Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2007, Article ID 48158, 5 pages
doi:10.1155/2007/48158

Research Article

High Girth Column-Weight-Two LDPC Codes Based on Distance Graphs

Gabofetswe Malema and Michael Liebelt

School of Electrical and Electronic Engineering, The University of Adelaide, North Terrace, Adelaide 5005, SA, Australia

Received 12 November 2005; Revised 2 September 2006; Accepted 25 October 2006

Recommended by Wolfgang Gerstacker

LDPC codes of column weight of two are constructed from minimal distance graphs or cages. Distance graphs are used to represent LDPC code matrices such that graph vertices that represent rows and edges are columns. The conversion of a distance graph into matrix form produces an adjacency matrix with column weight of two and girth double that of the graph. The number of 1's in each row (row weight) is equal to the degree of the corresponding vertex. By constructing graphs with different vertex degrees, we can vary the rate of corresponding LDPC code matrices. Cage graphs are used as examples of distance graphs to design codes with different girths and rates. Performance of obtained codes depends on girth and structure of the corresponding distance graphs.

1. INTRODUCTION

Low-density parity-check (LDPC) codes have been shown to have very good error correcting capability performing close to the Shannon limit [1]. It was also shown in [2] that LDPC codes with column weight $j \geq 3$ have a minimum distance that grows linearly with code length, N, for given j and row-weight k and that minimum distance for codes with $j = 2$ grows logarithmically with N. However, column-weight-two codes have shown potential in partial response channels [3, 4]. They also require less computational complexity compared to codes of higher column weights. Performance of LDPC codes depends on several factors including minimum distance, rate, diameter of graph, code length, and girth (minimum cycle length). There are several methods of constructing column-weight-two LDPC codes, some of which are found in [3–5]. We propose a method for constructing column-weight-two codes with very large girths.

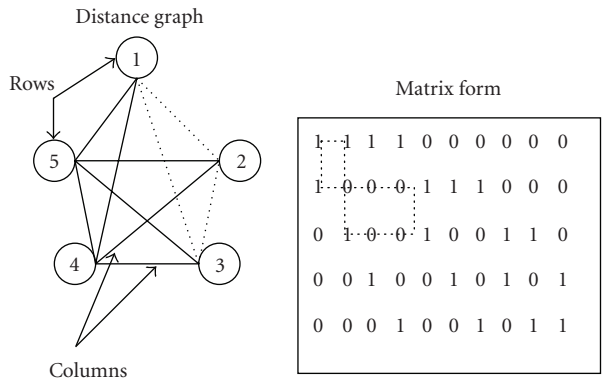
In this paper, we show how column-weight-two LDPC codes can be derived from distance graphs called cages (graphs with minimum number of vertices given vertex degree and minimum cycle length in graph). Using already known cage graphs, very high girths codes are obtained compared to previous methods. Cage graphs of varying girths and valencies (vertex degrees) could be used to construct LDPC codes over a wide range of girths and rates.

This paper is organized as follows. Section 2 describes LDPC code representation using a nonbipartite graph or row connections. The nonbipartite graphs are in the form of distance graphs. Section 3 presents some examples of distance graphs found in literature. Bit error rate performances of some codes were simulated and evaluated. Hardware implementation complexity of these codes is also discussed. Section 4 has concluding remarks.

2. GRAPH REPRESENTATION OF LDPC CODES

An $M \times N$ LDPC code matrix H is usually represented by a bipartite or Tanner graph on which one set of vertices represents rows (check nodes) and the other set represent columns (variable nodes). There is an edge between the vertices representing check node c_x and variable node v_y if and only if the corresponding element of the code matrix h_{xy} is set to 1.

The code matrix could also be represented by a distance graph in which vertices represent matrix rows and edges represent columns. In this representation, there will be an edge joining the vertices representing check nodes c_x and c_y if and only if there exists a column z such that $h_{xz} = h_{yz} = 1$. Thus a column is represented by a set of edges forming a complete graph between check node vertices that are connected to the same variable node. In the case of two rows per column (column weight of two), a complete graph between two vertices is a single edge between the vertices. We could therefore use



(a) Distance graph (b) Matrix form

Figure 1: LDPC matrix derived from a distance graph.

distance graphs to form LDPC codes of column-weight-two with each edge representing a column.

