On Some Properties of the Bit Decoding Algorithms

Ali Abedi , Pragat Chaudhari and Amir K. Khandani
Dept. of Elec. and Comp. Eng., University of Waterloo, Waterloo, ON, Canada, N2L 3G1
e-mail: ali, pragat, khandani@shannon.uwaterloo.ca, Tel: 519-8851211, Fax: 519-8884338

Abstract— **In this paper[1], we study certain properties of the bit decoding algorithms for the case of linear binary codes . We focus on the probability distribution function (pdf) of the bit LLR using an AWGN channel model with BPSK modulation. We prove that the pdf of the bit LLR of a specific bit position is independent of the transmitted code-word. It is also shown that the pdf of a given bit LLR when the corresponding bit takes the values of zero and one are symmetric with respect to each other (reflection of one another with respect to the origin). Among other things, this result shows that the resulting binary channel will be symmetrical in the sense that the probability of error for zero and one will be the same. Another important result finds a sufficient condition on the code structure such that the pdf of the bit LLR for two given bit positions are the same. We find such a condition using the code automorphism group and show that for the important class of cyclic codes this sufficient condition is always satisfied. This means that any given two bit positions in a cyclic code have the same pdf for their bit LLR.**

I. Introduction

In the application of channel codes, one of the most important problems is to develop an efficient decoding algorithm for a given code. The class of Maximum Likelihood (ML) decoding algorithms are designed to find a valid code-word with the maximum likelihood value. The ML algorithms are known to minimize the probability of the *Frame Error Rate (FER)* under the mild condition that the code-words occur with equal probability. Another class of decoding algorithms, known as bit decoding, compute the probability of the individual bits and decide on the corresponding bit values independent of each other. This results in minimizing the value of the *Bit Error Rate (BER)*. Note that unlike ML algorithms, in the case of the bit decoding algorithms the collection of decoded bits do not necessarily form a valid code-word. The straightforward approach to bit (or more generally symbol in the case of non binary codes) decoding is based on summing up the probabilities of different code-words according to the value of their component in a given position under consideration. A number of research works have addressed the problem of finding bit decoding algorithms of a reduced complexity (as compared to an exhaustive method) assuming a soft decision at the channel output. An optimum symbol decoding rule is proposed in [2] which is still exhaustive, but uses the set of code-words of the dual code in the decoding process. This method results in a lower complexity as compared to an exhaustive search if the dual code has a smaller number of code-words. Two modifications of the basic exhaustive method is presented in [3]. It gives a set of necessary and sufficient conditions for achieving minimum symbol error probability decoding and uses these conditions to derive a non-exhaustive optimum decoding algorithm of a reduced complexity. Bit-by-bit soft-decision decoding of binary cyclic codes is considered in [4] where the authors have modified the optimum decoding rule so as to reduce the complexity while maintaining good performance.

The problem of decoding linear block code used over a binary symmetrical channel with a given cross over probability is considered in [5–8] where the objective is to minimize the probability of an information bit error. Reference [5] gives an optimal rule for selecting coset leaders. Seguin [6] notes that the probability of an information symbol being in error is a function of the generator matrix chosen when the decoder is fixed. Seguin shows how to choose the optimal generator matrix when a fixed standard array code has been selected. A more difficult problem of simultaneously choosing the generator matrix and decoder that minimizes the probability of an information bit error is considered by Dunning [7]. Tolhuizen and van Gils [8] show that the large number of computations required for Dunning's procedure can be reduced somewhat by using the automorphism group of the code. Some authors [9,10] have considered specific coset leader rules for use when cross over probability of BSC is small and the encoding is systematic. Elia and Prati [9] give a decoding strategy, and note that for some codes it outperforms minimum weight coset leaders for small cross over probability. Montgomery and Vijaya Kumar [10] give another improved (though still sub-optimal) decoding strategy.

