Design of binary LDPC code using cyclic shift matrices

K.S. Kim, S.H. Lee, Y.H. Kim and J.Y. Ahn

An LDPC code structure using cyclic shift matrices and an efficient LDPC code construction algorithm using the characteristic matrix is proposed. It is shown that, with the proposed structure and construction algorithm, fast construction time, low encoding complexity and reduced memory without performance degradation can be achieved.

Introduction: Over recent years, low density parity check (LDPC) codes have received much attention owing to their possibility of providing near Shannon-limit performance and relatively simple decoding structure. However, there are some challenging issues relating to application of the LDPC codes in practice: how to design an LDPC code in the case of a finite block length and how to reduce the complexity in encoding and storing the parity check matrix. In [1], the density evolution method was proposed to optimise LDPC codes in an asymptotic sense. However, there is no guideline for constructing parity check matrices of the short block length. A relatively efficient encoding method for LDPC codes was proposed in [2]. Although the encoding complexity of the method in [2] is almost linear with the codeword length, the complexity in matrix storage and computation needs to be reduced for practical applications. To resolve these problems, several structured LDPC codes have been proposed [3, 4] at the cost of performance degradation. In this Letter, a design method of the LDPC code using cyclic shift matrices is proposed to obtain both good performance and practically allowable encoding complexity.

LDPC code structure: The parity check matrix H of the proposed (NB,KB) binary LDPC code can be represented by $H=[\gamma_{i,j}]$, where $\gamma_{i,j}$, $0\le i\le N-K-1$ and $0\le j\le N-1$, denotes a constituent sub-matrix that is either a $B\times B$ zero matrix or a $B\times B$ cyclic shift matrix σ_b, $0\le b\le B-1$. Here the element of σ_b at the pth row and the kth column is $\delta_K((k-b-p)\text{mod}B)$, where $\delta_K(\cdot)$ is the Kronecker delta function. Here, for $x<0$, $x\text{mod}B$ is defined as $(B+x)\text{mod}B$. Also, we define the $(N-K)\times N$ characteristic matrix C_H of the parity check matrix H as $C_H=[v_{i,j}]$, where $v_{i,j}=0$ if $\gamma_{i,j}$ is a zero matrix and $v_{i,j}=b+1$ if $\gamma_{i,j}=\sigma_b$. Then, from the definition, we obtain the following theorem.

Theorem 1: The $(kB+\kappa)$th column (row) of the matrix H, $0\le\kappa\le B-1$, contains Q elements of '1' if and only if the kth column (row) of the matrix C_H contains Q nonzero elements.

Proof: Since any column (row) of a cyclic shift matrix σ_b, $0\le b\le B-1$, has only one nonzero element, the number of '1' in the $(kB+\kappa)$th column (row) of the matrix H, $0\le\kappa\le B-1$, is equal to the number of nonzero elements in the kth column (row) of the corresponding characteristic matrix C_H, and vice versa.

From Theorem 1, we can see that the variable (check) node degree distribution of the parity check matrix H is identical to that of the corresponding characteristic matrix C_H. Now, let us investigate how a cycle in the characteristic matrix C_H relates to the cycles in the parity check matrix H. Let $\mu=\{\mu_0,\mu_1,\ldots,\mu_{2l_\mu-1}\}$ denote a cycle of length $2l_\mu$ in the bipartite graph defined by C_H. Here, μ_i represents the value of the element in C_H corresponding to the ith edge of the cycle μ. Let us also define

$$s_l(\mu)=\left(\sum_{k=0}^{l-1}(-1)^k\mu_k\right)\text{mod } B \tag{1}$$

for $1\le l\le 2l_\mu$, and $s(\mu)=s_{2l_\mu}(\mu)$. Then, the following theorem is obtained.

Theorem 2: Let $\beta(\mu)$ be the greatest common divisor of $s(\mu)$ and B, then the nodes in H, corresponding to the nodes in C_H that form the cycle μ, form $\beta(\mu)$ cycles with length $2Bl_\mu/\beta(\mu)$ in the bipartite graph defined by H.

