Bahçeşehir University

EEE-5043 Error Control Coding Term Project

Comparison of Hard Decision Decoding and Soft Decision Decoding according to the Hamming Distance

Alperen Ertürk 1505421
Miray Keskin 1500988

supervised by
Dr. Suzan Üreten

June 17, 2020

Abstract

This project investigates the efficiency comparison of binary codes within maximum likelihood soft decision and the maximum likelihood of hard decision decoding. (book) Communication systems improve itself all the time because people utilizing technology and they wanted to develop it effectively. Digitalization is the most important topic within the communication. People want to transmit their files from their computers to source destination as faithfully as possible and at this point, during the transmitting process, the system needs encoding and decoding.

Contents

Nomenclature

BER Bit Error Rate

HDD Hard Decision Decoding

MLD Maximum Likelihood Decoding

SDD Soft Decision Decoding

SNR Signal to Noise Ratio

1 Introduction

In common terms, people utilize communication through a process of encoding and decoding. Encoder basically means the person who wants to send a message and data, the communication process starts with encoding. Encoding is the first step of the communication system because all pieces of information have to be encoded for transmitting in the software systems. At this step, for encoding, the system creates a code word and by using those code words, some issues may occur even the system works automatically. Noise is the reason for those issues and noises have important roles in our communication. If the noises are few, messages can reach the destination without corrupting or with less corrupting. Pre-stage of the reaching destination, the code word that is created by the system at the beginning must be decoded again to complete the process. At this point there are several decoding techniques and those techniques affect our corruption. In this state, with this project, it is aimed to compare the decoding techniques which are Hard Decision and Soft Decision decoding with respect to Hamming Distance and observe the results in simulation. While using these systems, binary systems are used and in order to complete the communication process in ease we are using array algorithms with equations. Algorithms use in decoding techniques to find less corrupted code word for correct transmission. The right code word is important because it shows how many bit error rates do we have at the end of the transmit process via showing the simulation part interactively.

This project emphasizes the differences in hard decision decoding and soft decision decoding. Decoding algorithms are classified into two categories such are hard decision and soft decision algorithms. Hard decision decoding utilizes a binary system for the received information. (typically 0 or 1 in a binary code) The opposite way round, soft decision algorithms utilize symbols on the received. (efficient ınfo set decoding paper) Therefore, a soft decision decoder will typically perform better in the presence of corrupted data than it's hard decision counterpart. Minimum Hamming Distance is used for Hard Decision, meanwhile, minimum Euclidean Distance is used for Soft Decision Decoding. Soft decision decoders are often used in Viterbi decoders and turbo codes decoders. Hard decision decoding decodes using minimum Hamming distance rule and soft decision decoding uses minimum Euclidean distance. Hamming distance is related to the number of locations in which bivectors differ and those vector differences show BER(bit error rate). Due to files must not corrupt, BER has a significant role to transmit the data in communication systems. BER clearly shows us, the comparison of SDD and HDD differences for transmitting data without corrupting and which techniques are more fertile for transmitting the data.

In decoding, many different techniques can be used. If we focus on the sub-types of decodings like SDD and HDD, every little details are important for all steps of the transmitting process. To give an example; parity bit error, SNR ratio, BER etc.. All elaborations in the decoding genres can be compared and that differences may occur in various circumstances. Maximum Likelihood Decoding is a standard decoding technique. In this project especially focused on efficiency comparison of binary codes within maximum likelihood soft, hard decision decoding. In soft-decision decoding, the receiver quantizes received data and decoder uses likelihood information to decode. In maximum likelihood decoding technique, the maximum likelihood value of receiving all possible code words are calculated based on the received code word and the code word that has the maximum probability is chosen to be the one that was sent in the transmitter. If an error is detected, then we identify the most probable error and undo this error in order to correct the mistake.

Code words and restricted messages get a shape of a vector space each. That vector space of messages of all bit strings (or more commonly, symbol) of length n and space of all code words includes length m which is for all bit strings. The number of positions at the same dimension in which the bit strings vary is stated to be Hamming distance between two points.[4]Hamming Distance can be used SDD and HDD. Especially, to give an example, ın the hard decision decoding, Hamming Distance can be used with 0 and 1's and we can show the descriptions as a pattern with Hamming Distance Decoding. In the oldest application theory of error-correcting codes, measured hamming distance error introduced by noise over a channel when a message, especially order of bits are sent from sources to place of aim.