In an early paper Posner examines the information bit error probability obtained by using a linear block code over an AWGN channel with low signal to noise ratio and hard decision decoding [11]. More recently, in [12] the performance of linear block codes is examined when used on AWGN channel with soft decision decoding. Some asymptotic expressions are derived in [13] for bit error probability under optimum decoding for the AWGN channels. There have been also some works on bounds and approximation on the bit error probabilities of decoding convolutional codes [14] and trellis codes [15].

Maximum Likelihood decoding algorithms have been the subject of numerous research activities, while bit decod-

[1] This work is financially supported by Natural Sciences and Engineering Research Council of Canada (NSERC) and by Communications and Information Technology Ontario (CITO).
This work is a continuation of [1]

ing algorithms have received much less attention in the past. The reason being that the bit decoding algorithms are known to offer a BER performance very close to that of ML algorithms, while they have a substantially higher level of decoding complexity. More recently bit decoding have received increasing attention, mainly due to the fact that they deliver soft output decision (reliability information) which can be advantageously exploited in both uncoded and coded systems. In 1993, a new class of channel codes, called Turbo-codes [16], were announced which have an astonishing performance and at the same time allow for a simple bit decoding algorithm. Due to the importance of Turbo-codes, there has been a growing interest among communication researchers to work on the bit decoding algorithms. Reference [17] provides a method (known as BCJR) to compute the bit probabilities of a given code using its trellis diagram. An efficient exact APP decoding algorithm based on coset decoding principle proposed in [18]. There are also some special optimum methods for bit decoding of linear block codes based on sectionalized trellis diagrams [19] and based on using the code-words of the dual code [20]. The main simplification of BCJR has been the SOVA (soft output Viterbi Algorithm) of Hoeher and Hagenauer [21] which is a sub-optimum solution. A reduced-search BCJR algorithm is also proposed in [22]. Other researches have been done on reducing complexity of bit decoding like early detection and trellis splicing in [23].
This paper is organized as follows, in section II the model used to analyze the problem is presented. All notations and assumptions are in this section. We prove some theorems on bit decoding algorithms in III. Applying these theorems to cyclic codes is discussed in IV. We conclude in section V.

II. Modeling

Assume that a binary linear code $\mathcal{C}$ with code-words of length N is given. We use notation $\mathbf{c^i} = (c_1^i, c_2^i, \ldots, c_N^i)$ to refer to a code-word and its elements. We partition the code into a subcode C_k^0 and its coset C_k^1 according to the value of k^{th} bit position of its code-words $\mathbf{c^i}$.

$$\begin{aligned} \forall \mathbf{c^i} \in \mathcal{C}\text{: if } c_k^i = 0 &\Longrightarrow \mathbf{c^i} \in C_k^0 \\ \text{if } c_k^i = 1 &\Longrightarrow \mathbf{c^i} \in C_k^1 \\ C_k^0 \cup C_k^1 = \mathcal{C} \quad , &\quad C_k^0 \cap C_k^1 = \varnothing \end{aligned}$$

We use the following operators on our code book.

$$\mathbf{c^i} \oplus \mathbf{c^j} = \text{Bit wise binary addition of two code-words}$$

Note that the subcode C_k^0 is closed under binary addition.
The modulation scheme used here is BPSK which is defined as mapping M :

$$\begin{aligned} M : \mathbf{c} \longrightarrow \mathbf{m(c)} \\ M \text{ maps } 0 \longrightarrow -1 \text{ and } 1 \longrightarrow \quad 1 \end{aligned}$$

The dot product of two vectors $\mathbf{a} = (a_1, a_2, \ldots, a_N)$ and $\mathbf{b} = (b_1, b_2, \ldots, b_N)$ is defined as:

$$\mathbf{a.b} = \sum_{k=1}^{N} a_k.b_k$$

If we modulate a code-word $\tilde{\mathbf{c}}$ using BPSK modulation scheme and send it through an AWGN channel we will receive $\mathbf{x} = \mathbf{m}(\tilde{\mathbf{c}}) + \mathbf{n}$, where $\mathbf{n}$ is a Gaussian noise vector which has zero mean elements of variance $\sigma_{\mathbf{n}}^2$. A common tool to express the bit probabilities in bit decoding algorithms is based on using the so-called Log-Likelihood-Ratio (LLR). The LLR of the k^{th} bit is defined by the following equation.