Figure 1(a) shows a distance graph of five vertices. Taking each vertex as a row and each edge as a column, a corresponding matrix is formed as shown in part (b) of the figure. The matrix is formed by putting a "1" in each of the rows (vertices) that form a column (edge). Vertices 1 to 5 of the graph correspond to rows 1 to 5 in the matrix, respectively. The minimum cycle length in the graph is three. A cycle is formed by a closed path of edges or vertices. A cycle in the graph is formed by moving between vertices or edges. In an LDPC matrix, a cycle is formed by alternatively moving between rows and columns. The graph form represents half of a matrix cycle. Hence, a cycle of g in the graph is of length $2g$ in matrix form. A cycle of three, shown in dotted lines in Figure 1, between vertices 1, 2, and 3 in the graph corresponds to a cycle of six between rows 1, 2, 3, and columns 1, 2, 5 in matrix form.

The main objective of this paper is to use known distance graphs to construct column-weight-two LDPC codes. Distance graphs of varying vertex degrees and girths produce column-weight-two LDPC codes of different girths and rates. For a graph of size n, vertex degree of k, and girth of g, the corresponding LDPC code matrix is of size $n \times nk/2$, rate $1 - (2/k)$, and girth of $2g$.

3. CAGES

A (k,g)-cage is a k-regular graph of girth g with the fewest possible number of vertices. The lower bound (Moore bound) on the number of vertices for a given k and n depends on whether g is odd or even [6, 7].

If g is odd then

$$n(k,g) = 1 + k + k(k-1) + \cdots + k(k-1)^{(g-3)/2} \quad (1)$$

and if g is even, then

$$n(k,g) = 1 + k + k(k-1) + \cdots + k(k-1)^{g/2} - 2 + (k-1)^{g/2} - 1. \quad (2)$$

Table 1: Sizes of some known cubic cages.

(k,g)	$n(k,g)$	Code size $\left(n \times \frac{nk}{2}\right)$	Girth $(2g)$
$(3,8)$	30	30×45	16
$(3,9)$	58	58×87	18
$(3,10)$	70	70×105	20
$(3,11)$	112	112×168	22
$(3,12)$	126	126×189	24
$(3,13)$	272	272×408	26
$(3,14)$	384	384×576	28
$(3,15)$	620	620×930	30
$(3,16)$	960	960×1440	32
$(3,17)$	2176	2176×3264	34
$(3,18)$	2640	2640×3960	36
$(3,19)$	4324	4324×6486	38
$(3,20)$	6048	6048×9072	40
$(3,21)$	16028	16028×24042	42
$(3,22)$	16206	16206×24309	44

However, these bounds are met very infrequently [8]. Though there is no uniform approach to constructing arbitrary cages, there are many cages constructed for some vertex degrees and girths. The mathematics behind the construction of cages is beyond the scope of this paper. Examples of cage graphs and construction methods could be found in cited references in this paper. In [9], some methods of generating regular graphs and cages are described. There is also an associated software by the same author at [10] that generates cages.

3.1. Cubic cages

Cages with vertex degree of three are called cubic cages. Table 1 shows the number of vertices for some of the known cubic cages obtained from [11]. Cubic cages construction methods could be found in [6, 7, 11, 12]. These graphs produce an adjacency matrix with girth twice the corresponding graph girth, column weight of two and rate 1/3.

3.2. Cages of higher vertex degrees

Cages of higher degrees are harder to construct [8]. However, there are many examples of these cages in literature and some construction algorithms [9, 13]. Table 2 shows the number of vertices for some of the known high vertex degree cages [11, 13]. Corresponding code matrices have girths of $2g$ and have higher rates but smaller girths compared to cubic cages. Higher vertex degrees increase data transmission rate with some degradation in decoding performance.

Table 2: Some of known cages' graphs with vertex degree higher than three.

(k,g)	$n(k,g)$	Code size $\left(n \times \frac{nk}{2}\right)$	Girth $(2g)$
$(4,9)$	275	275×550	18
$(4,10)$	384	384×768	20
$(5,7)$	152	152×380	14
$(5,8)$	170	170×425	16
$(6,7)$	294	294×882	14
$(6,8)$	312	312×936	16
$(7,5)$	50	50×175	10
$(7,6)$	90	90×315	12
$(8,8)$	800	800×3200	16
$(9,8)$	1170	1170×5265	16
$(10,8)$	1640	1640×8200	16
$(12,8)$	2928	2928×17568	16
$(14,8)$	4760	4760×33320	16

3.3. Related studies

Cyclic column-weight-two codes of girth 12 were constructed in [4]. The size of the code is given by $n = k(k^2 - k + 1)$ where k is the row weight and $k - 1$ is a prime. In [5], two codes with girth 16 and 20 of rates 1/2 and 1/3, respectively, are constructed form graphical models. Both codes have size over 4000. From cage graphs with sizes shown in Tables 1 and 2, LDPC codes with much high girths could be constructed. The graphs would also produce codes of higher rates. We noted that codes constructed in [4] have the same size as that of $(k,6)$-cage graphs. However, using known cages, more codes could be constructed even when $k - 1$ is not prime.