1.1 Problem Definition

1.2 Motivation

To focus on this genre, comparison topics generally are more compeller than others. To compare two different topics, students have to make deep research and each topic's sub-clusters have to be studied for functional use. After learning the most common and specific details, students can make a comparison with respect to sub-clusters. In this topic, the research started with searching encoding and decoding techniques and why it needs those techniques in the communication process. Then, in this study, subtitles for each type has included. After, In this research, it has tried to begin finding similarities and differences in decoding.

2 Soft Decision and Hard Decision Decoding

Decoding is a challenging task in the communication process because when the data is wanted to send it exposes to lots of noises and corruption. To give an example; within this system, when the system is sending a signal via using a channel to start a communication, the system is incorporated in the noise signals inequality by the correct communication. Then, the purpose of the receiver to complete the communication progress, determining the original signal without a noisy signal. To increase the power of a signal, system parameters could be changed like the receiver type, decision device choices, etc. An optimal receiver is one that the design to minimize the probability of an occurred decision error. At the optimal levels, there does not exist any other structure that can provide a lower probability of errors. If the decision device was chosen as an optimal version of the parameter, it may have less probability of error data in the system. In the communication process, if we change the system's parameters in steps, we can increase or decrease the power of a signal in parallel with the communication process. In every communication system, each process integrated with corrupted data. (SNR, BER, noise, etc...) In this project, while the SDD and HDD researched, the corrupted data will also researched with those. The reason is the system can not send messages without those affecting data in the communication process. The main goal in here is to protect the initial status of the message in the software system, and while doing that, it needs quite an attention to look at some notions like SNR and BER. In fact, the features of the system that used for sending messages have a significant role at this point, such as; memoryless system because, maximum likelihood decision rule consists of probabilities like priori probabilities and if the system has a memory, each probability can be affected.

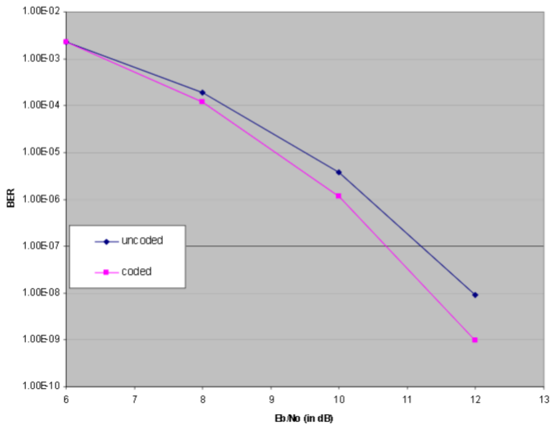

Figure 1: BER- decoding relation

In communication systems, decoding is a necessary because,without decoding message could faced with corruption. In Figure 1,ın the beginning of the graphic, uncoded datas and coded datas results are close each other. For a better comparison and aware of the importance of applying decoding in communication systems, end of the graph results are has to be comparison because for a exact consequences, BER ratio has to be high. That is why, uncoded version is not preferable in communication systems

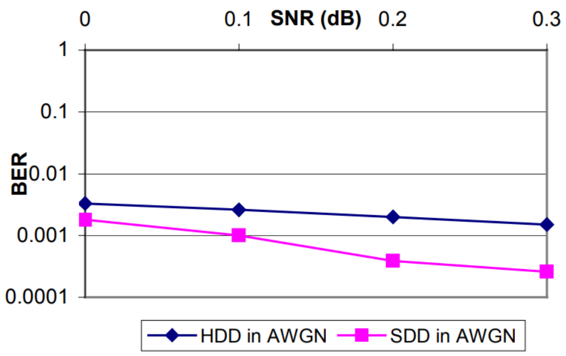

Figure 2: Performance of HDD and SDD in AWGN channel.