$$LLR(k) = \log \frac{P(\tilde{c}_k = 1|\mathbf{x})}{P(\tilde{c}_k = 0|\mathbf{x})} \qquad (1)$$

where $\tilde{c}_k$ is the value of k^{th} bit in the transmitted code-word. In the case of transmitting equally likely code-words over AWGN channel the bit LLR can be calculated as follows:

$$\begin{aligned} LLR(k) &= \log \frac{\sum_{\mathbf{c^i} \epsilon C_k^1} \exp\left(\frac{\mathbf{x.m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)}{\sum_{\mathbf{c^i} \epsilon C_k^0} \exp\left(\frac{\mathbf{x.m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)} \\ &= \log \frac{\sum_{\mathbf{c^i} \epsilon C_k^1} \exp\left(\frac{\mathbf{n.m(c^i)} + \mathbf{m(\tilde{c}).m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)}{\sum_{\mathbf{c^i} \epsilon C_k^0} \exp\left(\frac{\mathbf{n.m(c^i)} + \mathbf{m(\tilde{c}).m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)} \end{aligned} \qquad (2)$$

Given the value of bit LLR, decision on the value of bit k is made by comparing the LLR with a threshold of zero. We are interested in studying the probabilistic behavior of the LLR as a function of the Gaussian random vector $\mathbf{n}$. Assuming a linear code, we will show that the choice of $\tilde{\mathbf{c}}$ does not have any impact on the resulting probability distribution as long as the value of the k^{th} bit remains unchanged.
Lemma1 : Taking two different code-words $\mathbf{c^i}, \mathbf{c^j}$ and a noise vector $\mathbf{n} = (n_1, n_2, \ldots, n_N)$ we define a new sign changed vector $\mathbf{n}^s = (n_1^s, n_2^s, \ldots, n_N^s)$, where

$$n_k^s = (-1)^{c_k^i \oplus c_k^j} n_k \quad , \quad (k = 1, 2, \ldots, N)$$

Elements of this new noise vector and previous one are equal in positions where $\mathbf{c^i}, \mathbf{c^j}$ have the same value, and differ only in their signs elsewhere. Noting that the joint pdf of noise vector $\mathbf{n}$ is sign symmetrical these two noise vectors $\mathbf{n}, \mathbf{n}^s$ will have the same probability distribution. Noting that elements of modulated code-words are ± 1 then we can see $\mathbf{n}^s.\mathbf{m(c^i)}$ and $\mathbf{n.m(c^j)}$ will posses the same pdf, as the different sign of modulated code-words elements can be compensated by applying a sign change to noise vector.

III. Theorems

Using the above definitions and notation, we have the following theorems.

Theorem 1: The probability distribution of $LLR(k)$ is not affected by the choice of $\tilde{\mathbf{c}}$ as long as the value of the k^{th} bit remains unchanged.

Proof: Having chosen two different codes $\tilde{\mathbf{c}}^1$, $\tilde{\mathbf{c}}^2$ we form their bit LLR for k^{th} bit position using 2:

$$LLR^1(k) = \log \frac{\sum_{\mathbf{c^i} \epsilon C_k^1} \exp\left(\frac{\mathbf{n.m(c^i)+m(\tilde{c}^1).m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)}{\sum_{\mathbf{c^i} \epsilon C_k^0} \exp\left(\frac{\mathbf{n.m(c^i)+m(\tilde{c}^1).m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)} \quad (3)$$

$$LLR^2(k) = \log \frac{\sum_{\mathbf{c^i} \epsilon C_k^1} \exp\left(\frac{\mathbf{n.m(c^i)+m(\tilde{c}^2).m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)}{\sum_{\mathbf{c^i} \epsilon C_k^0} \exp\left(\frac{\mathbf{n.m(c^i)+m(\tilde{c}^2).m(c^i)}}{\sigma_{\mathbf{n}}^2}\right)} \quad (4)$$

