In addition, regular LDPC codes are more preferred in practical implementations because they require much less storage than irregular ones. Therefore, we propose an adaptable QC-LDPC code construction method. An LDPC code constructed with this method has following features: 1) having girth no less than 12, 2) having flexible code length and code rate, 3) having better performance in the case of shorter code length, and 4) having column weight of 3 and easily being constructed in regular form.

This paper is organized as follows. Section 2 presents some backgrounds. Section 3 describes a method of constructing QC-LDPC codes, each of which has girth of 12 and flexible block length and block rate. In Section 4, the codes are analyzed. In Section 5, the simulation results are given. Finally, Section 6 concludes this paper.

2. QC-LDPC Codes and Structure Graph.

2.1. QC-LDPC codes.

2.1. QC-LDPC codes. Let P^i denote a circulant matrix of $L \times L$-dimension, where $0 \le i < L$. To make it simple, suppose P^∞ denote the zero matrix. Then, the matrix of an $mL \times nL$-dimension QC-LDPC code is:

$$H = \begin{bmatrix} P^{a_{0,0}} & P^{a_{0,1}} & \cdots & P^{a_{0,n-1}} \\ P^{a_{1,0}} & P^{a_{1,1}} & \cdots & P^{a_{1,n-1}} \\ \vdots & \vdots & \ddots & \vdots \\ P^{a_{m-1,0}} & P^{a_{m-1,1}} & \cdots & P^{a_{m-1,n-1}} \end{bmatrix}, \tag{1}$$

where $a_{i,j} \in \{0, 1, \cdots, L-1, \infty\}$. Respectively substitute each zero matrix and each cyclic shift matrix in H with "0" and "1" to obtain a matrix $B(H)$ of $m \times n$-dimension. We call the cycles in $B(H)$ as "cycle-B". In H, a cycle-B with length of $2l$ can be denoted as $P^{a_1} \to P^{a_2} \to \cdots \to P^{a_{2l}} \to P^{a_1}$, where $a_i \in \{0, 1, \cdots, L-1\}$, $1 \le i \le 2l$.

Suppose

$$s(a) = \left(\sum_{k=0}^{2l-1} (-1)^k a_{k+1}\right) (\mathrm{mod}\ L), \tag{2}$$

where x (mod B) is defined as $(B + x)$ mod B.

If β is the greatest common divisor of $s(a)$ and L, then a cycle in matrix $B(H)$ with length of $2l$ is corresponding to β cycles with length of $2lL/\beta$ in matrix H [18].

2.2. Mapping of structure graph and matrix. A structure graph G is a simple graph which is finite and trivial [19]. Therefore, G has no multiple edges. G can be correspondingly mapped to a Galois field GF (2) matrix H. Suppose the structure graph G has m paths, $p_0, p_1, \ldots, p_{m-1}$, and n vertexes, $v_0, v_1, \ldots, v_{n-1}$. Any two paths in the graph are either disjoint or singularly crossing. Path p_j is mapped into the jth row of H, i.e., r_j of H. Vertex v_i is mapped into the ith column of H, i.e., c_i of H. Let $V(S)$ denote the vertex set of S. If $v_i \in V(p_j)$ $(0 \le i < n,\ 0 \le j < m)$ (i.e., path p_j passes vertex v_i), then the element of the jth row and the ith column of matrix H is "1", otherwise is "0".

Therefore, the number of vertexes passed by each path is much less than the total of vertexes of graph G, that is, $V(p_j) \ll V(G)$, where $0 \le j < m$; then matrix H is a sparse matrix.

Hence, if m paths of a structure graph respectively pass ρ vertexes; n vertexes are respectively passed by λ paths, then the structure graph is correspondingly mapped into a regular matrix with column weight of λ and row weight of ρ.

Figure 1 shows an example of mapping between a structure graph and a matrix. The left is a structure graph, and the right is the corresponding matrix of the structure graph.

Additionally, in matrix H, let define row line which passes all "1" elements of the jth row of H $(0 \le j < m)$, and define column line which passes all "1" elements of the ith column of H $(0 \le i < n)$. Then t row lines and t column lines can form a cycle with length of $2t$ ($2 \le t \le \min(m, n)$, where the symbol $\min(\cdot)$ denotes the minimum value of the aggregate).

