



Trends in
**Applied Sciences
Research**

ISSN 1819-3579



Academic
Journals Inc.

www.academicjournals.com

Calculation of Girth of Tanner Graph in LDPC Codes

¹H. Dehghani, ¹M. Ahmadi, ²S. Alikhani and ³R. Hasni

¹Department of Mathematics, Malek-Ashtar University of Technology, Shahin Shahr, Isfahan, 83145/115, Iran

²Department of Mathematics, Yazd University, 89195-741, Yazd, Iran

³Department of Mathematics, Faculty of Science and Technology, Universiti Malaysia Terengganu, 21030 Kuala Terengganu, Malaysia

Corresponding Author: M. Ahmadi, Department of Mathematics, Malek-Ashtar University of Technology, Shahin Shahr, Isfahan, 83145/115, Iran

ABSTRACT

LDPC codes can be described by a bipartite graph called Tanner graph. The girth g of the code is the length of the shortest cycle in its Tanner graph. We use the parity check matrix of code to present a method for existence of cycle of length girth g (i.e., g -cycle). This method give the number of cycles of g -cycles in the Low-density Parity Check (LDPC) codes.

Key words: LDPC code, parity check matrix, girth

INTRODUCTION

Low-density Parity-Check (LDPC) codes were introduced by Gallager (1962) and Gallager (1963). LDPC codes have a excellent performance with iterative decoding that is very close to the Shannon limit over Additive White Gaussian Noise (AWGN) channels (Kou *et al.*, 2001; Lucas *et al.*, 2000; MacKay and Neal, 1996; MacKay, 1999; Tanner, 1981). LDPC codes can be showed by a bipartite graph called Tanner graph (Tanner 1981). The Tanner graph is composed of two sets of nodes, that these namely, variable nodes and check nodes. Each variable nodes and check nodes correspond to the number of codeword symbols and parity symbols, respectively. If a variable node is constrained by a check node, there is an edge connecting these two nodes.

The girth of the code is the length of the shortest cycle in its Tanner graph and it is a crucial parameter.

Cycles, especially short cycles, leave a bad effect on the performance of LDPC decoders. Because they affect the independence of the extrinsic information which exchanged in iterative decoding (Sipser and Spielman, 1996; Wiberg, 1996). Several methods of counting the number of cycles have been presented by Fan and Xiao (2006) and Halford and Chugg (2004). In this paper a method is presented for counting the number of cycles with length g . In this method the parity-check matrix must be $(g-2)$ -cycle free (i.e. don't have any cycle of length $(g-2)$). Then we analyze the figures of 4-cycles, 6-cycles and in generally g -cycles in check matrix. Also we introduce algorithm for existence of g -cycle and computing the number of cycles with length g .

CYCLES OF LENGTH g AND ITS FIGURE

A LDPC code can be shown by a parity-check matrix. The nonzero components are too lesser than zero components in parity-check matrix. A cyclein a tanner graph refers to a finite set of

connected edges that starts and ends at the same node and satisfies the condition that no node (except the initial and final node) appears more than once. The length of the cycle is the number of it edges. The length of a smallest cycle in the graph is called the girth of the graph.

We know that four ones in 4-cycle belong to 2 rows and 2 columns equally in check matrix. So the figure of 4-cycle as shown in Fig. 1.

Consider H as a $m \times n$ check matrix. Figure 1 shows that 4-cycle must be in two rows. So we can calculate the number 4-cycles in all possible combinations of two rows in check matrix and add them (i.e. the total number of 4-cycles in check matrix). The number of 2-column rows combinations is (Fan and Xiao, 2006):

$$w = C_m^2 = \frac{m(m-1)}{2} \tag{1}$$

There are six number ones in 6-cycles belong equally to 3 rows and 3 columns. So, we can get $p_3^6 = 6$ number of 6-cycles with different figures. 6-cycles in parity-check matrix shown in Fig. 2.