As it observed in Figure 2[6], relevant BER results related to SDD and HDD separately. All simulations are applied on an AWGN channel. The table clearly shows, SNR-BER ratio analysis is more similar between HDD and SDD. For a better comparison between two results, the progress of values could be checked. [6]

Not all origin faults might be caused by encoding, decoding, or channels during the communication process. In communication systems, the errors might also occur because of the messages such as the content, length and transmitting process. F.e, distance is the most important factor in the Maximum Likelihood Decision rule because this rule could be applied with Map Decision Rule. Meantime, ML Decision Rule is equivalent to the Map Decision Rule if the a priori probabilities are all equal.

In digital communication system, it is impossible to transact the actual analog voltages, behalf of it, the sampled voltages are quantized into m-bit number. Otherwise, in hard decision, a signal is quantized into a one-bit binary number.For a better explained distance decoding relation, In the receiving the code word 00, for instance ;

$$d11^2 = (1-0)^2 + (1-0)^2 = 1 + 1 = 2 \tag{1}$$

is consistent with the Hamming distance described above. Two different dimensional distance models are shown in figure 3 and figure 4[1].

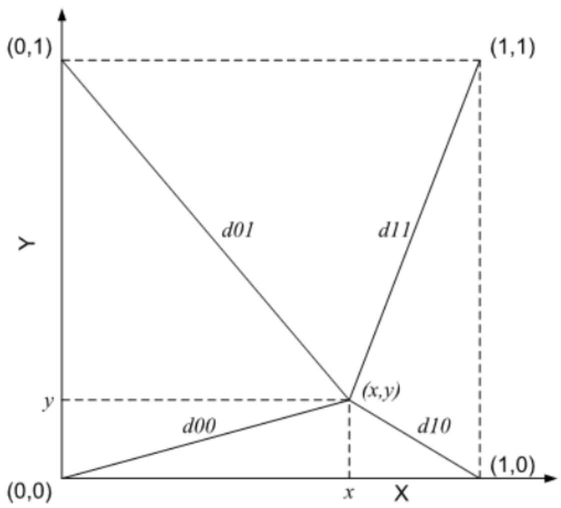

Figure 3: Distance model 2 symbol codeword represented in 2 dimension

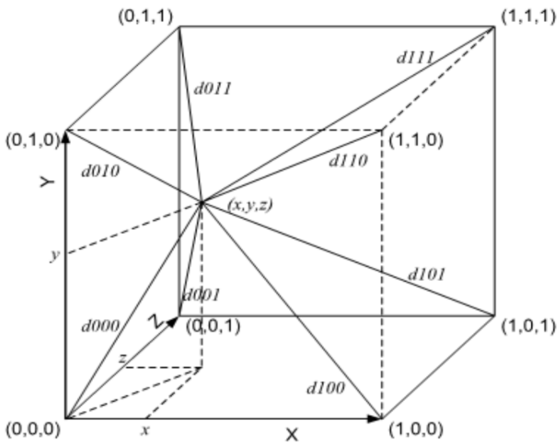

Figure 4: Distance model in 3 dimension 3 symbol code word representation

If an example should be given in this situation, while someone is sending a high-quality dissolvability picture, the system could cross with a high probability of error due to the lack of basic communication system availability to send higher quality of data. By contrasting with these errors, while sending short or long text messages and low-quality pictures, communication process faces with less probability of error even less than the sending high-quality resolution of the pictures. That is why the communication process should be researched in every single step to understand it deeply to avoid confusion. Hence, in this project, hard decision and soft decision decoding will be investigated step by step and they will be supported by simulations.

There are several decoding techniques, but the project process will continue with two of them, which are soft decision and hard decision decoding. In hard decision decoding, each bit represented with 1s and 0s. Hard decision decoding takes a stream of bits from the threshold stage and in the received side, when the bits exceed the threshold stage, become 1 otherwise decoding will be 0. In the other hand, soft decision decoding is shows a bits with a class of algorithms

3 Literature Survey

3.1 Viterbi Code Relation in Decoding

Soft decision decoding is a class of algorithms that takes a stream of bits or a block of bits and decodes them by considering a range of possible values that it may take. It considers the reliability of each received pulse to form better estimates of input data.Soft-decision decoders are often used in Viterbi decoders that are used for decoding convolutional codes.

