Makara Journal of Technology

Volume 11 | Issue 1 Article 5

4-1-2007

Construction of Short-Length High-Rates LDPC Codes Using Difference Families

Deny Hamdani
Communication Technology Institute, University of Dortmund, Otto-Hahn-Str. 4, 44221 Dortmund, Germany, denyhamdani@yahoo.com

Ery Safrianti
Communication Engineering Laboratory, University of Riau, Pekanbaru 28293, Indonesia, erysafrianti@yahoo.co.id

CONSTRUCTION OF SHORT-LENGTH HIGH-RATES LDPC CODES USING DIFFERENCE FAMILIES

Deny Hamdani[1], and Ery Safrianti[2]

1. Communication Technology Institute, University of Dortmund, Otto-Hahn-Str. 4, 44221 Dortmund, Germany
2. Communication Engineering Laboratory, University of Riau, Pekanbaru 28293, Indonesia

E-mail: denyhamdani@yahoo.com, erysafrianti@yahoo.co.id

Abstract

Low-density parity-check (LDPC) code is linear-block error-correcting code defined by sparse parity-check matrix. It is decoded using the massage-passing algorithm, and in many cases, capable of outperforming turbo code. This paper presents a class of low-density parity-check (LDPC) codes showing good performance with low encoding complexity. The code is constructed using difference families from combinatorial design. The resulting code, which is designed to have short code length and high code rate, can be encoded with low complexity due to its quasi-cyclic structure, and performs well when it is iteratively decoded with the sum-product algorithm. These properties of LDPC code are quite suitable for applications in future wireless local area network.

Keywords: low-density parity-check codes, quasi-cyclic codes, difference families, combinatorial design, iterative decoding, wireless local area network

1. Introduction

Low-density parity-check (LDPC) code was invented by Gallager [1] and ignored more than thirty years due to limited computational resources. This code sparked much interest in coding theory community after the success story of iterative decoding implemented by turbo code [2] and its rediscovery by MacKay and Neal [3], who proved their near-capacity performances. LDPC code can allow data transmission at code rate that is closed to the capacity of the channel within 0.0045 dB away from the Shannon limit [4].

LDPC code is error linear-block error-correcting code defined by a very sparse parity-check matrix $\boldsymbol{H}$, which is characterized with very low number of ones in column and row. Gallager [5] described regular code defined by parity-check matrix with constant column and row weight. Such code was pseudo-randomly constructed by avoiding 4-cycles, which can degrade decoding performance. These results were extended by Luby *et al.* [6] to irregular code, which has non-constant row and column weights in $\boldsymbol{H}$ and was proved to outperform regular ones due to their capability by discarding code which contains 4-cycles.

Contrary to turbo code, LDPC code has more reasonable decoding complexity in employing the message-passing algorithm, which involves passing probabilistic messages on a graph generated from the parity-check matrix. But, in performing the encoding algorithm, LDPC codes suffer from the computational complexity. In general, encoding is performed by matrix multiplication and so complexity is quadratic in code length.

In this paper, we present a class of LDPC codes, which is designed to meet requirement of applications in wireless local area network (WLAN). We analyze LDPC code with short code length and high code rate, which reduces the computational complexity in encoding to achieve power saving and latency shortening. The code is constructed using algebraic approach derived from a method from combinatorial design, i.e. difference families [7]. The resulting codes have quasi-cyclic structures. In decoder, we make use of sum-product decoding algorithm [5].

The remainder of the paper is outlined as follows. Section 2 provides the necessary theoretical background and design method for code construction. Section 3 discusses design results which show performance of investigated code. Finally, some concluding remarks are presented in Section 4.

2. Methods

Low-Density Parity-Check Code

LDPC code is a class of linear block code corresponding to the parity-check matrix $\boldsymbol{H}$ with very low density of ones. From the $(n–k)\times n$ matrix $\boldsymbol{H}$, we can derive corresponding $k\times n$ generator matrix $\boldsymbol{G}$, which

encodes k information bits into n codeword bits. The received codeword is decoded by $(n–k)$ check nodes. A regular LDPC code has constant number of ones in each column (column weight) and each row (row weight) in the parity-check matrix $\boldsymbol{H}$. If column weight and row-weight are not constant, then the code is irregular. Code rate r is equal to k/n, which means that $(n–k)$ redundant bits have been added to the message so as to correct the errors.

