International Journal of Advanced Computer Research (ISSN (print): 2249-7277 ISSN (online): 2277-7970)
Volume-3 Number-3 Issue-11 September-2013

Design of Structured Regular LDPC Codes without Short Cycles using Gray Code Representations

Vibha Kulkarni[1], K. Jaya Sankar[2]

Abstract

Low Density Parity Check (LDPC) codes are specified by the parity check matrix H. Smaller girths in the Tanner graphs of LDPC codes prevent the sum-product algorithm from converging, i.e., short cycles in H-matrix degrade the performance of LDPC decoder. In this paper, we present an algorithm for constructing LDPC codes without short cycles. The H-matrix must be sufficiently sparse to increase the girth of a code. In this work, we present a simple method to increase the sparsity of matrix and provide lemmas to avoid short cycles of lengths 4 and 6. We design sub-matrices for column-weight two and column-weight three codes. This design makes use of a simple shifting function given by the proposed scheme. These are then combined into a final H-matrix using Gray code representations. The proposed method provides structured, regular codes with flexibility in code rates and code lengths. The regularity of these codes gives the advantages like simplicity of hardware implementation and fast encoding.

Key words

LDPC codes, Tanner Graph, Girth, Gray codes

1. Introduction

LDPC codes are originally developed by Gallager in 1960s [1] [2]. An (n, γ, ρ) LDPC code has $m = n \times \gamma/\rho$ rows and 'n' columns [3] [4], and is graphically represented by a bipartite graph called a Tanner graph [3] [5] with 'm' check nodes and '(n-m)' bit nodes. An edge connecting the bit node and a check node corresponds to a '1'in the parity-check matrix. A chain of nodes, where initial and terminal nodes are the same such that they do not use the same edge more than once is a cycle. The shortest length cycle in the graph is called girth 'g' of a code. Short cycles lead to inefficient decoding. Sullivan [6] showed using bit error rate simulations that large girth codes perform better than those with lower girths. Large girth speeds the convergence of iterative decoding and improves performance. Hence, LDPC codes with large girth are preferred.

Vibha Kulkarni, Electronics and Communication Engineering, Vasavi College of Engineering, Hyderabad, A.P., India.
K. Jaya Sankar, Electronics and Communication Engineering, Vasavi College of Engineering, Hyderabad, A.P., India.

There are different methods for constructing LDPC codes with large girth. Some of these put complicated constraints on the construction of H-matrices. We propose a different method for constructing H-matrix which is simpler, with fewer constraints in design, based on Gray Code representations (GC-LDPC). The basic algorithm to generate LDPC codes using Gray code representations with different code rates and code lengths, has been proposed in [7] and is modified to generate column-weight two codes with girths 8 and 12. In the present work, we provide a simple method to increase the sparsity of H-matrix for column-weight two codes, thereby increasing the code length. The expansion method, retaining the basic rules of Galleger's method of construction of H-matrix, is proposed in [7]. The short cycles in H-matrix degrade the performance of LDPC decoder. Hence, in this paper, we present an algorithm for constructing LDPC codes without short cycles. The construction process considers LDPC code parameters such as row and column weights, code rate, girth, sparsity and code length. The main objectives in code construction are good decoding performance and easier hardware implementation. Structured connections generally reduce hardware complexity and the cost of encoders and decoders [8][9][10].

Rest of the paper is organized as follows. Our previous work to construct LDPC codes based on Gray-code representations is briefed in section II. A simple and different method of code expansion for column-weight two LDPC codes, with girth eight, is given in section III. Section IV discusses the different shapes of short cycles in H-matrix. Section V presents an algorithm to construct H-matrix of column-weight three with a girth of eight. Code expansion of the same is also briefed in this section. Section VI provides the lemmas, with proof, to avoid short cycles of 4 and 6 in LDPC code. The code rates and code lengths obtained from the proposed technique are compared with the other existing methods in section VII. Section VIII concludes the paper.

2. Code Construction Using Proposed Algorithm

Selection of point set

Let 'H' be the parity-check matrix of an LDPC code with 'm' parity-check equations, i.e., H is $m \times n$ matrix, where 'n' is the code length. We represent these parity check equations by a set 'X' of 'm' decimal points. We call 'X' the '*point set*' of the H-matrix.