Paths in a structure graph can form a *p-cycle*. If there are totally t paths involved in the *p-cycle*, then the length of the *p-cycle* is defined as t. As indicated in the structure graph of Figure 1, bold lines denote a *p-cycle* with length of 3 formed by three paths p_0, p_2 and p_3. The *p-cycle* in the structure graph on the left is transformed to a cycle depicted in its corresponding matrix on the right. The size of the cycle in the matrix is 6.

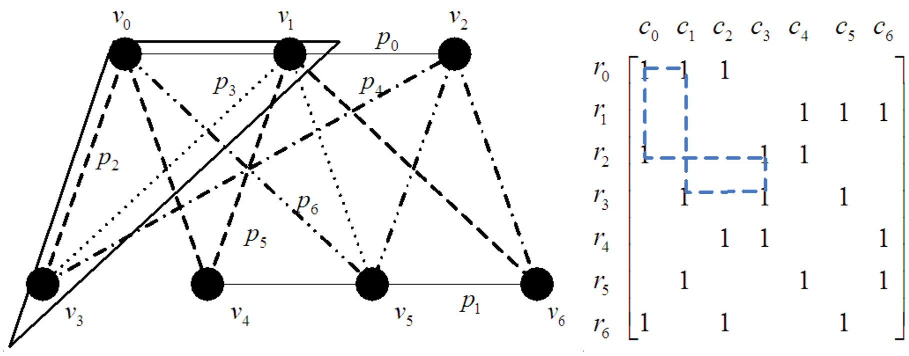

Figure 1. Example for mapping between structure graph and matrix

It is easy to prove that a *p-cycle* in structure graph G can be transformed to a cycle in matrix H, and the length of the *p-cycle* is only half of the length of the cycle, and vice versa.

3. Construction of QC-LDPC Codes with Girth of 12.

As for the check matrix of an LDPC code, if its column weight $\lambda \geq 3$, with its code length increasing, minimum distance of the code will increase linearly. If column weight is 2, minimum distance of code will only increase logarithmically [1]. Therefore, LDPC codes with column weight of 3 are preferred in the paper. In our method, first, we map some original matrixes into structural graphs. For the purpose of constructing QC-LDPC codes, the original matrixes are quasi-cyclic matrixes. We then add multiple paths in the structural graph and make sure two conditions: 1) every vertex in the structure graph is passed by 3 paths; 2) the smallest *p-cycle* of the structure graph is 6. In this way, we convert the structural graph back to a check matrix. The matrix has column weight of 3 and girth of 12. Figure 2 shows an example for a structure graph construction. Each original quasi-cyclic matrix maps to each layer in Figure 2, respectively. The submatrix $a_{l,i,j}$ of original quasi-cyclic matrix a_l maps to subgraph $g_{l,i,j}$, and paths are added to connect these subgraphs. The following steps are addressed to build GF(2) check matrix which has no small cycles:

1) Construct t small $m \cdot L \times n \cdot L$ original quasi-cyclic matrices in GF(2), each of which is marked as A_l ($2 \leq m, n \leq L$, $0 \leq t \leq L$, $0 \leq l < t$, L is the order of each submatrix of A_l), which do not include smaller cycles (for example each girth is 6). Because the matrices are small, the construction is relatively easy.

Each original matrix A_l can be obtained by searching L-tuple matrix $a_l = [a_{l,i,j}]$, where $a_{l,i,j} \in \{0, 1, \cdots, L-1\}$, $0 \leq i < m$, $0 \leq j < n$, $2 \leq m, n \leq L$. According to (2), $\forall 0 \leq i_1, i_2 < m$, $0 \leq j_1$, $j_2 < n$, $i_1 \neq i_2$, $j_1 \neq j_2$, when the girth of a original matrix A_l is 6, the following restriction condition should be met:

$$a_{l,i_1,j_1} - a_{l,i_1,j_2} + a_{l,i_2,j_2} - a_{l,i_2,j_1} \neq 0 (\text{mod } L). \tag{3}$$

2) Map each original matrix A_l into structure graph B_l. Aggregation of $B_0, B_1, \ldots, B_t$ form structure graph C. Each original matrix A_l is divided into $m \times n$ submatrices. Each submatrix is denoted as $m_{l,i,j}$. All the "1" elements in A_l are mapped into vertexes in structure graph C, marked as $v_{l,i,j,k}$, where k means that the vertex is mapped by the "1" element of the kth column of submatrix $m_{l,i,j}$; hence, $0 \leq k < L$, $0 \leq i < m$, $0 \leq j < n$, $0 \leq l < t$.