LDPC code can be represented effectively by a bi-partite graph called a *Tanner* graph [9]. A bi-partite graph is a graph (nodes or vertices are connected by undirected edges) whose nodes may be separated into two classes, and where edges may only be connecting two nodes not residing in the same class. The two classes of nodes in a Tanner graph are *bit nodes* and *check nodes*. The Tanner graph of a code is drawn according to the following rule: Check node j is connected to bit node i whenever element h_{ji} in $\boldsymbol{H}$ is a 1. One may deduce from this that there are $m = n – k$ check nodes, one for each check equation, and n bit nodes, one for each code bit. Further, the m rows of H specify the m c-node connections, and the n columns of $\boldsymbol{H}$ specify the n v-node connections. Figure 1 shows a Tanner graph made for a simple regular parity check matrix $\boldsymbol{H}$. In this graph each bit node is connected to two check nodes (bit node degree = 2) and each check node is connected to four bit nodes (check node degree = 4).

In parity-check matrix bit node degree and check node degree are represented by column weight w_c and row weight w_r, respectively. Column weight and row weight are the number of non-zero in column and in row, respectively.

For irregular LPDC codes, it is usual to specify the v-node and c-node degree distribution polynomials, denoted by $\lambda(x)$ and $\rho(x)$, respectively.

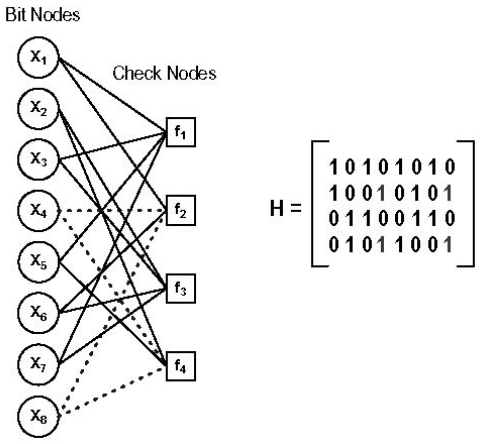

Figure 1. Tanner graph of parity check matrix $\boldsymbol{H}$.

$$\lambda(x) = \sum_{d=1}^{d_v} \lambda_d x^{d-1} \qquad (1.a)$$

$$\rho(x) = \sum_{d=1}^{d_c} \rho_d x^{d-1} \qquad (1.b)$$

In the polynomial (1.a) λ_d denotes the fraction of all edges connected to degree-d v-nodes and d_v denotes the maximum v-node degree. Similarly, in the polynomial (1.b) ρ_d denotes the fraction of all edges connected to degree-d c-nodes and d_c denotes the maximum c-node degree. Note for the example of the regular code in figure 1, for which $w_c = d_v = 2$ and $w_r = d_c = 4$, we have $\lambda(x) = x$ and $\rho(x) = x^3$.

A cycle (or loop) of length ν in a Tanner graph is a path comprising ν edges which closes back on itself. The Tanner graph in the above example possesses a *length*-4 cycle as exemplified by the four dashed edges in the figure 1. The *girth* γ of a Tanner graph is the minimum cycle length of the graph. The shortest possible cycle in a bipartite graph is clearly a length-4 cycle, and such cycles manifest themselves in the H matrix as four 1's that lie on the corners of a submatrix of H. We are interested in cycles, particularly short cycles, because they can degrade the performance of the iterative decoding algorithm used for LDPC codes.

