Study of Undetected Error Probability of BCH codes for MTTFPA analysis

Dunia Prieto
Rubén Pérez-Aranda
rubenpda@kdpof.com

IEEE 802.3bv Task Force - January 2015

Background & Objectives

- A binary BCH code is proposed to be used as component code within a Multi-Level Coset Code (MLCC) structure for gigabit ethernet over POF
- The parity introduced by the BCH code for error correction capability in the receiver can also be used to assert decoding failure when the decoder is not able to correct the received code-word
- Decoding failure assertion can be used by the PHY:
  - To indicate to MAC an ethernet frame is corrupted by using GMII RX_ER signal
  - To avoid bad packet delimiter detection that can produce either error propagation over ethernet packets that were received correctly, generation of extra packets that were not transmitted or packets overlapping
- As it will be demonstrated, FCS does not suffice to provide large enough Mean Time To False Packet Acceptance (MTTFPA), therefore detection capability of BCH code as part of the FEC will be necessary
- It is the main objective of this presentation to elaborate a method for correct computation of the Undetected Error Probability of BCH codes

Exact calculation of P_{ue}

- We are interested in the undetected error probability for bounded-distance decoding of binary BCH codes (e.g. Berlekamp algorithm) when they are used for both error correction and detection on a binary symmetric channel (BSC) with crossover probability p
- We are going to assume the equalizer provides an AWGN channel to decoder input and symbol hard detection is implemented, having a BSC at the demapper output
- In the following analysis it is also assumed that a bounded-distance decoder is used in conjunction with a t-error correcting (n,k) BCH code C with minimum distance d_{min} where $t = \lfloor (d_{\min} - 1)/2 \rfloor$
  - If a received word is within Hamming distance $\lfloor (d_{\min} - 1)/2 \rfloor$ of a codeword, the decoder selects that codeword as the one most likely to have been sent. If the selected codeword is not the one that was sent, we say that a **decoder error** has occurred
  - If there is no codeword within Hamming distance $\lfloor (d_{\min} - 1)/2 \rfloor$ of a received word, then a **decoder failure** is declared
- If the weight distribution $\{A_j\}$ for C is known, exact expressions can be obtained for the probabilities of decoder error and failure
  - Let us note that the probability of decoder error is exactly the undetected error probability

Exact calculation of P_{ue}

- The probability of undetected error for a bounded-distance decoder is [1]

$$P_{ue}(p) = \sum_{w=t+1}^{n} \phi(w) P_E(w) \qquad (1)$$

where $\phi(w)$ is the probability that a received word has weight w

$$\phi(w) = p^{w}(1-p)^{n-w} \begin{pmatrix} n \\ w \end{pmatrix}$$

and $P_E(w)$ is the decoder error probability which is defined as the ratio of number of words with weight w that lie within distance t from a codeword to the number of words with weight w in the whole vector space

$$P_E(w) = \frac{\displaystyle\sum_{s=0}^{t} \sum_{j=w-s}^{w+s} A_j \begin{pmatrix} n-j \\ \dfrac{s+w-j}{2} \end{pmatrix} \begin{pmatrix} j \\ \dfrac{s-w+j}{2} \end{pmatrix}}{\begin{pmatrix} n \\ w \end{pmatrix}}$$

Exact calculation of P_{ue}

- Another expression for the probability of undetected error for a bounded-distance decoder that appears in the literature is [2]

$$P_{ue}(p) = \sum_{j=d_{\min}}^{n} A_j \sum_{k=0}^{\lfloor (d_{\min}-1)/2 \rfloor} P_k^j \qquad (2)$$

  where P_k^j is the probability that a received word is exactly Hamming distance k from a weight-j binary codeword

$$P_k^j = \sum_{r=0}^{k} \binom{j}{k-r} \binom{n-j}{r} p^{j-k+2r} (1-p)^{n-j+k-2r}$$

- The weight distributions for most BCH codes are not known
  - It is necessary to examine either its 2^k codewords or the 2^{n-k} codewords of its dual code
    - The computation becomes practically impossible as n, k, and n-k become large.
- There are, however, a few useful results to be considered
  - The weight distributions for all double- and triple-error-correcting binary primitive BCH codes have been found