The row lines and the column lines of A_l are mapped into paths in B_l, named as row paths and column paths, respectively. Because each "1" element of A_l is respectively

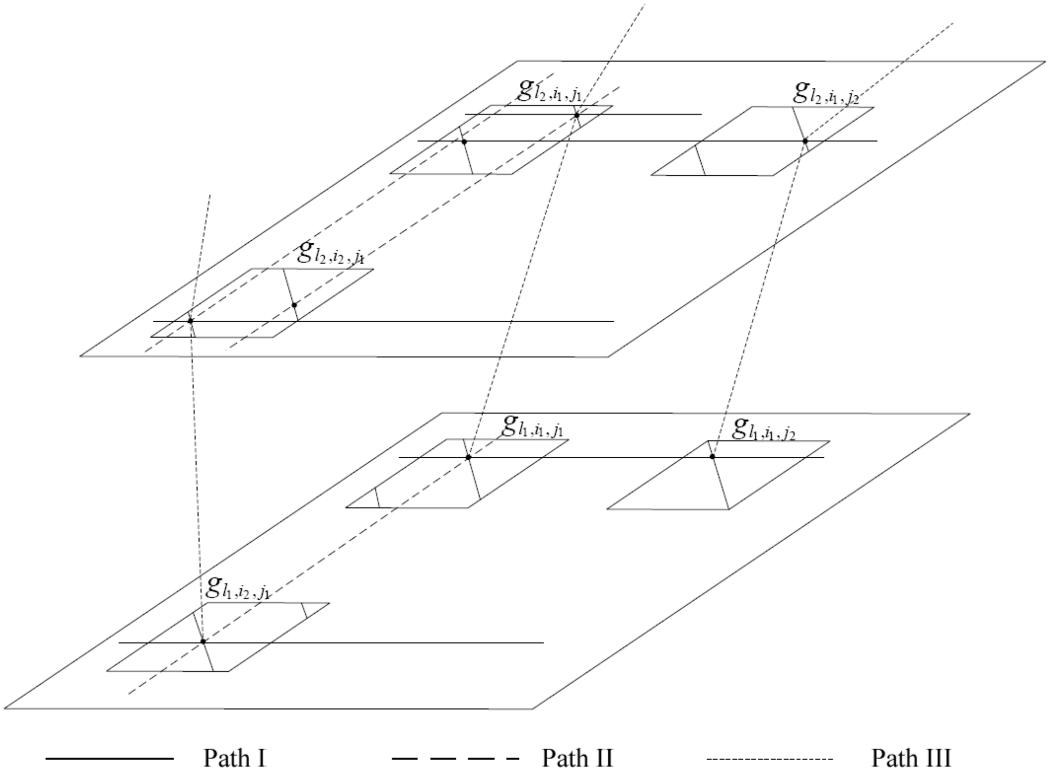

Figure 2. Structure graph construction

intersected by one row line and one column line, the vertexes in B_l are intersected by two paths. In this way, a cycle of length q in A_l is mapped into a *p-cycle* of length q in B_l.

In order to construct an LDPC code of column weight 3, each vertex in B_l should be passed by another path. Therefore, there are rules to add paths among structure graphs $B_0, B_1, \ldots, B_t$ to avoid that the length of each *p-circle* in C is small.

Each original matrix A_l is divided into $m \times n$ submatrixes, which are denoted as $m_{l,i,j}$, correspondingly, and each structure graph B_l is divided into $m \times n$ subgraphs, which are denoted as $g_{l,i,j}$, where $0 \le i < m$, $0 \le j < n$. Structure graph C is divided into $m \times n$ groups. Each group is composed of subgraphs $g_{0,i,j}$, $g_{1,i,j}$ and $\ldots g_{t-1,i,j}$. L twisty paths are added in each group, and each twisty path passes one vertex of each subgraph in the group.