The modulator alters the output of the encoder (which is digital), into a shape that is appropriate for the channel (Usually analog such as telephone channel).The demodulator tries an enterprise to recover the correct channel symbol in the existed noise. If the incorrect symbol is selected, the decoder tries to correct any errors that is in the result.because of the error coming from the channel, frequently utilizes, determines to correct such errors in encoder and decoder. The encoder, for detecting errors is including some bits in codeword like sender's bitstream and redundant bits. The most usable algorithm in encoding and decoding messages to be transmitted are respectively the convolutional coding and the Viterbi Algorithm which was proposed by Andrew J. Viterbi in 1967.[1]

3.2 Hamming Distance and Coding Relation

Hamming Distance is important in several ways. Channel memory, distance, or codeword relations could be criteria. In the definition of Hamming Distance, between two codewords is a simplified number of bit positions in which they differ. Many problems in communication systems depend on a reliable measure of the distance or similarity between data. That is why most frequently system use vectors likewise vector of bits. Such data structures apply entities as diverse as bitmaps that indicate the circumstances of terms and bitstrings indicating the presence of edges in distance.The popular distance measure application is Hamming Distance is defining a "Generalized Hamming distance" that extends the Hamming concept to give partial credit for near misses, and suggest a dynamic programming algorithm that permits it to be computed efficiently[4].A model for distance and decoding with minimum distance and their relations can be observed in figure 5.

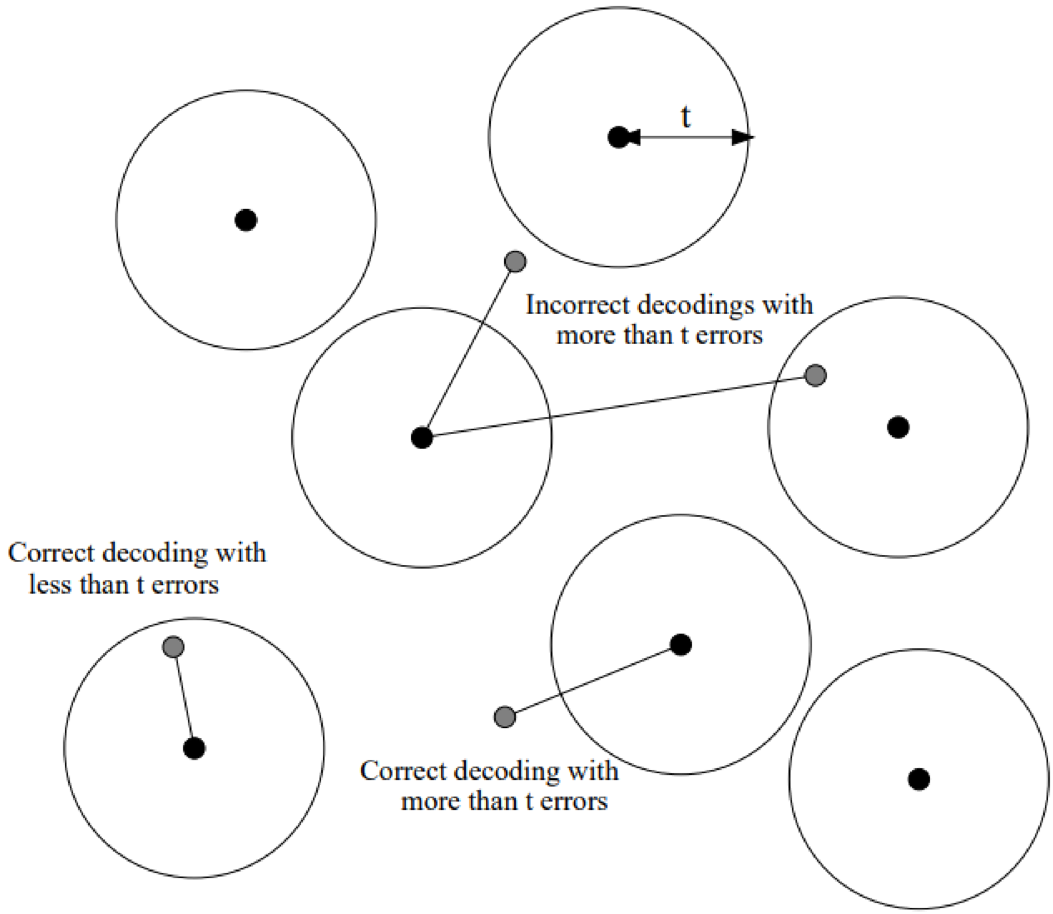

Figure 5: A Picture of Distance and Decoding with minimum distance (2t+1)