If H is a $m \times n$ check matrix. Figure 2 shows that a 6-cycle must be in 3 rows. So, we can calculate the number of 6-cycles in all possible combinations of 3 rows in check matrix and so the

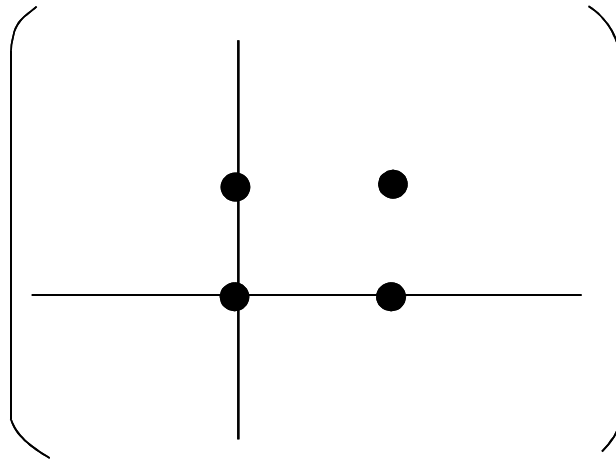


Fig. 1: 4-Cycle in parity-check matrix

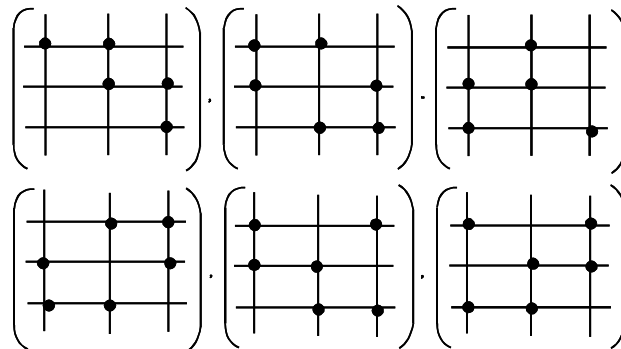


Fig. 2: 6-Cycles in parity-check matrix

total number of 6-cycles of check matrix is sum of the number of 6 cycles in all possibilities. The number of 3-rows combinations w is (Fan and Xiao, 2006):

$$w = C_m^3 = \frac{m(m-1)(m-2)}{6} \tag{2}$$

In generally g ones in g -cycle belong equally to $g/2$ rows and $g/2$ columns of check matrix. We can calculate the number of g -cycles in all possible combinations of $g/2$ in check matrix. Therefore to obtain the total number of g -cycles in check matrix we add the number of g cycles in all cases. The number of $g/2$ -rows combinations w is:

$$w = C_m^{\frac{g}{2}} = \frac{m(m-1)\dots(m-(\frac{g}{2}-1))}{\frac{g}{2}!} \tag{3}$$

ALGORITHM FOR CALCULATING OF THE GIRTH AND THE NUMBER OF CYCLES OF LENGTH GIRTH

Here, a method for calculating of the girth and its number in parity-check matrix is presented. First, we describe this method for 4-cycle and 6-cycle and then we describe it in general.

As we observed in previous section, a t_i -cycle is formed with four ones of two rows. By adding these two rows, the components of the new vector are 0, 1 or 2. If the number of 2 in the new row is t_i , where $1 \leq i \leq w$ and $t_i \geq 2$, then there is at least one 4-cycle in parity-check matrix and the number of 4-cycle is:

$$\sum_{i=1}^w C_{t_i}^2 = \sum_{i=1}^w \frac{t_i(t_i-1)}{2} \tag{4}$$

The following is algorithm of 4-cycle.

Check Algorithm of 4-Cycles

```

for  $i_1 = 1: m$ 
for  $i_2 = (i_1+1): m$ 
 $h = H(i_1,:) + H(i_2,:)$ 
If the number of 2 in  $h$  is  $t$  as  $t \geq 2$  There is  $C_t^2$  4-cycle in this combination.
end
end

```

Now we give an example for algorithm of 4-cycles.