Define twisty factor $\tau_{l,i}$ for each twisty path when the twisty path intersects subgraph $g_{l,i,j}$, where $0 \le \tau_{l,i} < L$, $0 \le l < t$. $\forall 0 \le i < m$, let $\tau_{0,i} = 0$. Each twisty path is added to follow the following rules:

The twisty path starts from $g_{0,i,j}$, vertex $v_{0,i,j,u}$ is connected to vertex $v_{l,i,j,(u+\tau_{l,i})(\bmod L)}$ when the twisty path intersects subgraph $g_{l,i,j}$, where $0 \le u < L$, $0 < l < t$, $0 \le i < m$, $0 \le j < n$.

According to (2), $\forall 0 \le l_1, l_2 < t$, $0 \le i_1, i_2 < m$, $0 \le j < n$, where $l_1 \ne l_2$, $i_1 \ne i_2$, in order to guarantee that any two column paths passing subgraphs $g_{l_1,i_1,j}$, $g_{l_1,i_2,j}$, $g_{l_2,i_1,j}$ and $g_{l_2,i_2,j}$ and any two twisty paths intersecting subgraphs $g_{l_1,i_1,j}$, $g_{l_1,i_2,j}$, $g_{l_2,i_1,j}$ and $g_{l_2,i_2,j}$ do not form a *p-cycle* with length of 4, the following two restrict conditions must be met:

$$\tau_{l_1,i_1} - \tau_{l_2,i_1} \ne \tau_{l_1,i_2} - \tau_{l_2,i_2} \tag{4}$$

$$\tau_{l,i_1} \ne \tau_{l,i_2}. \tag{5}$$

Also, in order to ensure that any two row paths passing subgraphs g_{l_1,i,j_1}, g_{l_1,i,j_2}, g_{l_2,i,j_1} and g_{l_2,i,j_2} $(1 \le l_1, l_2 \le L)$ and any two twisty paths intersecting subgraphs g_{l_1,i,j_1}, g_{l_1,i,j_2},

g_{l_2,i,j_1} and g_{l_2,i,j_2} $(1 \le l_1, l_2 \le L)$ do not form a *p-cycle* with length of 4, the following restrict condition must be met:

$$a_{l_1,i,j_1} - a_{l_1,i,j_2} + a_{l_2,i,j_2} - a_{l_2,i,j_1} \neq 0 (\bmod L). \tag{6}$$

3) Map structure graph C into parity check matrix D: Define each path in structure graph C as p_q. Row paths are formed by the row lines in structure graph B_l, $q = ((m+n)l + i)L + w$, where w denotes the w^{th} row of submatrix $m_{l,i,j}$ $(0 \le w < L)$. Column paths are formed by the column lines in B_l, $q = ((m+n)l + m + j)\,L + k$. As for twisty paths connecting each structure graph B_l, $q = ((m+n)t + in + j)\,L + k$, where k is denoted with the column label in $m_{0,i,j}$.

Map the path p_x in structure graph C as the xth row of check matrix D, and map the vertex v_y in C as the yth column of D, where $0 \le x < (tm+tn+mn)L$, $0 \le y < t \cdot m \cdot n \cdot L$. If $v_y \in V(p_x)$, then the element of the xth row and the yth column of the check matrix D is "1", otherwise is "0".

In this approach, given $2 \le m, n, t \le L$, we can construct QC-LDPC codes with column weights 3, and varied row weights: m, n or t. However, generally speaking, when weight of every column is the same, an LDPC code whose weight of each row is roughly equal can more likely approach the Shannon limit than an LDPC code whose weight of each row is arbitrary [20]. Therefore, when this approach is employed in constructing the LDPC codes, the values of m, n and t need to be chosen as close as possible under the prerequisite of design demands on rate and code length.