Let the elements of point set 'X' be denoted as $X = \{X_0, X_1, X_2, X_3 \ldots X_\rho\}$ where 'ρ' is the row-weight of H-matrix. The set of elements can be selected using either of the following two equations. The first element X_0 is zero in both the cases.

$$X_{i+1} = 2^i + X_i \quad (1)$$

$$X_{i+1} = 2X_i + 1 \quad (2) \qquad \text{for } i = 0,1,2,3,\ldots,\rho$$

Construction of sub-matrices

Let $H_1, H_2, H_3, \ldots, H_\gamma$ be the sub-matrices of H-matrix of an LDPC code. Let R_1H_1, R_1H_2, $R_1H_3, \ldots, R_1H_\gamma$ be the first rows of $H_1, H_2, H_3, \ldots, H_\gamma$, sub-matrices respectively. The construction of sub-matrices is briefed below.

Construction of H_1

The point set 'X' is selected according to the required row-weight as $X=\{X_0, X_1, X_2, \ldots X_i\}$ where $i=\rho$. The elements of 'X' form the first row of H_1 i.e. R_1H_1. The subsequent rows of H_1, i.e., R_2H_1, $R_3H_1, \ldots, R_{\rho+1}H_1$ are obtained by circularly left shifting the elements of their preceding rows, until the first row, R_1H_1, repeats.

Construction of H_2

The first row of H_2 is obtained by exchanging the first and second elements of the point set 'X', keeping the rest elements in the same positions i.e. $X = \{X_1, X_0, X_2, \ldots X_i\}$ forms the first row of H_2. The subsequent rows of H_2 are obtained by circularly shifting the elements, left, similar to that of sub-matrix H_1.

Construction of H_3

Similarly, the first row of the third sub-matrix is obtained by exchanging the first and third elements of the point set 'X', retaining the rest as they are. Hence, $X = \{X_2, X_1, X_0, \ldots X_i\}$ forms the first row of H_3. Subsequent rows are obtained as described for H_1 and H_2. This procedure is repeated for all the 'γ' sub-matrices of H-matrix. The minimum distance of the code is at least one more than the γ. The transpose of H matrix is given below:

$$H' = [H_1, H_2, H_3, \ldots, H_\gamma].$$

For example, the sub-matrix H_1, for row-weight 5, can be constructed using the five decimal values from the defined point set. The first row R_1H_1 is formed directly from the point set. The second and subsequent rows are obtained by circularly shifting the first and preceding rows, respectively, as shown below:

$$H_1 = \begin{pmatrix} 1 & 3 & 7 & 15 & 31 \\ 3 & 7 & 15 & 31 & 1 \\ 7 & 15 & 31 & 1 & 3 \\ 15 & 31 & 1 & 3 & 7 \\ 31 & 1 & 3 & 7 & 15 \end{pmatrix}$$

Similarly, all other sub-matrices are constructed. The elements of these sub-matrices are chosen as explained in the construction method. Finally, these matrix elements, which are in decimal form, are represented in Gray codes. The decimal numbers selected are such that, when converted into Gray code representations, they have more number of 0's compared to number of 1's, so that the generated H-matrix is sparse. The proposed algorithm gives the systematic way of selecting the decimal numbers using one of the design equations.

3. Column-Weight Two LDPC Codes

The applications and construction of column-weight two LDPC codes with girth eight are discussed in [7]. Structure graphs are used to find the girth of a code. The H matrix of column-weight two codes is constructed with two sub-matrices H_1 and H_2. We present a method for construction of H, by modifying the previous method described above, to achieve a girth of eight.

Construction of H_1

The point set 'X' can be selected according to the required row-weight i.e. $i=\rho$. Thus the set is $X = \{X_0, X_1, X_2, X_3, \ldots, X_\rho\}$. This forms the first row of H_1 i.e. R_1H_1. The subsequent rows of H_1 are obtained by cyclically shifting the elements of first row, either to the left or to the right, until the first row repeats.

Construction of H_2

The elements of the first row of H_1, in reverse order, form the first row of H_2 i.e. R_1H_2. The subsequent rows are obtained by cyclically shifting the elements of first row, either left or right, until R_1H_2 repeats.

Code expansion

Another method of code expansion, keeping the regularity of the graph unchanged, is to include equal number of zeros after each integer in the selected set. For example, if a single zero is included, after each integer, the code length doubles. Two zeros increase the code length by three times and so on. Thus ‘p’ number of zeros increases the code length by (p+1) times its original length.