3.4. Performance simulations

Some cage graphs result in too small codes for practical use. An expansion method is therefore needed to get larger codes. We suggest the expansion method used in [14]. The code obtained from a cage graph can be used as a base matrix. Each "0" entry in the matrix is replaced by a $p \times p$ zero submatrix and each "1" entry is replaced by a shifted $p \times p$ identity submatrix. The expanded code is larger than the base matrix by a factor of p and has girth at least that of the base matrix. Using shifted identity submatrices simplifies addressing in hardware implementation. Obtained codes were expanded using the described method in our experiments. The expansion factors (p) are shown in brackets in the performance graphs.

Decoding performances of obtained codes was measured using bit-error rate (BER) simulations on AWGN channel with BPSK modulation. Obtained codes show good BER performances approaching BER of 10^{-6} between 5 and 6 dB for some codes. Figure 2 shows performance curves for codes derived from $(3,17)$, $(4,9)$, and $(4,10)$ cage graphs. The codes from $(4,9)$ and $(4,10)$ cages are expanded by 8 and 4, respectively. The codes perform better than a random code free of four cycles. The $(3,17)$-cage LDPC code has the best performance which could be attributed to its large girth of 34. Figure 3 shows performances of codes with higher rates derived from a family of $(k,5)$ cages. Both codes are expanded by a factor of 2. The code from the $(12,5)$ cage performs better than that from the $(11,5)$ cage and a random code of about the same size free of four cycles. Though the $(11,5)$ cage code has lower rate and same girth as the $(12,5)$-cage, its performance is the worst. Performance differences between $(12,5)$ and $(11,5)$ cages may be attributed to structural differences of the graphs.

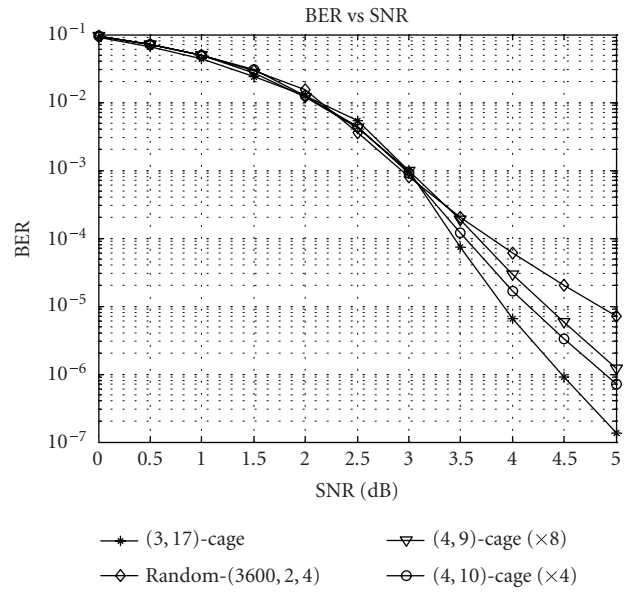


Figure 2: BER performances of high girth LDPC codes from distance graphs (25 iterations).

3.5. Hardware implementation

Codes obtained from cage graphs have low implementation complexity in that they are structured and have only two entries per column. However, not all structured codes are easily implementable. It is therefore important to study the structure of each graph to best exploit it for implementation. Cage graphs differ in construction methods and structure. Figure 4 shows a $(6,4)$ cage graph from which we derive a $(36,2,6)$ LDPC code with girth of eight, where $(N = 36, j = 2, k = 6)$. An odd vertex is connected to all even vertices and an even vertex to all odd vertices. In fact, all $(k,4)$ cage graphs are formed this way with $n = 2k$. The columns of the code matrix could be arranged as shown in Figure 5. In this matrix, connections are arranged cyclically such that the matrix comprises of 6×6 or $k \times k$ shifted identity submatrices. We could thus group the matrix rows in two groups of six (or k)

