LCPC CODES - A BRIEF OVERVIEW

BY
ZOHAN B PHILIP
NAKKA CHAKRADHAR

INTRO TO LCPC CODES

- One major concern in IoT and WSNs is the energy conservation and consumption. Therefore, avoiding or reducing the number of the faulty packets’ retransmissions is significant.

- A Low Complexity Parity Check (LCPC) code that detect and correct single and double bit errors is more apt for this case.

- The main characteristic of LCPC codes is that it is capable of correcting few of the two bit error cases, helping IoT devices in burst error scenarios instead of requesting for packet repeat.

WHY LOW COMPLEXITY ?

- IoT devices and wireless sensor networks have a very tight constraints when it comes to battery life and power consumption in general. Hence a packet re-request is a very costly affair in terms of energy
- We use an ECC to counter this scenario
- The ECC itself shouldn’t be too computationally intensive to consume more power and negate the advantages of overcoming packet re-request.
- Hence, a low complexity parity check code should do the job, depending on the channel conditions

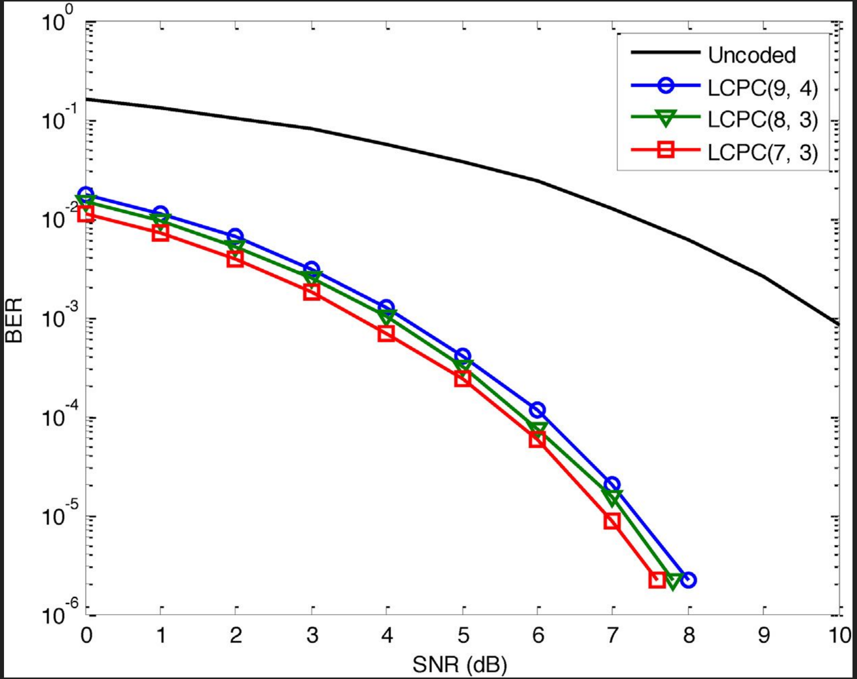
THEORY BEHIND LCPC CODES

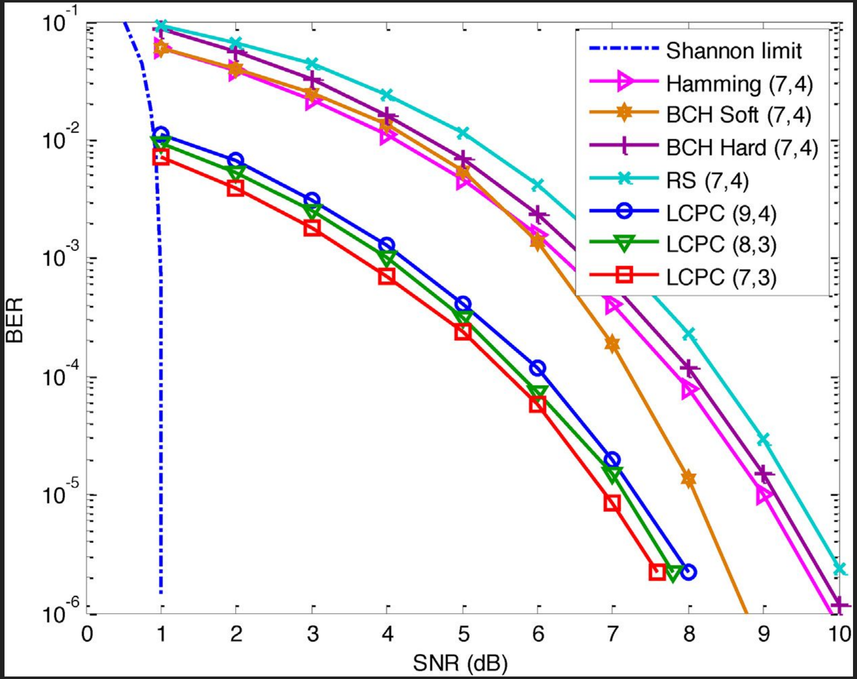
- LCPC codes are nothing too fancy, they are just linear parity check codes with a relatively small error lookup table and syndrome calculation is easy.
- The expressions for the parity bits are as follows :