Example 1: Consider the parity-check matrix H as following:

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

If we add the first and second rows of this matrix, we get the vector (2 1 1 2 0). Since there are two entry 2 in this vector, so there is at least a 4-cycle. Therefore in this matrix there is a 4-cycle

which is obtained from the first and the second rows and the first and the fourth columns. The two 4-cycle in this matrix are obtained from rows 1,2 and 2,3.

Now, we investigate check algorithm of 6-cycles. First we check the algorithm of 4-cycles for matrix, if there isn't any 4-cycle (4-cycle free), then we check the 6-cycle and the number of 6-cycles in matrix. The figures of 6-cycle were shown in Fig. 2. By adding the any three rows in check matrix, we get a vector with possible entries 0, 1, 2 or 3. Then there is a 6-cycle in parity-check matrix. The number of 6-cycles in parity-check matrix which is 4-cycle free is $\sum_{i=1}^m t_i$, where t_i is defined as following:

$$t_i = \begin{cases} 1 & \text{if there are three entry 2 in } i\text{-th combination} \\ 0 & \text{o.w} \end{cases} \quad (5)$$

The following algorithm is check algorithm of g-cycle for matrix.

Check Algorithm of 6-cycles

```

for i1 = 1: m
for i2 = (i1+1): m
for i3 = (i2+1): m
h = H(i1,:)+H(i2,:)+H(i3,:)
If the number of 2 in h is 3- There is a 6-cycle in this combination
end
end
end
    
```

Example 2: Consider a 4-cycle free check matrix H as:

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

The number of 6-cycles in this matrix is 28. For example, by adding the first, second and forth rows we have a vector (2 2 2 0 1 1 1) as shown in the following image.

$$\begin{matrix} 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 0 & 1 \\ 2 & 2 & 2 & 0 & 1 & 1 & 1 \end{matrix}$$

There are three entry 2 in this vector. So there is a 6-cycle in matrix H. By checking all possible combination of three rows in this matrix, the number of 6-cycles is 28.

In this subsection we present algorithm of g-cycles in check matrix. First, we check the algorithms of g'-cycle for check matrix, where g' < g. If this matrix is (g-2)-cycle free, then we run the check algorithm of g-cycle as following:

By adding the g/2 rows of check matrix, we get the vector that every entry is a nonnegative integer number belong to:

$$\left[0, \frac{g}{2} \right]$$

If the number of 2 in the new vector obtained from adding rows is, g/2 then there is a g-cycle in this row-combination. The number of g-cycle in (g-2)-cycle free parity-check matrix is equal to $\sum_{i=1}^w t_i$, where t_i is defined as following:

$$t_i = \begin{cases} 1 & \text{if there are } \frac{g}{2} \text{ entry 2 in } i\text{th combination} \\ 0 & \text{o.w.} \end{cases} \quad (6)$$

Check Algorithm of g-Cycles

```

for i1 = 1: m
for i2 = (i1+1): m
.....
for ig = (ig - 1 + 1): m
h = H(i1,:) + H(i2,:) + H(ig,:)
If the number of 2 in h is  $\frac{g}{2}$  There is a g-cycle in this combination
end
.....
end
end
end

```

RESULT SIMULATION AND AFFECT SHORT CYCLES IN PERFORMANCE LDPC CODES

In this paper, we presented a method of detecting the g-cycles in parity-check matrix. Cycles, especially short cycles, degrade the performance of LDPC decoders, so it is requirement that remove the short cycles in parity-check matrix. Since we like to grow the girth of check matrix by extending the binary field to F_q (i.e., by replacing the element field in check matrix we can extending the binary field to F_q), we need to find the short cycles with length girth. By finding these cycles and set the field member in it, we can grow the girth and prevent from forming the cycles with length g. Then the new obtained check matrix is free of g-cycle. So by applying this method, we can grow the girth of LDPC code. BER performance of a non binary LDPC code with girth 6 and 8 over AWGN channel, BPSK modulation and 20 repeat simulated as shown in Fig. 3. The parity-check matrix of this code is an 42×14 matrix and the number of cycles length girth (g = 6) equal to 1225. Using this method we can find the 6-cycles and try set the field elements in check matrix such that the new parity-check matrix free of 6-cycle. The simulation results show that this code has better performance than random Mackay and Nil codes in g = 8.