4. Short Cycles in Parity-Check Matrix

Shapes of 4-Cycles in Parity-Check Matrix

If the number of 1’s that are in common between any two columns is greater than 1, then 4-length cycle exists. Hence, the devised algorithm must provide a method so that no two rows share ‘1’ in more than one column. In Figure 1, X_i and X_j represent two integers selected from the set to construct the H-matrix where $1 \le i \le \rho$ and $1 \le j \le \rho$. Let ‘h_a’ be the a^{th} row of H-matrix and ‘h_b’ be that of b^{th} row. Similarly, h_c and h_d are c^{th} and d^{th} columns, respectively. Here, X_i belongs to column h_c and X_j belongs to column h_d. If there are no four edges connecting X_i and X_j as shown, then there is no 4-cycle in the LDPC code. Let C_4 (X_i, X_i, X_j, X_j) denote a 4-cycle in H-matrix.

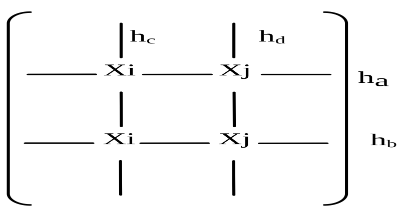

Figure 1: Shape of 4-cycle

Shapes of 6-Cycles in Parity-Check Matrix

Figure 2, shows the shape of a 6-cycle in H-matrix. The integers X_i and X_j of row h_a belong to the columns h_d and h_e, respectively. Likewise, X_i and X_k of row h_b belong to columns h_d and h_f, respectively. Similarly, X_j and X_k of row h_c belong to columns h_e and h_f, respectively. Here, X_i , X_j and X_k are the integers selected from the design equation to construct H-matrix where $1\le$ (i, j, k) $\le \rho$. The shape shown represents a 6-cycle, denoted as C_6 (X_i, X_i, X_k, X_k, X_j, X_j).

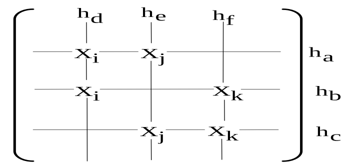

Figure 2: Shape of 6-cycle

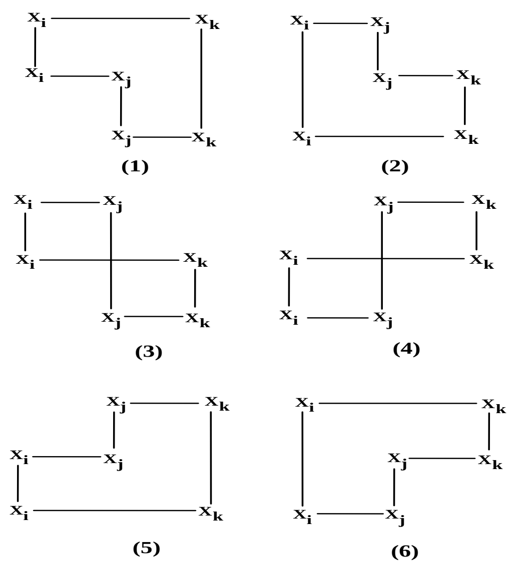

Figure 3: Different shapes of 6-cycle

The six different shapes of 6-cycles are shown in Figure 3. If H-matrix does not contain the set of edges connected in any one of the shown types, then there is no 6-cycle in the LDPC code.

5. Proposed Algorithm to construct Parity-Check matrix

We design three sub-matrices namely H_1, H_2 and H_3 to construct H-matrix of an (n, γ, ρ) LDPC code for γ=3 and $\rho > 3$. These matrices are combined to construct H-matrix and can be expanded to the desired length using identity matrices and cyclic shift matrices of the identity matrix.

Construction of sub-matrix H_1

Construction of H_1 is based on designing the matrices $H_{11}, H_{12}, \ldots\ldots, H_{1\rho}$.

1. Design a matrix H_{11} with the dimension $\rho \times \rho$. Let $H_{11}(1, 1) = H_{11}(2, 1) = H_{11}(3, 1) = \cdot \cdot \cdot = H_{11}(\rho, 1) = X$, other elements are ‘0’, where ‘X’ is a set of integers

selected to construct H-matrix. For example, for $\rho = 5$, set $X = \{X_1, X_2, X_3, X_4, X_5\}$.

$$H_{11} = \begin{bmatrix} X & 0 & \cdots & \cdots & 0 & 0 \\ X & 0 & \cdots & \cdots & 0 & 0 \\ X & 0 & \cdots & \cdots & 0 & 0 \\ \vdots & \vdots & & & \vdots & \vdots \\ X & 0 & \cdots & \cdots & 0 & 0 \end{bmatrix}_{\rho \times \rho}$$

The elements of the set 'X' are arranged in the form of a column in the matrix at the specified position. Rest all elements are zeros.