4. Analysis on Structural Matrix.

4.1. The lower bound of search complexity.

Suppose the L-tuple matrix of original matrix A_l searched is $m \times n$-dimension $(2 \le m, n \le L,\ 0 \le l < t)$. The value of factor $a_{l,i,j}$ $(0 < i < m,\ 0 < j < n$ and $0 \le l < t)$ needs search for $i \times j$ times to meet (3) at least, and $l \times j$ times to meet (6) at least.

When twisty factor $\tau_{l,i}$ is being searched $(1 < l < t,\ 0 < i < m)$, $l \times i$ time searches are required to meet (4) at least; i time searches are required to meet (5) at least.

Hence, search complexity is lower bounded as follows:

$$\begin{aligned} C &= t \cdot \sum_{j=1}^{n-1} \sum_{i=1}^{m-1} i \cdot j + m \cdot \sum_{j=1}^{n-1} \sum_{l=1}^{t-1} l \cdot j + \sum_{i=1}^{m-1} \sum_{l=2}^{t-1} l \cdot i + t \cdot \sum_{i=1}^{m-1} i \\ &= \frac{tmn(n-1)(m+t-2) + m(m-1)(t+2)(t-1)}{4}. \end{aligned} \tag{7}$$

Since the code length of Check matrix D is $t \cdot m \cdot n \cdot L$, the lower bound increases in a geometric ratio of increasing of the code length. Therefore, the method is preferred to construct QC-LDPC codes with short length.

Similarly, structure graph C with smallest *p-cycle* length of 8 or even larger length can also be searched according to (2). However, the search is conducted under the prerequisite that structure graph C has no *p-cycles* with length of 4. Therefore, the search complexity will be much larger.

4.2. Matrix rate.

Check matrix D is $(tm + tn + mn)L \times t \cdot m \cdot n \cdot L$-dimension. Each vertex of structure graph C is passed by a row path, a column path and a twisty path. The column weights of check matrix D are the same and equal to 3. The row weights can be varied among m, n or t. Therefore, the sequence distribution of check matrix D

is denoted as follows:

$$\lambda(x) = x^2 \tag{8}$$

$$\rho(x) = \frac{tm}{tm+tn+mn}x^{n-1} + \frac{tn}{tm+tn+mn}x^{m-1} + \frac{mn}{tm+tn+mn}x^{t-1}. \tag{9}$$

Hence, the rate of check matrix D is

$$\begin{aligned} R &= 1 - \frac{\int_0^1 \rho(x)dx}{\int_0^1 \lambda(x)dx} \\ &= \frac{t^2m^2(n-3) + t^2n^2(m-3) + n^2m^2(t-3)}{mnt(tm+tn+mn)}. \end{aligned} \tag{10}$$

When the codelength c and rate t of a QC-LDPC are given, the value of the parameters m, n, t and L can be solved by following equation set:

$$\begin{cases} c = t \cdot m \cdot n \cdot L \\ k = \frac{t^2m^2(n-3)+t^2n^2(m-3)+n^2m^2(t-3)}{mnt(tm+tn+mn)} \end{cases}. \tag{11}$$

Since the equation set is underdetermined, the value of m, n, t and L can be varied in multiple options.

When $m = n = t$, a regular matrix can be obtained: its row weight is m, its column weight is 3, and its rate is

$$R = \frac{m-3}{m}. \tag{12}$$

The check matrix constructed with this approach is not necessary full rank. Therefore, its rate can be slightly higher than the calculation results of (10).

5. **Simulation.**

In Figure 3, horizon axis is Signal to Noise Ratio (SNR) and vertical axis is Bit Error Rate (BER). The performance curve for a (1250, 500) QC-LDPC code constructed by the approach is presented in Figure 3. The LDPC code is generated under the circumstances $L = 10$, and $m = n = t = 5$. This is a regular LDPC code whose column weight is 3 and row weight is 5. The check matrix of this LDPC code has 15 redundant rows, so its rate is approximately 0.412. Its corresponding curve is marked as MGM in Figure 3.

The PEG algorithm is a classical random LDPC code construction method and was recognized the best one by MacKay [6-8]. For the purpose of comparison, the performance curve of a PEG LDPC code (1250, 500) is presented in Figure 3, which is marked as PEG. The girth of the LDPC code is 8, and the average length of its cycles is 9.99.