$$\beta 5 = \gamma 1 = v1 \oplus v2 \oplus v3 \oplus v4$$
$$\beta 6 = \gamma 2 = v1 \oplus v2 \oplus v3$$
$$\beta 7 = \gamma 3 = v1 \oplus v2 \oplus v4$$
$$\beta 8 = \gamma 4 = v1 \oplus v3 \oplus v4$$
$$\beta 9 = \gamma 5 = v2 \oplus v3 \oplus v4$$

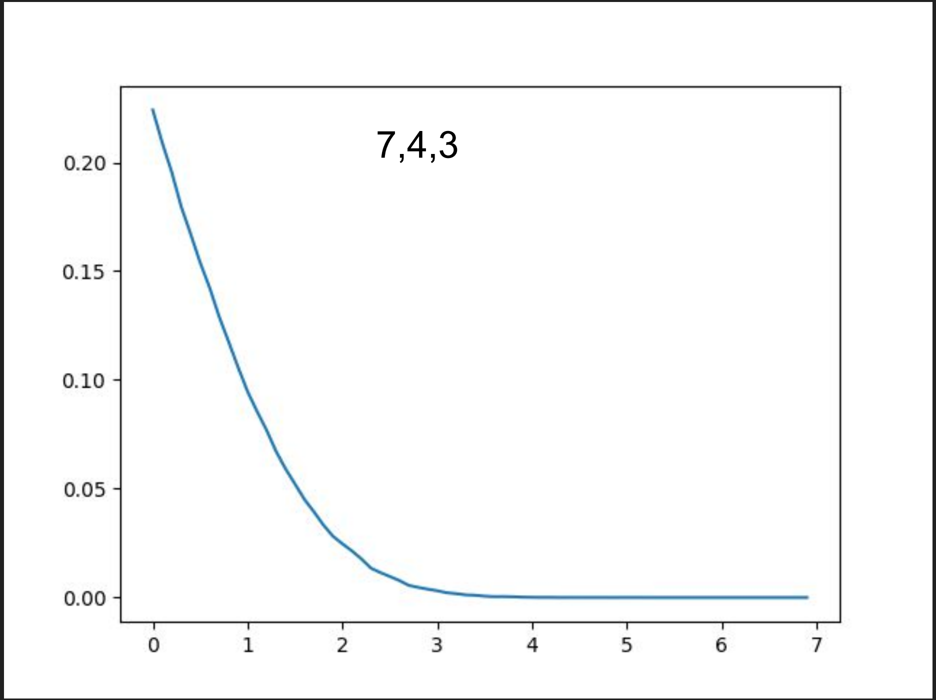
Where v1, v2, v3, v4 are the 4 incoming message bits and they are being mapped to a 9 bit string.