4 Simulation

```
Remain to process 0 samples ...
[ Wrong Bits | Correct Bits]

result_decod_sof =

     4   496

[ Wrong Bits | Correct Bits]

result_decod_hard =

   303   197

[ Wrong Bits | Correct Bits]

result_uncoded =

     1   499

[BER Soft |  BER Hard  | BER Uncoded ]

ber =

    0.0080    0.6060    0.0020
```

Figure 6: 500 sample simulation result

According to the matlab code that generates G matrix, which can be observed in figure 7; SDD, HDD and uncoded wrong-correct bits are shown in figure 6 with 500 samples.

```
systematic G matrix:
    1    0    0    0    0    0    0    0    0    0    0    1    0    0    1
    0    1    0    0    0    0    0    0    0    0    0    1    1    0    1
    0    0    1    0    0    0    0    0    0    0    0    1    1    1    1
    0    0    0    1    0    0    0    0    0    0    0    1    1    1    0
    0    0    0    0    1    0    0    0    0    0    0    0    1    1    1

 Table with  Cm words:
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
    0    0    0    0    1    0    0    0    0    0    0    0    1    1    1
    0    0    0    1    0    0    0    0    0    0    0    1    1    1    0
    0    0    0    1    1    0    0    0    0    0    0    1    0    0    1
    0    0    1    0    0    0    0    0    0    0    0    1    1    1    1
    0    0    1    0    1    0    0    0    0    0    0    1    0    0    0
    0    0    1    1    0    0    0    0    0    0    0    0    0    0    1
    0    0    1    1    1    0    0    0    0    0    0    0    1    1    0
    0    1    0    0    0    0    0    0    0    0    0    1    1    0    1
    0    1    0    0    1    0    0    0    0    0    0    1    0    1    0
    0    1    0    1    0    0    0    0    0    0    0    0    0    1    1
    0    1    0    1    1    0    0    0    0    0    0    0    1    0    0
    0    1    1    0    0    0    0    0    0    0    0    0    0    1    0
    0    1    1    0    1    0    0    0    0    0    0    0    1    0    1
    0    1    1    1    0    0    0    0    0    0    0    1    1    0    0
    0    1    1    1    1    0    0    0    0    0    0    1    0    1    1
    1    0    0    0    0    0    0    0    0    0    0    1    0    0    1
    1    0    0    0    1    0    0    0    0    0    0    1    1    1    0
    1    0    0    1    0    0    0    0    0    0    0    0    1    1    1
    1    0    0    1    1    0    0    0    0    0    0    0    0    0    0
    1    0    1    0    0    0    0    0    0    0    0    0    1    1    0
    1    0    1    0    1    0    0    0    0    0    0    0    0    0    1
    1    0    1    1    0    0    0    0    0    0    0    1    0    0    0
    1    0    1    1    1    0    0    0    0    0    0    1    1    1    1
    1    1    0    0    0    0    0    0    0    0    0    0    1    0    0
    1    1    0    0    1    0    0    0    0    0    0    0    0    1    1
    1    1    0    1    0    0    0    0    0    0    0    1    0    1    0
    1    1    0    1    1    0    0    0    0    0    0    1    1    0    1
    1    1    1    0    0    0    0    0    0    0    0    1    0    1    1
    1    1    1    0    1    0    0    0    0    0    0    1    1    0    0
    1    1    1    1    0    0    0    0    0    0    0    0    1    0    1
    1    1    1    1    1    0    0    0    0    0    0    0    0    1    0
```