2. Let H_{12} be the matrix obtained by circularly right-shifting the elements of H_{11}, once.
3. H_{13} is obtained by circularly right-shifting the elements of H_{11} twice and so on.
4. This is repeated for $(\rho - 1)$ times to get all the matrices up to $H_{1\rho}$ matrix.
5. Now, by arranging the matrices H_{11}, H_{12},……,$H_{1\rho}$ vertically we get a matrix whose dimension is $\rho^2 \times \rho$.
6. The final matrix H_1 is the transpose of the matrix obtained in the previous step. The dimension of H_1 is $\rho \times \rho^2$.
7. The integer elements of H_1 are now converted into their corresponding Gray code representations such that each one of the elements is ρ-bits wide. Hence, the size of H_1 is $\rho^2 \times \rho3$, i. e.,

$$H_1 = [\, H_{11}, H_{12},\ldots\ldots.., H_{1\rho} \,]^T$$

For example, for $\rho = 5$, each integer is represented by 5-bits and the rest all elements of the matrix are zeros which are also represented in ρ bits. Hence, the size of H_1 is 25×125.

Construction of sub-matrix H_2

Similar to H_1, sub-matrix H_2 is also constructed by arranging H_{21}, H_{22},……..,$H_{2\rho}$ vertically and taking the transpose of the resulting matrix.

1. H_{21} is designed as shown below.
2. The sub-matrix H_{22} is obtained by circularly right shifting of the elements of H_{11} once.

$$H_{21} = \begin{bmatrix} X & 0 & \cdots & \cdots & 0 & 0 \\ 0 & X & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \cdots & \cdots & 0 & 0 \\ \vdots & \vdots & & & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & 0 & X \end{bmatrix}_{\rho \times \rho}$$

3. H_{23} is obtained by shifting the elements of H_{21} right, circularly, twice.
4. Similarly, all matrices up to $H_{2\rho}$ are obtained and are vertically arranged. The dimension of H_2 is $\rho^2 \times \rho$.
5. The transpose of the matrix obtained in the previous step gives H_2 having dimension of $\rho \times \rho^2$.
6. Each integer of H_2 is converted into its corresponding ρ-bit Gray code representation and hence, after code conversion, the dimension of H_2 is $\rho^2 \times \rho^3$.

$$H_2 = [\, H_{21}, H_{22},\ldots\ldots.., H_{2\rho} \,]^T$$

Construction of sub-matrix H_3

The design of sub-matrix H_3 is based on the construction of each of its rows. Design algorithm is as below.

1. The elements of the first row of H_3 are obtained as, $H_3(1, 1) = H_3(1, \rho + 1) = H_3(1, 2\rho + 1) = \cdot\ \cdot\ \cdot = H_3(1, ((\rho\text{-}1) \times \rho)+1) = X$, other elements are '0'.
2. The second row of H_3 is obtained by shifting the elements of the first row, to the right, once.
3. The third row is obtained by right shifting first row twice and so on.
4. Thus, the dimension of H_3 with integers as its elements is $\rho \times \rho^2$ and when converted into their corresponding Gray code representations, its size is $\rho^2 \times \rho^3$.
5. It can be observed that, the dimensions of all the sub-matrices H_1, H_2 and H_3 are $\rho^2 \times \rho^3$.
6. Finally, the H-matrix is constructed using these sub-matrices as given below. The dimension of H-matrix is $3\rho^2 \times \rho^3$.

$$H = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \end{bmatrix}_{3\rho^2 \times \rho^3}$$

Code expansion:

The expansion algorithm [11] is as follows:

1. Select an identity matrix I with dimension p × p.
2. Let $I_1 = I^1$, $I_2 = I^2$, . . . , $I_{(p-1)} = I^{(p-1)}$, where I^k represents the matrix I with circularly right-shifting 'k'steps.
3. Exchanging the 1's in H by the elements in the matrix set $\{I, I_1, I_2, \ldots, I_{p-1}\}$ randomly and exchanging 0's by null matrices with the same dimension as I, then we get the parity-

check matrix H, whose dimension is $(3\times p\times \rho^2) \times (p\times \rho^3)$.