Proof: Let us define the functions $c(b,i)$ and $r(b,j)$ as the column position of the element '1' at the ith row and the row position of the element '1' at the jth column in σ_b, respectively. Then, it is easily seen that $c(b,i)=(i+b)\text{mod}B$ and $r(b,j)=(j-b)\text{mod}B$. Without loss of generality, μ is regarded as a counter-clockwise directed path starting from the left-uppermost sub-matrix in H. Now, let us start with the element '1' at the qth row of the first sub-matrix in the cycle μ. Then, the column position of the element '1' at the qth row in the first sub-matrix is $c(\mu_0-1,q)=(q+\mu_0-1)\text{mod}B$. Also, the row position of the element '1' at the $c(\mu_0-1,q)$th column in the second sub-matrix in the cycle μ is $r(\mu_1-1,\ c(\mu_0-1,\ q))=(q+\mu_0-\mu_1)\text{mod}B$. Thus, it is straightforward that the row position at the first sub-matrix after traversing the cycle t times is $(q+ts(\mu))\text{mod}B$. Since $\beta(\mu)$ is the greatest common divisor of $s(\mu)$ and B, the smallest number t such that $(q+ts(\mu))\text{mod}B=q$ is $B/\beta(\mu)$. Now, let us consider the cycle in H starting from the element '1' at the qth row in the first sub-matrix, where $0\le q<\beta(\mu)$. Then, the set of row (or column if l is odd) positions of the constituent element '1' of the cycle, contained in the lth sub-matrix, is $\{(q+ts(\mu)+s_l(\mu))\text{mod}B,\ \ 0\le t<B/\beta(\mu)\}$. Thus, for $0\le p\ne q<\beta(\mu)$, the cycles starting from the element '1' at the qth row and the element '1' at the pth row in the first sub-matrix, respectively, have no common element of '1' in H because $(q+t_1s(\mu)+s_l(\mu))\text{mod}B\ne(p+t_2s(\mu)+s_l(\mu))\text{mod}B$, for any $0\le t_1,t_2<B/\beta(\mu)$. Therefore, the nodes in H, corresponding to the nodes in C_H that form the cycle μ, form exactly $\beta(\mu)$ cycles with length $2Bl_\mu/\beta(\mu)$.

Corollary 1: If $s(\mu)$ and B are relative prime, then the nodes in H, corresponding to the nodes in C_H that form the cycle μ, form only one cycle with length $2Bl_\mu$ in the bipartite graph defined by H.

Proof: Since $s(\mu)$ and B are relative prime, we have $\beta(\mu)=1$. Thus, from Theorem 2, the nodes in H, corresponding to the nodes contained in the cycle μ, form only one cycle with length $2Bl_\mu$ in the bipartite graph defined by H.

LDPC code design: The density evolution methods [1, 5] can be used to determine the node degrees of the characteristic matrix C_H. Then, from Theorem 1, the node degrees of the parity check matrix H are the same as those of the characteristic matrix C_H. With the determined node degrees, the positions and values of nonzero elements in C_H are determined by using Theorem 2 in a bit-filling manner. The outline of the proposed algorithm is as follows.

Inputs : $N, K, B, L, N_v(n), n=0,\ldots,N-1, N_c(m),$
$m=0,\ldots,N-K-1$
1 : set $n=0, ind=0, R_c=\{0,1,\ldots,N-K-1\},$
$R_g=\{1,2,\ldots,B\}$
2 : do {
3 : for $(i=0; i<N_v(n); i++)$ {
4 : $R'_c=R_c$
5 : while($Cardinality(R'_c)>0$){
6 : randomly select $c\in R'_c$ among the elements R'_c in with
: the largest remaining degree $N_c(m)$
7 : compute the cost function $f(c,g)$ up to length
: $2L$ caused by the edge connected to the check node c
with the value $g\in R_g$
8 : if $f(c,g)$ is better than $f(c^*,g^*)$, update $c^*=c, g^*=g$
9 : $R'_c=R'_c-\{c\}$
10 : }
11 : $VarIndex(ind)=n, CheckIndex(ind)=c^*, Value(ind)=g^*$
: $N_c(c^*)=N_c(c^*)-1, ind=ind+1$
12 : $R_c=R_c-\{c^*\}$
13 : }
14 : Reconstruct $R_c=\{m|N_c(m)>0, m=0,\ldots,N-K-1\}$
15 : $n=n+1$
16 : } while $(n<N)$
Outputs : $Varindex, CheckIndex, Value$