In addition, the performance curve of a random LDPC code which has the same length and rate is also shown in Figure 3 and marked as RAND.

BPSK transmission over an AWGNC and the BP algorithm are employed for all simulations, and the maximum iterations are set as 20.

As indicated by Figure 3, the LDPC code constructed with the approach described in the paper is better than the LDPC code constructed with PEG algorithm. The constructed random LDPC code has similar performance with these two codes in low SNR areas; however, error floor starts to appear when BER is close to 10^{-6}. Hence, both this approach and the PEG algorithm have low error floors. However, the PEG algorithm is to construct random LDPC codes. As mentioned at the beginning of the paper, random LDPC codes are inconvenient in practice because they require much bigger storage than structure LDPC codes do. Therefore, the LDPC codes constructed with this approach ensure both of performance and application convenience on the premise of flexible code length and rate.

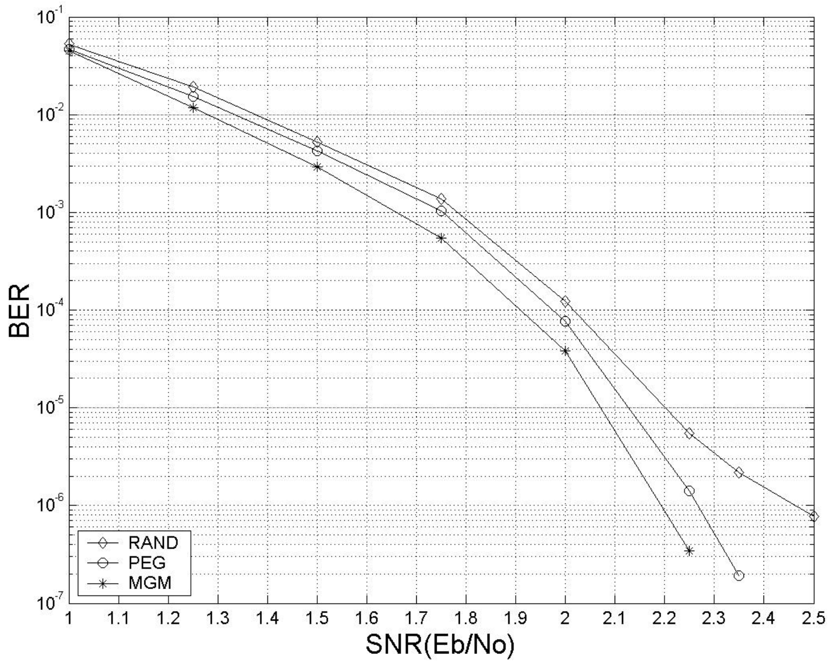

Figure 3. Comparison of (1250, 500) LDPC code performances

6. **Conclusions.** With the principles of constructing large girth LDPC codes, an approach of constructing QC-LDPC codes based on graphical model and search algorithm is put forward in the paper. Original matrixes of girth 6 are constructed firstly, and then they are mapped into structure graphs. Paths are added to these structure graphs to form a large structure graph. Finally, the large structure graph is mapped into a QC-LDPC code as required. The column weight of the QC-LDPC code is 3, and the girth is 12. Both the code lengths and code rates of QC-LDPC codes in this approach are very flexible. Furthermore, when construction parameters are selected, regular QC-LDPC codes can be constructed. These features all make the codes constructed be convenient in applications. Final simulation results also indicate that our QC-LDPC codes have similar performances with LDPC codes constructed with the classic PEG algorithm. The future work includes: 1) applications of the proposed scheme, 2) further simulation in test bed, 3) larger girth QC-LDPC construction algorithm with lower complexity.

Acknowledgement. The authors wish to thank the editor and reviewers for their very constructive comments and suggestions which have greatly helped improve the presentation of the paper. This work was partially supported by the National Key Basic Research Program, China (2012CB215202), the 111 Project (B12018), the National Natural Science Foundation of China (61174058, 61134001 and 11126333), Key Program of Sichuan Education Department (09ZA160), and Xihua University Key Program (Z0920912).

REFERENCES

[1] R. G. Gallager, Low-density parity-check codes, *IEEE Trans. Inform. Theory*, vol.8, no.1, pp.21-28, 1962.