From the structure of the parity-check matrix H, we can see that the proposed algorithm is a structured method and flexible in selection of the required row-weight. The proposed algorithm provides a regular LDPC code of column-weight 3 which is expandable to the desired code length. To get the LDPC code with the large length $(3\times p\times \rho^2)$, we can expand the H-matrix by selecting the desired dimension p of the identity matrix I.

6. H-Matrix without Short Cycles of 4 and 6

A method of constructing regular H-matrix of column-weight 3 and girth eight is discussed in the previous section. In this section we attempt to provide two lemmas to avoid short cycles of 4 and 6.

Lemma-1: If H-matrix is constructed using the proposed method, there is no 4-cycle in H.

Proof:
In Figure 1, the element X_i belonging to row h_a and another X_i belonging to row h_b appear in the same column h_c and we denote the distance between them as $d(X_i, X_i)$. Similarly, $d(X_j, X_j)$ is defined for another two elements X_j belonging to column h_d. From Figure1, we can know that if there are 4-cycles in the matrix H, the edges connecting $\{X_i, X_i\}$ and $\{X_j, X_j\}$ must be in two different sub-matrices H_1, H_2 and H_3 because X_i and X_j are in the same row. The following formula must be true:

$$d(X_i, X_i) = d(X_j, X_j)$$

In terms of the proposed construction method, the distances $d(X_i,X_i)$ and $d(X_j,X_j)$ could be 1, 2, . . . , ρ between the sub-matrices H_1 and H_2, and distances between the sub-matrices H_2 and H_3 could be 1, 2,…..,ρ. Similarly, between H_1 and H_3, the distances could be 2ρ, 2ρ + 1,…..,3ρ-1. From this it can be noted that $d(X_i, X_i) \neq d(X_j, X_j)$. Hence, there is no 4-cycle in the matrix H.

Lemma-2: If a matrix H is constructed by proposed method, there is no 6- cycle in H.
Proof: As shown in Figure 2, if there are 6-cycles in the matrix H, the edges connecting $\{X_i, X_i\}$, $\{X_j, X_j\}$, and $\{X_k, X_k\}$ must be in H_1, H_2 and H_3, respectively. If we assume that $d(X_j, X_j)$ is the longest length among the three lengths, the following formula must be true:

$$d(X_j, X_j) = d(X_i, X_i) + d(X_k, X_k)$$

We know that, the length of an edge connecting the 1's in H_1 and H_2, and in H_2 and H_3, can be 1, 2, . . . , ρ. The length of any edge between H_1 and H_3 can be 2ρ, 2ρ + 1,….., 3ρ - 1. Hence, it can be noticed that the length of the longest edge is not equal to the sum of the other two edges, thus, it is possible to avoid 6-cycles in matrix. i.e.,

$$d(X_i, X_i) + d(X_k, X_k) \neq d(X_j, X_j)$$

7. Results

In paper [11], the construction method of 'New' LDPC codes is proposed. These codes are compared with good codes like Tanner's QC (QC) codes and Mackay's (Mac) random codes. Table 1 below, lists the typical values of the row-weight 'ρ', the column-weight 'γ', the code length 'n', the code rate 'r', and the girth 'g' for the proposed GC-LDPC codes, 'New' codes, QC codes and Mac codes. Table 2 lists the typical values of the proposed codes and 'New' codes for different high code rates and code lengths. It is found that the parameters of GC-LDPC codes and 'New' LDPC codes are same.

Table 1: Code Size of Example Codes with Rate -½, Row-weight-6, Column-weight-3

	Code size (m × n)	**Code length (n)**	**Girth (g)**
GC-LDPC	1080 × 540	1080	8
New	1080 × 540	1080	8
Mac	1074 × 537	1008	6
QC	1074 × 537	1002	8

Table 2: Parameters of GC-LDPC and New Codes of Column-weight-3

	Code size (m × n)	**Row-weight (ρ)**	**Code length (n)**	**Code rate (r)**	**Girth (g)**
GC-LDPC	5103×1701	9	5103	0.67	8
New	5103×1701	9	5103	0.67	8
GC-LDPC	5000×1500	10	5000	0.7	8