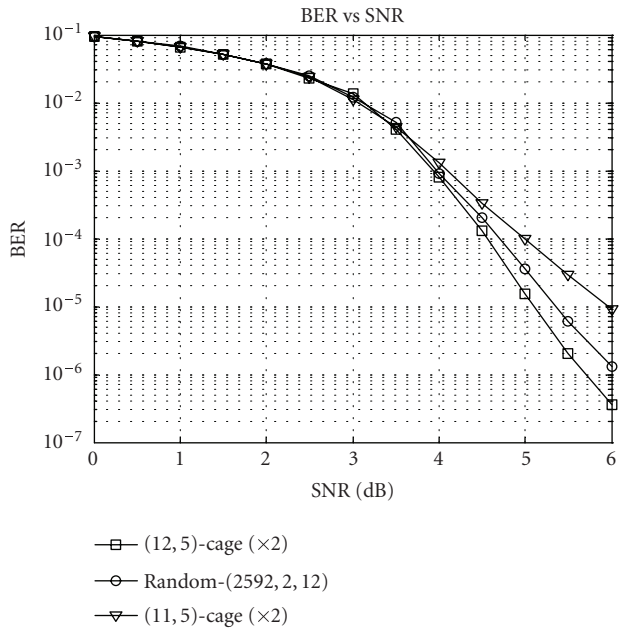


Figure 3: BER performances of $(k, 5)$ cage codes (25 iterations).

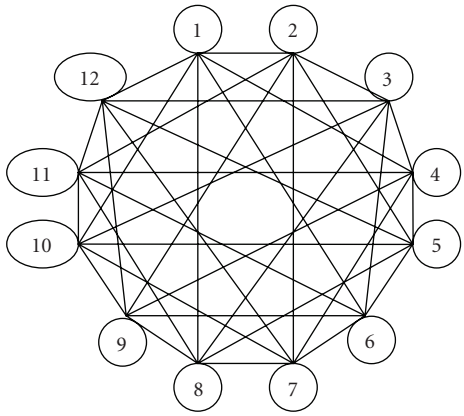


Figure 4: A $(6, 4)$ cage graph.

and columns into six (or k) groups of six (or k). The grouping reduces the interconnect complexity between processing nodes. There are fewer groups than individual columns or rows. The number of interconnections and destinations is reduced. Addressing within a group is also simplified. With one known row-column connection in submatrix, the rest of the submatrix connections could be deduced.

Column-weight-two codes also have reduced number of elements in the code matrix. This results in less computations and memory requirements. The variable node (column) computation involves the summation of the incoming messages and the channel estimate of the information bit. With two incoming messages, the computation is reduced to exchanging incoming messages and adding them to the channel estimation before sending them as outgoing messages.

100000	100000	100000	100000	100000	100000
010000	010000	010000	010000	010000	010000
001000	001000	001000	001000	001000	001000
000100	000100	000100	000100	000100	000100
000010	000010	000010	000010	000010	000010
000001	000001	000001	000001	000001	000001

100000	000001	000010	000100	001000	010000
010000	100000	000001	000010	000100	001000
001000	010000	100000	000001	000010	000100
000100	001000	010000	100000	000001	000010
000010	000100	001000	010000	100000	000001
000001	000010	000100	001000	010000	100000

Figure 5: Matrix representation of a $(6, 4)$ cage graph.

4. CONCLUSIONS

An approach for constructing LDPC codes with column weight of two has been described. Cage graphs are used to represent the code matrix, where vertices are rows and edges are columns. From known cage graphs, codes with very high girths and rates could be constructed. Some derived codes have good bit error rates compared to random codes. However, performance of each code depends on the structure of individual cage graphs from which the codes are derived.

REFERENCES

[1] S.-Y. Chung, G. D. Forney Jr., T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, 2001.

[2] R. G. Gallager, "Low-density parity-check codes," *IRE Transactions on Information Theory*, vol. 8, no. 1, pp. 21–28, 1962.

[3] H. Song, J. Liu, and B. V. K. Vijaya Kumar, "Low complexity LDPC codes for partial response channels," in *Proceedings of IEEE Global Telecommunications Conference (GLOBECOM '02)*, vol. 2, pp. 1294–1299, Taipei, Taiwan, November 2002.

[4] H. Song, J. Liu, and B. V. K. Vijaya Kumar, "Large girth cycle codes for partial response channels," *IEEE Transactions on Magnetics*, vol. 40, no. 4, part 2, pp. 3084–3086, 2004.