As long as the value of the k^{th} bit remains unchanged both $\tilde{\mathbf{c}}^1$, $\tilde{\mathbf{c}}^2$ are in the same subset namely C_k^0 or C_k^1. No matter they are in which subset $\tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2$ will be in subcode C_k^0. We must show that both above LLR's have identical probability distribution. Now we define an endomorphism $\phi_{\mathbf{c}}$ on code $\mathcal{C}$ which permutes the code-words by adding code-word $\mathbf{c}$ to them:

$$\phi_{\mathbf{c}} : \mathcal{C} \longrightarrow \mathcal{C}$$
$$\phi_{\mathbf{c}} : \mathbf{c^i} \longrightarrow \mathbf{c^i} \oplus \mathbf{c}$$

Noting that $\tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2 \in C_k^0$ we use $\phi_{\tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2}$ to map the subcode C_k^0 onto itself.

$$\phi_{\tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2} : C_k^0 \longrightarrow C_k^0$$
$$\phi_{\tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2} : \mathbf{c^i} \longrightarrow \mathbf{c^i} \oplus \tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2$$

Note that this endomorphism will shuffle elements of subcode within itself. Applying this mapping to arguments of denominator of 3 we have :

$$\mathbf{n.m(c^i) + m(\tilde{c}^1).m(c^i)} \longrightarrow$$
$$\mathbf{n.m(c^i \oplus \tilde{c}^1 \oplus \tilde{c}^2) + m(\tilde{c}^1 \oplus \tilde{c}^1 \oplus \tilde{c}^2).m(c^i \oplus \tilde{c}^1 \oplus \tilde{c}^2)}$$

As C_k^0 is closed under addition the result of $\mathbf{c^i} \oplus \tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2$ is also in C_k^0 so we name it $\mathbf{c^j}$ hereafter. Hence the mapped denominator becomes :

$$\sum_{\mathbf{c^j} \epsilon C_k^0} \exp\left(\frac{\mathbf{n.m(c^j)+m(\tilde{c}^2).m(c^j)}}{\sigma_{\mathbf{n}}^2}\right)$$

Note that these $\mathbf{c^j}$'s are shuffled version of previous $\mathbf{c^i}$'s. Now using $\mathbf{n}^s$ defined in Lemma 1 we can say $\mathbf{n.m(c^j)}$ have the same pdf as $\mathbf{n}^s.\mathbf{m(c^i)}$. Recalling the property that the joint pdf of the components of the noise vector is not affected by a sign change of its coordinates we conclude that mapping of denominator of 3 to 4 will be compensated by a sign change of noise vector coordinates. Applying the mapping $\phi_{\tilde{\mathbf{c}}^1 \oplus \tilde{\mathbf{c}}^2}$ to the numerator of 3, the elements of coset shuffle with the same permutation which can be compensated by the same sign change of $\mathbf{n}$ as used in the case of the denominator. We conclude that 3 , 4 will posses the same pdf independent of the transmitted code-word. ∎

The following theorem explains the effect of a change in the specific value taken by bit k on the probability distribution of $LLR(k)$.

Theorem 2: The probability distribution of $LLR(k)$ for value of bit $k = 0$ *or* 1 are the reflections of one another through the origin (threshold point).

Proof: To proceed with the proof, assume that the elements of C_k^0 are mapped by adding a code-word $\mathbf{c}$ to them which contains a 1 in position k. This will change the value of bit k from zero to one. This operation results in each component in the set of the code-words $\mathbf{c^i} \in C_k^0$, to be exchanged with a counterpart element within the set of the code-words $\mathbf{c^i} \in C_k^1$. In this case, if we replace $\mathbf{n}$ by $\mathbf{n}^s$ which is the sign changed version of noise vector defined in Lemma 1, we interchange the values of numerator and denominator. Moreover, as $\mathbf{n}$ and $\mathbf{n}^s$ occur with the same probability, and due to the properties of logarithm, we conclude that a given value of $LLR(k)$ if $k^{th}\text{bit} = 0$ occurs with the same probability as $-LLR(k)$ if $k^{th}\text{bit} = 1$, and vice versa. Therefore, changing the value of bit k in the transmitted code-word is equivalent to inverting the sign of the random variable corresponding to $LLR(k)$. ∎

We will now concentrate on the conditions for two bit positions to have the same pdf for their bit LLR by examining the values of the LLRs in these positions. These conditions are presented in the following theorems. First we visit the definition of automorphism group which is used in the following theorems.