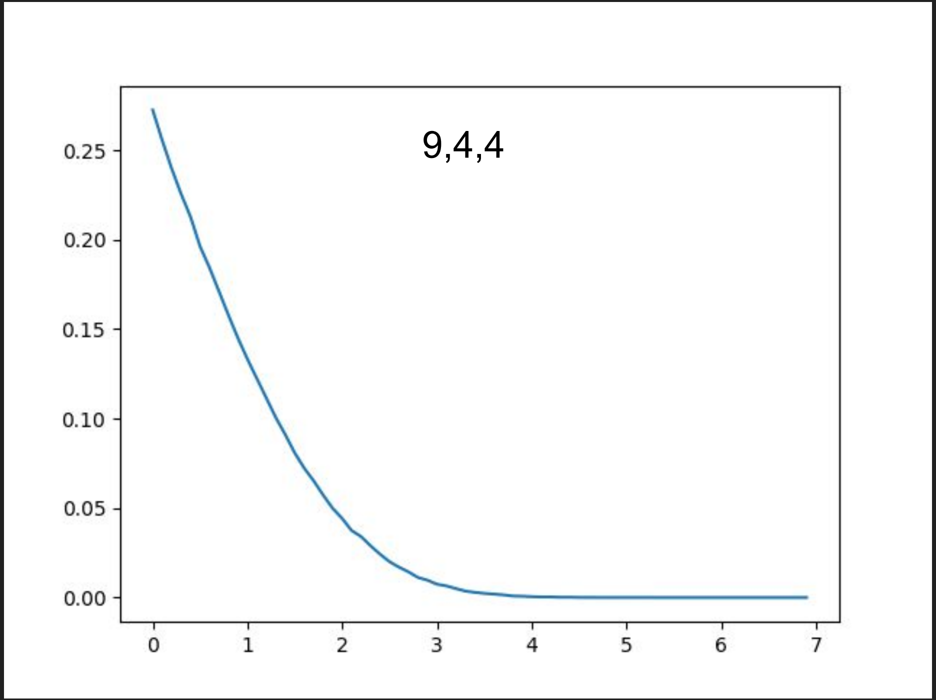
SIMULATION RESULTS BY THE PAPER





REPLICATED SIMULATION RESULTS





UNFAIR BASIS OF COMPARISON

- In the paper, probability of error with respect to SNR of (7,4,3) Hamming Code and (9,4,4) LCPC were compared unfairly.
- The SNR in each case was the ratio of the energy needed to send one bit to the variance of noise. When such a comparison is made, it has been ignored that one message bit does not correspond to same number of information bits in both cases.
- The energy required to transmit one codeword is 7 times the energy required to transmit one bit in (7,4,3) Hamming code and 9 times in case of (9,4,4) LCPC. Both encode the same number of information bits.
- Thus the energy per bit for (9,4,4) LCPC must be 7/9 times the energy per bit for (7,4,3) Hamming code given the same noise for a fair comparison between error performance for same SNR.

PROPOSED SOLUTION

- The 9,4,4 code discussed earlier when compared fairly with 7,4,3 gives a poor error performance.
- Thus we have a proposed a better solution which is a (4,2,3) code on F4.
- Such a code does not induce a harsh drop in rate from (7,4,3) and has a slightly better error performance than (7,4,3) hamming code on a normalised scale of comparison.
- The complexity of decoding is low since since we have used syndrome decoding. Since the rate is ½ in case of (4,2,3), 4/7 in case of (7,4,3) and 4/9 in case of (9,4,4) the SNR v Probability of error was done by multiplying the variance of noise in each case by 8/9, 7/9 and 1 respectively.

PROPOSED SOLUTION

- The generator matrix for the proposed (4,2,3) is as follows:

$$\begin{pmatrix} 1 & 0 & 1 & w \\ 0 & 1 & 1 & w^2 \end{pmatrix}$$

- Such that if s1&s2 are the input symbols, p1 and p2 are the parity symbols and(c1,c2,c3,c4) is the codeword, then

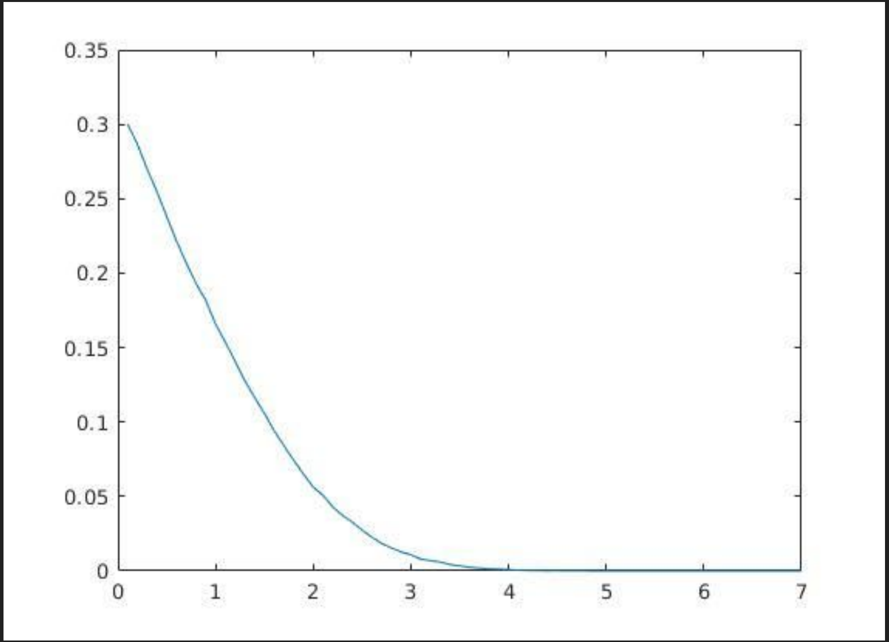
1. $c1=s1$
2. $c2=s2$
3. $c3=p1=s1+s2$
4. $c4=p2=(s1*w)+(s2*w^2)$