Quasi-Cyclic Codes

A code is quasi-cyclic if, for any cyclic shift of a codeword by p places, the resulting word is also a codeword [8]. A cyclic code is a quasi-cyclic code with $p = 1$. Binary quasi-cyclic codes can be described by a parity-check matrix

$$\boldsymbol{H} = [\boldsymbol{H}_1\, \boldsymbol{H}_2 \ldots \boldsymbol{H}_p] \qquad (2)$$

where $\boldsymbol{H}_1 \ldots \boldsymbol{H}_p$ are binary $(v \times v)$ circulant submatrices. Provided that one of them is invertible (say $\boldsymbol{H}_p$) the generator matrix for the code can be constructed in systematic form resulting in a quasi-cyclic code of length vp and dimension $v(p\text{-}1)$. Encoding can be achieved with linear complexity using a $v(p\text{-}1)$-stage shift register in much the same way as for cyclic codes resulting in a (vp, v(p-1)) quasi-cyclic code [8,10]

$$G = \begin{bmatrix} & (H_p^{-1}H_1)^T \\ I_{v(p-1)} & (H_p^{-1}H_2)^T \\ & \\ & (H_p^{-1}H_{k-1})^T \end{bmatrix} \qquad (3)$$

A circulant submatrix H_i is completely characterized by the polynomial $a(x) = a_0 + a_1x + \ldots + a_{v-1}x^{v-1}$ with coefficients from its first row, and a code C with parity-check matrix of the form (2) is completely characterized

by the polynomials $a_1(x)$ … $a_k(x)$. Polynomial transpose is defined as

$$a(x)^T = \sum_{i=0}^{n-1} a_i x^{n-i} \quad \text{where } x^n = 1 \qquad (4)$$

For a binary [*n,k*] code, length $n = vp$ and dimension $k = v(p-1)$, the *k*-bit message $[i_0\ i_1\ \ldots\ i_{k-1}]$ is described by the polynomial $i(x) = i_0 + i_1x + \ldots + i_{k-1}x^{k-1}$ and the codeword for this message is $c(x) = i(x), x^k p(x)$, where $p(x)$ is given by

$$p(x) = \sum_{i=1}^{p-1} i_j(x) * \left(a_p^{-1}(x) * a_j(x) \right)^T \qquad (5)$$

The polynomial $i_j(x)$ is the representation of the information bits $v(j-1)$ to vj and polynomial multiplication (*) is mod x^v-1.

To construct a quasi-cyclic code for sum-product decoding we shall require that $\boldsymbol{H}$ is very sparse and that the Tanner graph of the code is free of 4-cycles. For this purpose code construction using difference families is proposed.

Difference Families

A difference family (henceforth, *DF*) is an arrangement of a group of *p c*-element subsets, such that every non-zero element of a group occurs λ times among the differences of element of subsets [9]. More precisely:

Definition: The *p c*-element subsets of the group Z_v, D_1 ... D_p with $D_i = \{d_{i,1}\ d_{i,2} \ldots\ d_{i,\gamma}\}$ form a (v,c,λ)-DF if the differences $d_{i,x} - d_{i,y}$, $(i = 1 \ldots p;\ x,y = 1 \ldots \gamma,\ x \neq y)$ give each nonzero element of Z_v exactly λ times. This can be formulated more compact in (6)

$$\Delta D := \left[x - y : x, y \in D, x \neq y \right] \qquad (6)$$

where ΔD is the collection of all differences of elements of *D*. For example, the subsets $D_1 = \{0,1,4\}$, $D_2 = \{0,2,7\}$ of Z_{13} form a (13,3,1)-*DF* with differences

$\Delta\{0,1,4\} = [0\text{-}1,0\text{-}4,1\text{-}0,1\text{-}4,4\text{-}0,4\text{-}1] = [12,9,1,10,4,3]$
$\Delta\{0,2,7\} = [0\text{-}2,0\text{-}7,2\text{-}0,2\text{-}7,7\text{-}0,7\text{-}2] = [11,6,2,8,7,5]$

Note that these subsets of 3-elements give the differences consisting of nonzero elements within the group of 13-elements exactly 1 time.