Let $\mathcal{C}$ be a binary linear code of length N. We define a permutation Π which simply permutes the elements of each code-word. The set of permutations which maps the code-book $\mathcal{C}$ onto itself, form a group and called Automorphism group of code $\mathcal{C}$.

Theorem 3: Consider two bit positions of a code-word, i, j such that $1 \leq i, j \leq N$, $i \neq j$. If there exists a permutation Π within Automorphism group of code $\mathcal{C}$ which transfers bit position i to j , the $LLR(i)$ and $LLR(j)$ possess the same probability distribution.

Proof: Note that such a permutation will map the arguments of the summation in the numerator of 3 to the arguments of the summation in the numerator of 4 (and similarly will map the arguments of the summation in the denominators of 3,4 to each other). Applying the permutation Π to the noise vector will not change the corresponding probability value. Therefore the effect of this permutation will be compensated by permuting the noise vector coordinates. Hence the probability distribution of the LLR will not change. ∎

Note that set of permutations form a group, It is clear that inverse of Π exists and transfers bit position j to i. The existence of the permutation to yield two bit positions with the same probability distribution for their LLR is our next concern.

IV. Cyclic codes

We can apply the last result in III to the class of cyclic codes as a good example for checking the existence of the desired permutation. Cyclic codes have many interesting properties that simplifies their analysis. In the following theorem we just use their definition which states that a shifted cyclic code is still a cyclic code.

Theorem 4: The permutation mentioned in theorem 3 exists for the class of cyclic codes.

Proof: Transferring bit position i to j $(i \leq j)$ is equivalent to shifting elements of the code-words $j - i$ times to the right. It is the property if cyclic codes that any times of shift results in another code-word. Hence this permutation in automorphism group of code $\mathcal{C}$ exists for cyclic codes. ∎

V. Conclusion

After showing that pdf of bit LLR is independent of the choice of transmitted code-word, we showed that this pdf is also symmetric. Then we examined two bit positions and presented a sufficient condition for those two bits to have the same probability distribution for their bit LLR. This condition is satisfied when there exists a permutation within the automorphism group of the code which transfers one bit position to the other. At last it was shown that the class of cyclic codes have this property.

References

[1] P. Chaudhari, ,"Analytical Methods for the Performance Evaluation of Binary Linear Block Codes", M.A.Sc. Thesis, Dept. of Elec. and Comp. Eng., University of Waterloo, Spring 2000.

[2] C.R.P. Hartmann, L.D. Rudolph ,"An Optimum Symbol-by-Symbol Decoding Rule for Linear Codes," *IEEE Transactions on Information Theory*, Vol.22, No.5, pp. 514-517, September 1976.

[3] T.Y. Hwang ,"Decoding Linear Block Codes for Minimizing Word Error Rate," *IEEE Transactions on Information Theory*, Vol.25, No.6, pp. 733-737, November 1979.

[4] L.D. Rudolph, C.R.P. Hartmann, T.Y. Hwang, N.Q. Duc ,"Algebraic Analog Decoding of Linear Binary Codes," *IEEE Transactions on Information Theory*, Vol.25, No.4, pp. 430-440, July 1979.

[5] A.B. Kiely, J.T. Coffey, M.R. Bell ,"Optimal Information Bit Decoding of Linear Block Codes," *IEEE Transactions on Information Theory*, Vol.41, No.1, pp. 130-140, January 1995.

[6] G. Seguin ,"Optimal Symbol Error Rate Encoding," *IEEE Transactions on Information Theory*, Vol.32, No.2, pp. 319-322, Marchy 1986.

[7] L.A. Dunning ,"Encoding and Decoding for the Minimization of Message Symbol Error Rates in Linear Block Codes," *IEEE Transactions on Information Theory*, Vol.33, No.1, pp. 91-104, January 1987.