[5] J. M. F. Moura, J. Lu, and H. Zhang, "Structured low-density parity-check codes," *IEEE Signal Processing Magazine*, vol. 21, no. 1, pp. 42–55, 2004.

[6] N. Biggs, "Cubic graphs with large girth," in *Processdings of the 3rd International Conference on Combinatorial Mathematics*, pp. 56–62, New York, NY, USA, June 1989.

[7] G. Exoo, "A simple method for constructing small cubic graphs of girths 14, 15, and 16," *Electronic Journal of Combinatorics*, vol. 3, no. 1, pp. 1–3, 1996.

[8] P. Wong, "Cages—a survey," *Journal of Graph Theory*, vol. 6, pp. 1–22, 1982.

[9] M. Meringer, "Fast generation of regular graphs and construction of cages," *Journal of Graph Theory*, vol. 30, no. 2, pp. 137–146, 1999.

[10] M. Meringer, *Genreg-download manual*, http://www.mathe2.unibayreuth.de/markus/genreg.html.

[11] G. Royle, "Cages of higher valency," http://people.csse.uwa.edu.au/gordon/cages/allcages.html.

[12] N. Biggs, "Constructions for cubic graphs with large girth," *Electronic Journal of Combinatorics*, vol. 5, no. 1, 1998.

[13] E. Weisstein, "Cage graph," From MathWorld-A Wolfram Web Resource, http://www.mathworld.wolfram.com/CageGraph.html.

[14] H. Zhong and T. Zhang, "Design of VLSI implementation-oriented LDPC codes," in *Proceedings of 58th IEEE Vehicular Technology Conference (VTC '03)*, vol. 1, pp. 670–673, Orlando, Fla, USA, October 2003.

EURASIP Journal on Wireless Communications and Networking

Special Issue on Wireless Physical Layer Security

Call for Papers

Security is a critical issue in multiuser wireless networks in which secure transmissions are becoming increasingly difficult to obtain in highly mobile and distributed environments. In his seminal works of the late 1940s, Shannon formalized the concepts of capacity (as a transmission efficiency measure) and equivocation (as a measure of secrecy). Together with Wyner's fundamental formulation of the wiretap channel in the 1970s, this work laid the groundwork for the area of wireless physical area security. Interest in this area has exploded in recent years, motivated by the rise of wireless networking in general and by the increasing interest in large mobile networks with light infrastructure, which are extremely difficult to secure by traditional methods.

The objective of this special issue (whose preparation is carried out under the auspices of the EC Network of Excellence in Wireless Communications NEWCOM++) is to gather recent advances in the area of wireless physical layer security from the theoretical, such as the analysis of the secrecy capacity of various channel models, to more practical interests such as the development of codes and other communication schemes that can provide security in real networks. Suitable topics for this special issue dedicated to physical layer security include but are not limited to:

- Opportunistic secrecy
- The wiretap channel with feedback
- Authentication over the wiretap channel
- Information theoretic secrecy of fading channels
- Secrecy through public discussion
- Wireless key distribution
- Multiuser channels with secrecy constraints
- MIMO wiretap channels
- Relay-eavesdropper channel
- Scheduling for secure communications
- Secure communication with jamming
- Game theoretic approaches for secrecy
- Codes for secure transmission
- Secure compression
- Cognitive approaches for secrecy
- Physical Secrecy and Common Randomness
- Secrecy with channel uncertainty

Authors should follow the EURASIP Journal on Wireless Communications and Networking manuscript format described at the journal site http://www.hindawi.com/journals/wcn/. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/, according to the following timetable.

Manuscript Due	October 1, 2008
First Round of Reviews	January 1, 2009
Publication Date	April 1, 2009

Guest Editors

Mérouane Debbah, Alcatel-Lucent Chair on Flexible Radio, Supélec, 3 rue Joliot-Curie, 91192 Gif-sur-Yvette Cedex, France; merouane.debbah@supelec.fr

Hesham El-Gamal, Department of Electrical & Computer Engineering, Ohio State University, 205 Dreese Labs, 2015 Neil Avenue Columbus, OH 43210, USA; helgamal@ece.osu.edu

H. Vincent Poor, Department of Electrical Engineering, Princeton University, Engineering Quadrangle, Olden Street, Princeton, NJ 08544, USA; poor@princeton.edu

Shlomo Shamai, Department of Electrical Engineering, Technion, Technion City, Haifa 32000, Israel; sshlomo@ee.technion.ac.il