In this work we are interested in *DF* with λ=1 which allows the design of codes free of 4-cycles. The existence of $(v,3,1)$-*DF* has long been established for all $v \equiv 1 \bmod 6$, *v* a prime power [7]. In addition to that, existence results for $(v,4,1)$- *DF* and $(v,5,1)$- *DF*, $v \equiv 1 \bmod 12$ and $v \equiv 1 \bmod 20$, respectively, have been proven for all *v* a prime power. In the following the construction we analyse is described using these *DF*. For an irregular quasi-cyclic code we define the column weight distribution of a length *vp* with rates p-1/p code as the vector $\boldsymbol{W} = [w_1, w_2, \ldots, w_p]$, where w_j is the column weight of the columns in the *j*-th circulant submatrix. We denote by w_{max} the maximum column weight of $\boldsymbol{H}$

$$w_{max} = \max\{ w_1, w_2, \ldots, w_p \} \qquad (7)$$

To construct a length *vp* rates $(p-1)/p$ irregular quasi-cyclic code, $\boldsymbol{H} = [a_1(x), a_2(x), \ldots, a_p(x)]$, with weight distribution $W = [w_1, w_2, \ldots, w_p]$, we take *p* sets $D_1 \ldots D_p$ of a $(v,\gamma,1)$-*DF* with $\gamma \geq w_{max}$, such that $a_j(x)$ is defined, using w_j of the elements of D_j, as

$$a_j(x) = x^{d_{j,1}} + x^{d_{j,2}} + \Lambda + x^{d_{j,w_j}} \qquad (8)$$

To ensure invertibility at least one $a_j(x)$ must be able to divide x^v-1.

For a regular code all of the elements in each set are included in each circulant submatrix, while for an irregular code the choice of which elements in the set to use is arbitrary, and in fact a single set can be used to construct two circulant submatrices provided that different elements are chosen for each. The row weight ρ of the parity-check matrix is constant, and given by (9)

$$\rho = \sum_{i=j}^{p} w_i \qquad (9)$$

By choosing λ=1 this quasi-cyclic code is free of 4-cycles. In the regular case, each column of $\boldsymbol{H} = [a_1(x), a_2(x), \ldots, a_p(x)]$ is a translate of one of the sets D_j in the *DF*. The matrix $\boldsymbol{H}$ has no 4-cycles if two columns of $\boldsymbol{H}$ have no nonzero entry in the same two rows, which is equivalent to requiring that two elements of Z_v can occur together in at most one of all the translates of the sets in the *DF*. Since two elements occur together in exactly λ=1 translate, the 4-cycles can be avoided.

3. Results and Discussion

Using the (101,5,1)-*DF* from [7], we have the following subsets

$D_1 = \{0,14,42,47,55\}$,
$D_2 = \{0,52,63,83,95\}$,
$D_3 = \{0,17,21,51,74\}$
$D_4 = \{0,7,26,36,92\}$,
$D_5 = \{0,61,76,98,100\}$

In constructing code, we are interested in LDPC code with short length in order of hundred bits and high rates

(code rate ≥ 3/4). Therefore, we construct a quasi-cyclic irregular (404,303) DF-LDPC code with the column vector W = [5,5,3,2] characterized by the polynomials of circulant matrices in equation (10). Henceforth, this code is regarded as our reference code.

$$a_1(x) = x^{D_1}$$
$$a_2(x) = x^{D_2}$$
$$a_3(x) = 1 + x^{d_{21}} + x^{d_{74}}$$
$$a_4(x) = x^{d_{36}} + x^{d_{92}} \qquad (10)$$

We make use of BPSK modulation in this scheme. Notice that all codes are decoded by using sum-product decoding with maximal iterations 20 over additive white *gaussian* noise (AWGN) channel.

Firstly, our DF-LDPC code is compared to random codes proposed by MacKay-Neal [3] with comparable parameters. Figure 2 shows that the DF-LDPC code can compete the random code. It performs about 2.7 dB away from the Shannon limit at the BER of 10^{-5}. In addition to that, as quasi-cyclic code, this code offers advantage in encoding, which is conducted using a shift-register circuit of size equal to the code dimension. Encoding of the quasi-cyclic code requires $(n\text{-}k)\alpha$ binary operations, where α is one less than the row weight of $\boldsymbol{G}$, while matrix multiplication requires $(n\text{-}k)(2k\text{-}1)$ binary operations [10].

Moreover, we can reduce storage requirement significantly by just specifying the elements of submatrices.

We are also interested in comparing performance of irregular DF-LDPC code with regular one and the effect of their column weight. We construct regular (404,303) DF-LDPC codes with w=3 and w=5 whose polynomials are presented in (11.a) and (11.b), respectively.