[8] L.M.G.M. Tolhuizen, W.J. van Gils ,"A Large Automorphism Group Decreases the Number of Computations in the Construction of an Optimal Encoder/Decoder Pair for a Linear Block Code," *IEEE Transactions on Information Theory*, Vol.34, No.2, pp. 333-338, March 1988.

[9] M. Elia , G. Prati ,"On the Complete Decoding of Binary Linear Codes," *IEEE Transactions on Information Theory*, Vol.31, No.4, pp. 518-520, July 1985.

[10] B.L. Montgomery, B.V.K. Vijaya Kumar ,"On Decoding Rules to Minimize the Probability of Information Bit Errors," *IEEE Transactions on Information Theory*, Vol.34, No.4, pp. 880-881, July 1988.

[11] E.C. Posner,"Properties of Error-Correcting Codes at Low Signal-to-Noise Ratios," *SIAM J. of App. Math.*, Vol.15, No.4, pp. 775-798, July 1967.

[12] Chi-Chao Chao, R.J. McEliece, Laif Swanson ,E.R. Rodemich ,"Performance of Binary Block Codes at Low Signal-to-Noise Ratios," *IEEE Transactions on Information Theory*, Vol. 38, No.6, pp. 1677-1687, November 1992.

[13] C.R.P. Hartmann, L.D. Rudolph, K.G. Mehrotra ,"Asymptotic Performance of Optimum Bit-by-Bit Decoding for the White Gaussian Channel," *IEEE Transactions on Information Theory*, Vol.23, No.4, pp. 520-522, July 1977.

[14] Steven S. Pietrobon ,"On the Probability of Error of Convolutional Codes," *IEEE Transactions on Information Theory*, Vol.42, No.5, pp. 1562-1568, September 1996.

[15] E. Baccarelli, R. Cusani, G.D. Blasio ,"Performance Bound and Trellis-Code Design Criterion for Discrete Memoryless Channels and Finite-Delay Symbol-by-Symbol Decoding," *IEEE Transactions on Communications*, Vol.45, No.10, pp. 1192-1199, October 1997.

[16] C. Berrou, A. Glavieux, and P. Thitimajshima. ,"Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes (1)," *Proceedings IEEE International Conference on Communications*, Geneva, Switzerland, pp. 1064-1070, May 1993.

[17] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv. ,"Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Transactions on Information Theory*, Vol.20, pp. 284-287, March 1974.

[18] L. Ping, K.L. Yeung ,"Symbol-by-Symbol Decoding of the Golay Code and Iterative Decoding of Concatenated Golay Codes," *IEEE Transactions on Information Theory*, Vol.45, No.7, pp. 2558-2562, November 1999.

[19] Ye Liu, Shu Lin, M.P.C. Fossorier ,"MAP Algorithms for Decoding Linear Block Codes Based on Sectionalized Trellis Diagrams," *IEEE Transactions on Communications*, Vol.48, No.4, pp. 577-586, April 2000.

[20] Sven Riedel ,"Symbol-by-Symbol MAP Decoding Algorithm For High-Rate Convolutional Codes That Use Reciprocal Dual Codes," *IEEE Journal on Selected Areas in Communications*, Vol.16, No.2, pp. 175-185, February 1998.

[21] J. Hagenauer, P. Hoeher ,"A Viterbi Algorithm With Soft Decision Outputs and Its Applications," *Proceedings of IEEE GLOBECOM*, Dallas, TX, November 1989, pp. 47.1.1-47.1.6.

[22] V. Franz, J.B. Anderson ,"Concatenated Decoding With a Reduced-Search BCJR Algorithm," *IEEE Journal on Selected Areas in Communications*, Vol.16, No.2, pp. 186-195, February 1998.

[23] B.J. Frey, F.R.Kschischang ,"Early Detection and Trellis Splicing: Reduced-Complexity Iterative Decoding," *IEEE Journal on Selected Areas in Communications*, Vol.16, No.2, pp. 153-159, February 1998.