Estimation of BCH weight distribution

- There are a number of different theorems that helps in estimating the weight distribution of a BCH code
- One of the most important is the Peterson weight estimation, which it is not an upper or a lower bound but an approximation [3]
- **Peterson Estimation** : The weight A_j of a primitive BCH code of length n and error correction capability t can be approximated as

$$A_j \cong \frac{\binom{n}{j}}{(n+1)^t} \qquad (3)$$

- By using the estimated weights it is now possible to obtain an estimation of the undetected error probability of binary primitive BCH codes
  - However, for **m > 10** (n = $2^m - 1$ for binary primitive BCH codes), infinity values result from Peterson Estimation in MATLAB and it is not possible to calculate P_{ue}

Estimation of BCH undetected error probability

- In [1], it is shown that the undetected error probability of binary linear codes can be quite simplified and quantified if weight distribution of the code is binomial like
    - Large subclasses of binary primitive BCH codes have approximate binomial-like weight distribution
- For those BCH codes the following expression for the undetected error probability is derived

$$P_{ue}(p) \simeq 2^{-mt} \sum_{s=0}^{t} \binom{n}{s} \cdot \sum_{h=t+1}^{n} \binom{n}{h} p^{h}(1-p)^{n-h} \qquad (4)$$

- By using the formula above it is possible to calculate P_{ue} for binary primitive BCH codes with m > 10
- Let us prove the validity of equation (4) by applying it to calculate the undetected error probability of a number of binary primitive BCH codes and then comparing the estimated value for P_{ue} with the result obtained by using alternative methods

Estimation of BCH undetected error probability

Weight distribution for all double-error-correcting binary primitive BCH codes is known, then their exact undetected error probabilities can be calculated

Peterson estimation is good!!!!

(n,k,t) BCH code	p	P_{ue} by (1)	P_{ue} by (1) + (3)	P_{ue} by (2) + (3)	P_{ue} by (4)
(31, 21, 2)	1.26e-8	3.7e-21	4.6e-21	3.3e-21	4.4e-21
(127, 113, 2)	1.26e-8	3.2e-19	3.3e-19	3.1e-19	3.3e-19
(511, 493, 2)	1.26e-8	2.2e-17	2.2e-17	2.2e-17	2.2e-17
(511, 250, 31)	4.4e-3	-	1.6e-60	1.9e-61	1.6e-60
(1023, 708, 34)	4.4e-3	-	1.1e-58	3.3e-59	1.1e-58
(1023, 443, 73)	4.4e-3	-	1.2e-169	3.7e-172	1.2e-169

Equation (4) gives a good estimation of the undetected error probability

Conclusions

- An accurate approximation of the undetected error probability for bounded-distance decoding of binary primitive BCH codes that have binomial-like weight distribution when they are used for both error correction and detection on a binary symmetric channel has been presented (equation (4))
- Equation 4 is numerically stable for large BCH codes, therefore provides to us a method to calculate the P_{ue} of BCH for MTTFPA analysis
- The proposal for GEPOF is based on a shortened BCH code
- An (n_c,k_c) shortened BCH code is obtained by prepending l_c zeros to the k_c information bits, then using the corresponding primitive code to encode the resulting k-bit information block and finally removing the l_c leading zeros from the obtained n-bit codeword.
    - It consists of a subset of codewords of the primitive BCH code from which it was generated and that's why **the error correction and detection properties of the primitive BCH code are guaranteed for the corresponding shortened code**

References

[1] M.-G. Kim and J.H. Lee. “Undetected error probabilities of binary primitive BCH codes for both error correction and detection”. *IEEE Transactions on Communications*, vol. 44, no. 5, pp. 575-580, May 1996.

[2] S.B. Wicker. “Error Control Systems for Digital Communication and Storage”. Englewood Cliffs, NJ: Prentice Hall, 1995.

[3] R. Micheloni, A. Marelli and K. Eshghi. “Inside Solid State Drives (SSDs)”. Springer Series in Advanced Microelectronics 37.

Questions?