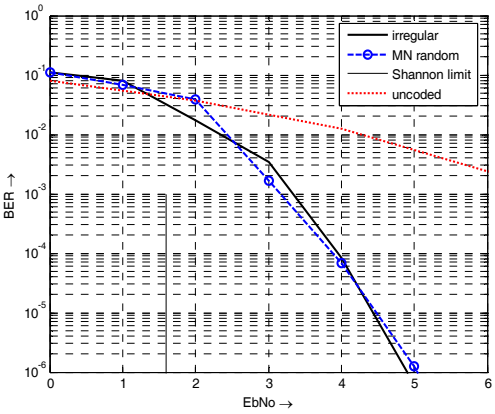

Figure 2. Error correction performance of DF-LPDC code

$$a_1(x) = 1 + x^{d_{14}} + x^{d_{42}}$$
$$a_2(x) = x^{d_{52}} + x^{d_{63}} + x^{d_{95}}$$
$$a_3(x) = x^{d_{21}} + x^{d_{51}} + x^{d_{74}}$$
$$a_4(x) = x^{d_7} + x^{d_{36}} + x^{d_{92}} \qquad (11.a)$$

$$a_1(x) = x^{D_1}$$
$$a_2(x) = x^{D_2}$$
$$a_3(x) = x^{D_3}$$
$$a_4(x) = x^{D_4} \qquad (11.b)$$

Figure 3 shows that irregular DF-LDPC code is slightly superior against regular ones. Meanwhile, the regular LDPC code with larger column weight (w=5) outperforms that with less column weight (w=3) after the BER of 10^{-5}.

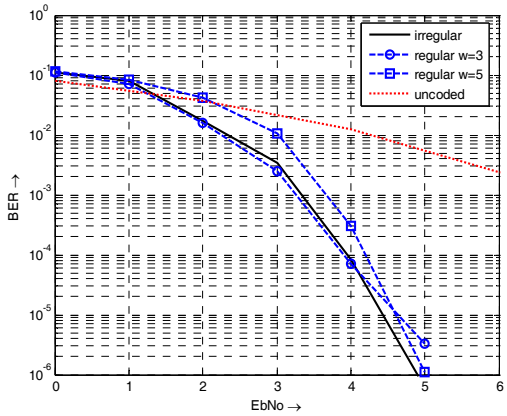

Figure 3. Performance of irregular and regular DF-LDPC code

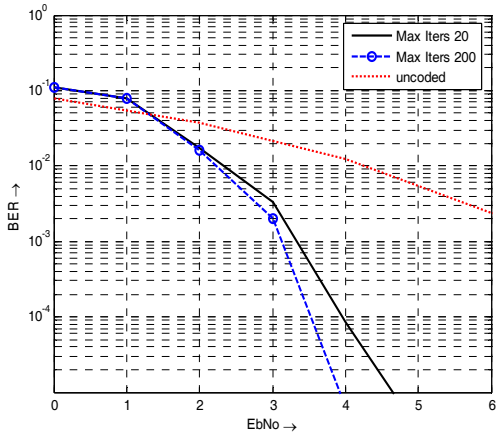

Figure 4. Effect of maximum iteration on code performance

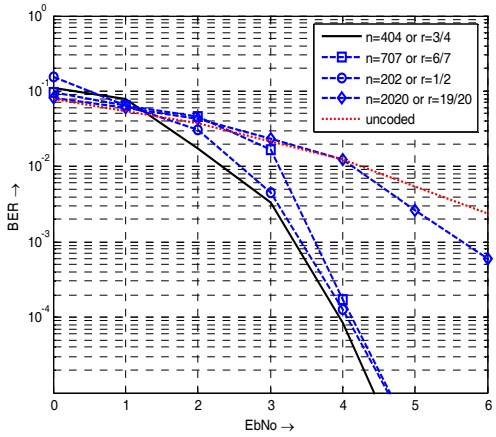

Figure 5. Performance of DF-LPDC code with different code rates

Furthermore, we will study the effect of maximum iteration number on code performance. Figure 4 demonstrates that large maximum iteration number provides larger code gain at less bit error probability. A code gain of 0.5 dB is achieved at the BER of 10^{-5} with ten times of reference maximum iteration. However, in practice, this performance improvement should be traded-off with latency requirement of system.