Since the generator matrix is in systematic form, the parity check matrix matrix for this code can be constructed easily.

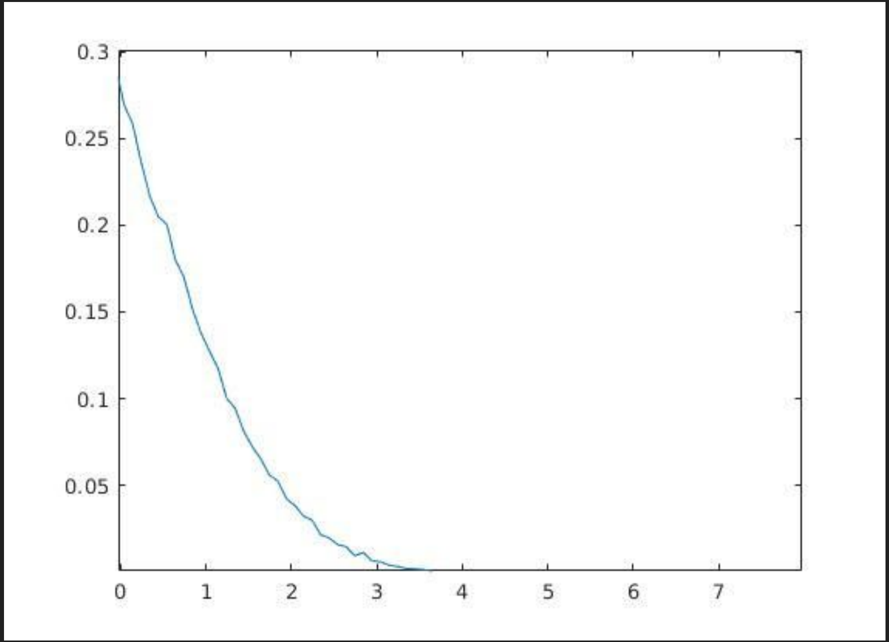
DECODING IN (4,2,3)

- In (4,2,3) the hamming distance is 3 thus we can correct a maximum of 1 symbol error across a binary symmetric channel.
- If one symbol is wrong, then it has three possible cases ($1, w, w^2$) since we can correct all cases of one-symbol error, there are $4*3=12$ possible cases that can be corrected.
- A lookup table with respect to these error codewords can be generated and their corresponding syndrome vectors can be calculated.
- Since the Syndrome vector has 16 possible values, 3 more cases of error are decoded. All these error cases are two symbol error cases.

SIMULATION RESULTS OF (4,2,3)



NOT NORMALIZED. CANNOT BE OVERLAPPED TO SCALE WITH PREV. GRAPHS



NORMALIZED. CAN BE COMPARED WITH PREVIOUS GRAPHS TO SCALE

SCOPE OF IMPROVEMENT

We have also proposed a different code, namely the (8,4,5) code on F_8 with a generator matrix as follows:

1	0	0	0	1	1	1	1
0	1	0	0	a	a^2	a^3	a^4
0	0	1	0	a^2	a^4	a^6	a
0	0	0	1	a^3	a^6	a^2	a^5

Where a is the primitive element of F_8 such that $1+a+a^3=0$.

Since this matrix is in systematic form, parity check matrix can be generated easily.Like the previous (4,2,3) code , we can decode this using syndrome decoding. Each element of the error lookup table must be generated iteratively (Since there are 1429 error codewords and two symbol error codewords are linear combinations of one symbol error combinations).

THANK YOU