[2] S. Myung, K. Yang and J. Kim, Quasi-cyclic LDPC codes for fast encoding, *IEEE Trans. Inform. Theory*, vol.51, no.8, pp.2894-2901, 2005.

[3] Framing structure, channel coding and modulation for digital television terrestrial broadcasting system, *R. P. China Standard Number: GB20600-2006*, 2006.

[4] Q. Huang, Q. Diao, S. Lin and K. Abdel-Ghaffar, Cyclic and quasi-cyclic LDPC codes on constrained parity-check matrices and their trapping sets, *IEEE Trans. Inform. Theory*, vol.58, no.5, pp.2648-2671, 2007.

[5] Q. Diao, Q. Huang, S. Lin and K. Abdel-Ghaffar, A matrix-theoretic approach for analyzing quasi-cyclic low-density parity-check codes, *IEEE Trans. Inform. Theory*, vol.58, no.6, pp.4030-4048, 2007.

[6] X. Y. Hu, E. Eleftheriou and D. M. Arnold, Regular and irregular progressive edge-growth tanner graphs, *IEEE Trans. Commun.*, vol.51, no.1, pp.386-388, 2005.

[7] X. Y. Hu, *Source Code for Progressive Edge Growth Parity-Check Matrix Construction*, http://www.inference.phy.cam.ac.uk/mackay/PEG_ECC.html.

[8] A. G. D. Uchoa, C. Healy, R. C. de Lamare and R. D. Souza, Design of LDPC codes based on progressive edge growth techniques for block fading channels, *IEEE Communications Letters*, vol.15, no.11, pp.1221-1223, 2011.

[9] X. Liu and H. Cheng, Study on the construction method of quasi-cyclic LDPC codes based on progressive edge-growth (PEG) algorithm, *Journal of Circuits and Systems*, vol.14, no.4, pp.115-119, 2009.

[10] L. Huang, Y. Wang and P. Gong, An improved construction method of QC-LDPC codes based on the PEG algorithm, *Proc. of IEEE PACCS*, pp.1-4, 2001.

[11] Y. Liu, X. Wang, R. Chen and Y. He, Generalized combining method for design of quasi-cyclic LDPC codes, *IEEE Commun. Lett.*, vol.12, pp.392-394, 2008.

[12] K. Lally, Explicit construction of Type-II QC LDPC codes with girth at least 6, *Proc. of IEEE International Symposium on Information Theory*, pp.2371-2375, 2007.

[13] J. Craddock, M. Flanagan, S. J. Redmond et al., Construction of girth 8 LDPC codes based on multidimensional finite lattices, *Proc. of the 12th IEEE Symposium on Computers and Communications*, pp.655-659, 2007.

[14] F. Zhang, X. Mao, W. Zhou and H. D. Pfister, Girth-10 LDPC codes based on 3-D cyclic lattices, *IEEE Trans. Vehicular Technology*, vol.57, no.2, pp.1049-1060, 2008.

[15] I. B. Djordjevic, L. Xu Lei, T. Wang et al., Large girth low-density parity-check codes for long-haul high-speed optical communications, *Conference on Optical Fiber Communication/National Fiber Optic Engineers*, pp.1-3, 2008.

[16] X. Jiang and M. H. Lee, Large girth quasi-cyclic LDPC codes based on the Chinese remainder theorem, *IEEE Communication Letters*, vol.13, no.5, pp.342-344, 2009.

[17] L. J. Jing, J. L. Lin and W. L. Zhu, Design of quasi-cyclic low-density parity-check codes with large girth, *ERTI Journal*, vol.29, no.3, pp.381-389, 2007.

[18] K. S. Kim, S. H. Lee, Y. H. Kim et al., Design of binary LDPC code using cyclic shift matrices, *Electronics Letters*, vol.40, no.5, pp.325-326, 2004.

[19] X. Zhang and Z. Li, *Graph Theory and Its Applications*, HEP, Beijing, 2004.

[20] D. J. C. MacKay, Good error correcting codes based on very sparse matrices, *IEEE Trans. Inform. Theory*, vol.45, no.2, pp.399-431, 1999.