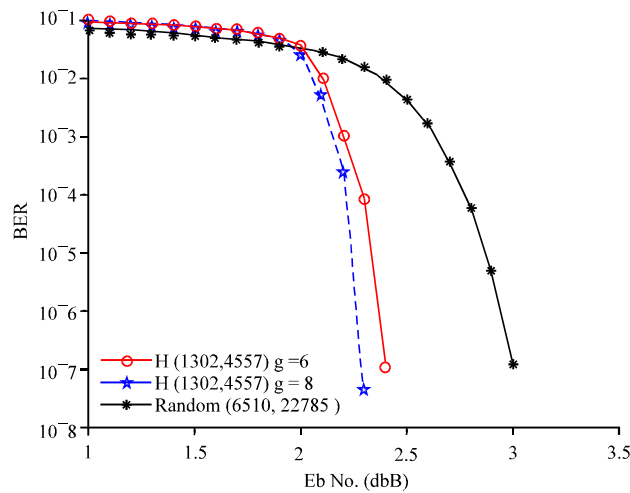


Fig. 3: The simulation results to find better performance

CONCLUSION

In this paper, a method for counting the number of cycles with length g is presented. Also we analyzed the figures of 4-cycles, 6-cycles and in generally g -cycles in check matrix. We introduced algorithm for existence of g -cycle and computing the number of cycles with length g .

ACKNOWLEDGMENT

The authors would like to thank the referee for his/her helpful comments.

REFERENCES

- Fan, J. and Y. Xiao, 2006. A method of counting the number of cycles in LDPC codes. Proceedings of the International Conference on Signal Processing, Volume 3, August 16-20, 2006, Beijing.
- Gallager, R., 1962. Low-density parity-check codes. IRE Trans. Inform. Theory, 8: 21-28.
- Gallager, R.G., 1963. Low-Density Parity-Check Codes. MIT Press, Cambridge, MA., USA., ISBN-13: 9780262070072, Pages: 102.
- Halford, T.R. and K.M. Chugg, 2004. Enumerating and counting cycles in bipartite graphs. Proceedings of the Workshop on Communication Theory, May 5-8, 2004, Capri, Italy.
- Kou, Y., S. Lin and M.P.C. Fossorier, 2001. Low-density parity-check codes based on finite geometries: A rediscovery and new results. IEEE Trans. Inform. Theory, 47: 2711-2736.
- Lucas, R., M.P.C. Fossorier, Y. Kou and S. Lin, 2000. Iterative decoding of one step majority logic decodable codes based on belief propagation. IEEE Trans. Commun., 48: 931-937.
- Mackay, D.J.C. and R.M. Neal, 1996. Near Shannon limit performance of low-density parity-check codes. Electron. Lett., 32: 1645-1646.
- Mackay, D.J.C., 1999. Good error-correcting codes based on very sparse matrices. IEEE Trans. Inform. Theory, 45: 399-431.
- Sipser, M. and D. Spielman, 1996. Expander codes. IEEE Trans. Inform. Theory, 42: 1710-1722.
- Tanner, R., 1981. A recursive approach to low complexity codes. IEEE Trans. Inform. Theory, 27: 533-547.
- Wiberg, N., 1996. Codes and decoding on general graphs. Thesis Ph.D. Linkoping University, Linkoping, Sweden.