New	5000×1500	10	5000	0.7	8
GC-LDPC	5184×1296	12	5184	0.75	8
New	5184×1296	12	5184	0.75	8

8. Conclusion

In this paper, we introduced a method to design structured LDPC codes, based on Gray code representation, without short cycles of 4 and 6, with large flexible code rates. This approach is simpler with fewer constraints in construction compared to other works in literature. Two lemmas are provided to prevent small cycles. This work presents algorithms for construction of column-weight three regular LDPC codes with girth 8. A simple method of expanding column-weight two codes, with girth eight, is presented. The code rates obtained from the proposed method are compared with good codes like Tanner's QC codes, Mackay's random codes and 'New' codes. The proposed method provides the similar results as for 'New' codes.

References

[1] R.G. Gallager, "Low Density Parity Check Codes," IRE Transactions on Information Theory, IT-8: 21-28, January 1962.

[2] R.G. Gallager, "Low Density Parity Check Codes," MIT Press, Cambridge, 1963.

[3] R.M. Tanner, "A recursive approach to low complexity codes," IEEE Transactions on Information Theory, vol. 27, no. 5, pp.533–547, 1981.

[4] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," IEEE Transactions on Information Theory, vol. 45, no. 2, pp. 399–431, 1999.

[5] R. M. Tanner, D. Sridhara, A. Sridharan, T. E. Fuja, and D. Costello Jr., "LDPC block and convolutional codes based on circulant matrices," IEEE Transactions on Information Theory, vol.50,no. 12,pp.2966–2984,2004.

[6] M.O'Sullivan, "Alzebraic Construction of Sparse Matrices with Large Girth,". IEEE Transactions on Information Theory, 52(2):719-727, February, 2006.

[7] Vibha Kulkarni, and Dr. K. Jaya Sankar, "A Novel Method for Construction of Structured Regular LDPC codes with Girth twelve using Gray Code representations," Springer proceedings of VCASAN-2013, The International Conference on VLSI, Communication, Advanced Devices, Signals and Systems and Networking, chapter 23, vol. 258, pp 169- 177, July, 2013.

[8] Shu Lin and Daniel J. Costello, Jr., "Error Control Coding," 2nd ed., Pearson, 2011.

[9] R. M. Tanner, "A recursive approach to low complexity codes," IEEE Transactions Information Theory, vol. 47, no. 2, pp. 498-519, Feb. 2001.

[10] H. Song, J. Liu, and B. Kumar, "Large Girth Cycle Codes for Partial Response Channels," IEEE Transactions on Magnetics, 40(4):2362/2371, July 2004.

[11] Jun Fan, Yang Xiao, and Kiseon Kim, "Design LDPC Codes without Cycles of Length 4 and 6," Hindawi Pub. Corp., Research, Letters in Communications, Vol. 2008, Article ID 354137, 5 pages, doi: 10.1155/2008/354137, doi:10.1155/2008/354137.

Vibha Kulkarni received B.E. degree in Electronics and Comm. Engg. from B. L. D. E.A's College of Engineering, Bijapur, Karnataka and M.Tech. Degree from M. S. Ramaiah Institute of Technology, Visvesvaraya Technological University, Belgaum, Karnataka, in 1992 and 2007, respectively. She is working as Assistant Professor in the Dept. of Electronics and Comm. Engg. at Vasavi College of Engineering, Hyderabad, India. She has 12 years of teaching experience in Communications, Microprocessors and Microcontrollers and Coding Theory and techniques. Currently she is a research scholar of Osmania University, pursuing her Ph. D. under the guidance of Dr. K. Jaya Sankar. She has published five papers in national and international conferences. She is a Member of IETE.

K Jaya Sankar received B.Tech degree in Electronics and Comm. Engg. from NBKR Institute of Science and Technology, Vidya Nagar, Andhra Pradesh and M.E. and Ph.D. degrees from the Dept. of ECE, Osmania University, Hyderabad, Andhra Pradesh in 1988, 1994, and 2004 respectively, He is working as Professor and Head of the Department of Electronics and Comm. Engg. at Vasavi College of Engineering, Hyderabad. He has 23 years of teaching experience in Communication, Electromagnetics, Antennas, Microwave Engg. His research areas of interest are in Coding Theory, Wireless Communications and Genetic Algorithm based Antenna design. He has published more than twelve papers in national and international journals and more than twenty papers in national and international conferences. He is a Member of IEEE, Fellow of IE (I) and IETE.