Here, $N_v(n)$ and $N_c(m)$ are the degrees for the nth variable node and the mth check node determined from the density evolution result, respectively. For a given column, all possible positions and values for a new nonzero element are tested for cycle length up to $2L$ based on the given cost function f. Then, the position and value of the new nonzero element, which draw the best cost function, are selected. Here, the cost function can be any well-known criterion, such as the minimum cycle length, the cycle distribution, the ACE [6], and so forth. In this Letter, the cycle distribution caused by the new nonzero element is used as the cost function and compared as follows: (i) set the current cycle length $2l$ at four, (ii) choose one with the smaller number of cycles with length $2l$, and (iii) if two distributions have the same number of cycles, increase l by one and repeat (ii) until $l < L$. In Fig. 1, the performance of the binary irregular LDPC codes, constructed with the proposed structure and algorithm, is shown when $B = 32$. Here, the degree distributions are obtained from the density evolution using Gaussian approximation [5]. For comparison, the performance of the conventional binary irregular LDPC codes with the same degree distributions is also shown in Fig. 1. Here, the conventional LDPC codes are constructed with the minimum cycle length as the cost function. As shown in Fig. 1, the performance of the proposed code is even slightly better than that of the conventional code. This is mainly due to the fact that we investigated the minimum cycle length only in constructing the conventional code. However, the construction time required for the proposed code is much smaller than that required for the conventional code since B times smaller number of nodes needs to be investigated.

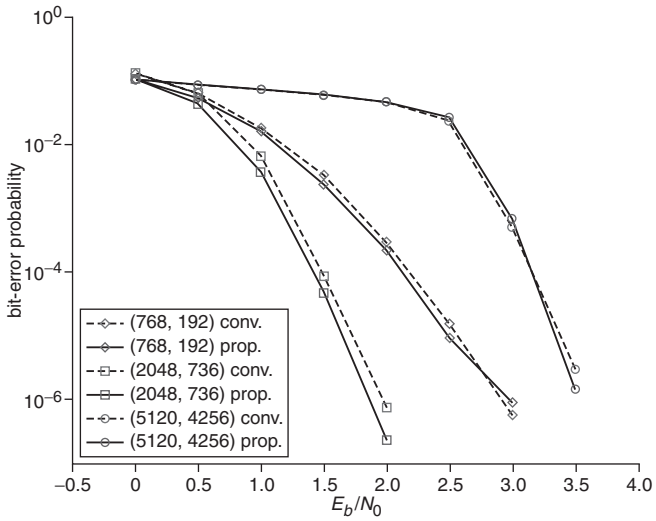

Fig. 1 *Performance of proposed LDPC code*

Conclusions: In this Letter, the LDPC code structure using cyclic shift matrices and the efficient code construction algorithm using the characteristic matrix are proposed. The construction time required for the proposed LDPC code is much smaller than that required for a conventional LDPC code with the same codeword length and code rate since the number of rows (column) in the characteristic matrix is B times smaller than those in the parity check matrix. Also, since cyclic shift matrices are used as the sub-matrix in the proposed LDPC code, an efficient B-bit word-by-word encoding scheme can be utilised by slightly modifying the algorithm proposed in [2] with only shift and bit-wise exclusive-or operations. Moreover, by employing the proposed LDPC code structure, we can reduce the required memory for storing an LDPC code by roughly B times. Simulation results showed that the proposed LDPC code can achieve those improvements in complexity without performance degradation.

17 December 2003
Electronics Letters online no: 20040188
doi: 10.1049/el:20040188

K.S. Kim, S.H. Lee, Y.H. Kim and J.Y. Ahn (*Mobile Communications Research Lab., Electronics and Telecommunications Research Institute, 161 Kajeong-dong, Yuseong-gu, Daejeon 305-350, Korea*)

E-mail: kskim@ieee.org

References

1 Richardson, T.J., Shokrollahi, M.A., and Urbanke, R.L.: 'Design of capacity-approaching irregular low-density parity-check codes', *IEEE Trans. Inf. Theory*, 2001, **47**, pp. 619–637
2 Richardson, T.J., and Urbanke, R.L.: 'Efficient encoding of low-density parity-check codes', *IEEE Trans. Inf. Theory*, 2001, **47**, pp. 638–656
3 Echard, R., and Chang, S.-C.: 'Deterministic π-rotation low-density parity check codes', *Electron. Lett.*, 2002, **38**, pp. 464–465
4 Prabhakar, A., and Narayanan, K.: 'Pseudorandom construction of low-density parity-check codes using linear congruential sequences', *IEEE Trans. Commun.*, 2002, **50**, pp. 1389–1396
5 Chung, S.-Y., Richardson, T.J., and Urbanke, R.L.: 'Analysis of sum-product decoding of low-density parity-check codes using a Gaussian approximation', *IEEE Trans. Inf. Theory*, 2001, **47**, pp. 657–670
6 Tian, T., Jones, C., Villasenor, J.D., and Wesel, R.D.: 'Construction of irregular LDPC codes with low error floors'. Proc. IEEE Int. Communication Conf. (ICC), Anchorage, AK, USA, May 2003, Vol. 5, pp. 3125–3129