It is of interest to study the DF-LDPC code with different code rates. As mentioned in previous section, the DF construction results in DF-LPDC code with code length vp and code rate $(p\text{-}1)/p$. It means that prolonging the code by adding submatrices into the parity-check matrix will make the code rate higher. We construct the LDPC codes with code rate ½ and 19/20 with the polynomials in (12.a) und (12.b), respectively.

$$a_1(x) = x^{D_1} \quad a_2(x) = 1 + x^{d_{52}} + x^{d_{95}} \qquad (12.a)$$

$$a_1(x) = x^{D_1} \quad a_2(x) = x^{D_2} \quad a_3(x) = x^{D_3}$$
$$a_4(x) = 1 + x^{d_7} + x^{d_{36}} \quad a_5(x) = 1 + x^{d_{76}} + x^{d_{98}}$$
$$a_6(x) = x^{d_{14}} + x^{d_{55}} \quad a_7(x) = x^{d_{52}} + x^{d_{83}} \qquad (12.b)$$

In figure 5, we find out that performance of DF-LDPC codes with code length in order of hundred bits is better than that in thousand bits. With v = 101 we can still achieve good parameter at the code rate 0.85 or the code length 707. To achieve good performance in longer code length, we need larger submatrix dimension or accordingly larger parameter v.

4. Conclusions

Constructing quasi-cyclic LPDC code with difference families offer a significant advantage in encoding, which is main drawback in its application. Thanks to quasi-cyclic form, the encoding of this code can be implemented with linear shift-registers, which reduces the encoding complexity significantly.

The decoding performance of this code demonstrates a modest performance gain for reasonably short lengths and high rates in practical iteration number. The simulation results show that the selection of the parameter v plays an important role in code design. With smaller parameter v we can achieve good performance with shorter code length at high rate. This construction seems to be suitable for WLAN application with high rate short packet.

Further study is needed to prove the error-correcting capability of this code in real WLAN channels. Moreover, the characteristics of this code applicable for WLAN applications will be explored in the future.

References

[1] Gallager, R.G. Low-density parity check codes. 1962. *IRE Trans. Inform. Theory*, IT-8: 21-28.

[2] Berrou, C., Glavieux, A., Thitimajshirna, P. Near Shannon limit error correcting coding and decoding: Turbo codes. 1993. in *Proceeding of International Conference on Communications*. Geneva, Switzerland: 1064-1070.

[3] MacKay, D.J.C., and Neal, R.M. 1995. Good codes based on very sparse matrices. in Colin Boyd, Ed *Cryptography and Coding: Proceeding of 5th IMA Conference, Lecture Notes in Computer Science No.1025*. Springer-Verlag, Berlin, p. 100-111.

[4] Chung, S.-Y., Richardson, T., Urbanke, R. 2001. On the design of low-density parity-check codes within 0.0045 dB of the Shannon limit. *IEEE Communications Letters,* vol.**5** no.2: 58-60.

[5] Gallager, R. G. 1963. Low-Density Parity Check Codes. Ph.D dissertation. MIT Press, Cambridge, USA.

[6] Luby, M., Mitzenmacher, M., Shokrollahi, M., Spielman, D. 2001. Efficient erasure correcting codes. *IEEE Trans. Inform. Theory*, **47:** 569-584.

[7] Vasic, B., Milenkovic, O. 2004. Combinatorial Constructions of Low-Density Parity-Check Codes for Iterative Decoding, accepted for publication in *IEEE Transactions on Information Theory*.

[8] R.L. Townsend and E. Weldon. 1967. Self-orthogonal quasi-cyclic codes. *IEEE Trans. Inform. Theory*, vol. IT-13, **2**: 183-195.

[9] R.M. Tanner. 1981. A Recursive Approach to Low Complexity Codes. *IEEE Trans. Inform. Theory,* vol. IT-27, **5**: 533-547.

[10] Johnson, S.J., Weller, S.R. 2002. Quasi-cyclic LDPC codes from difference families. In *Proceeding 3rd AusCTW*, Canberra, Australia.