Figure 7: Used random matrices for the 500 sample simulation

5 Conclusion

Through the whole project, we tried to explain the communication concept with differences especially between soft decision and hard decision decoding. At the beginning of the project, the work started with some fundamental notations for a better expression. HDD and SDD notations could affect extrinsic and internal factors. Because of that detail, we tried to show the effects of the most occurred factors on decoding techniques especially, since the simulation was beneficial to this purpose. Thus, in software explanations, Matlab has a strong effect on comparisons. That is why, in the Matlab part, we checked in different fields in decoding. In general, soft decision decoding has enough for communication with fewer errors and being able to estimate for a final product. SDD and HDD BER evaluation according to simulation seems correct, but for an extra, we also tried to implement uncoded BER too, the uncoded BER should increase with respect to time if it is compared with SDD, but the result can not be observed as time related, it calculates the total of it in the matrix.

A Conditional Vectors

[7]
function [condvects] = getcondvects(i)

g = 2;
i2 = 2^i;
$condvects = false(i2, i)$;
$form = 1 : 1 : i$
$m2 = 2^m$;
$m3 = (m2/2) - 1$;
$i3 = i - m + 1$;
$forg = g : m2 : i2$
$fork = 0 : 1 : m3$
$condvects(g + k, i3) = true$;
end
end
$g = m2 + 1$;
end
end

B HDD-SDD-Uncoded calculations

clc;clear all;close all;format short
n=15;k=5;
Pot=1e-5; vtx=1e5; no=2e-11; Eb=Pot/vtx;
Ec=(k/n)*Eb;
var$_n oise = sqrt(no/2)$;
$source_E c = sqrt(Ec)$;
$source_E b = sqrt(Eb)$;
$sample = input('Specifytheamountofsamplestocalculate :')$;

pg=[1 0 0 1 1]; Gsys = [];

disp('Sistematic G matrix:')
for j=n:-1:n-k+1 pol=zeros(1,j);
pol(1)=1;
coc,res
=deconv(pol,pg);
abs(res);
for t=1:length(res)
if mod(res(t),2)==0
res(t)=0;

else res(t)=1;
end
end

f=pol+res;
z=zeros(1,n-length(f));
fila=[z f];
Gsys = [Gsys ; fila];

end
Gsys

C=getcondvects(k);

Cm$_aux = zeros(2^k, n)$;
$forw = 2 : 2^k$
$auxcm1 = C(w, :)$;
$forvar = 1 : k$
$if(auxcm1(var) == 1)$
$Cm_aux(w, :) = Cm_aux(w, :) + Gsys(var, :)$;
$vec_cm = Cm_aux(w, :)$;

end
end
end
Cm$_aux$;
$disp('TablewithCmwords :')$;
$Cm = Cm_aux$;
$forfil_cm = 1 : 2^5$
$forcol_cm = 1 : n$
$if(mod(Cm(fil_cm, col_cm), 2) == 0)$
$Cm(fil_cm, col_cm) = 0$;
$elseif(Cm(fil_cm, col_cm) == 0)$
$Cm(fil_cm, col_cm) = 0$;
$else$
$Cm(fil_cm, col_cm) = 1$;
end
end
end

xp=[];
CM$_vec = []$;
$error_soft = 0$;

$correct_soft = 0;$
$ok_hard = 0;$
$error_hard = 0;$
$error_uncoded = 0;$
$ok_uncoded = 0;$
$restante = sample;$
$forj = 1 : sample$
$x = round(rand(1, 5));$
$forfil = 1 : 2^k$

if(x==Cm(fil,1:5))

xp=Cm(fil,:); x2=x;

for r$_col = 1 : nif(xp(1, r_col)) == 1vec_r(1, r_col) = source_Ec + (randn(1, 1)) * var_noise;$
$elsevec_r(1, r_col) = -source_Ec + (randn(1, 1)) * var_noise;$
end

end

for r$_col2 = 1 : 5if(x2(1, r_col2)) == 1vec_r2(1, r_col2) = source_Eb + (randn(1, 1)) * var_noise;$
$else$
$vec_r2(1, r_col2) = -source_Eb + (randn(1, 1)) * var_noise;$
end

end

vec$_{rh}d = vec_r;$
for fil$_CM = 1 : 2^k$
$forcol_CM = 1 : n$

aux1$_CM = 2 * (Cm((fil_CM), (col_CM))) - 1;$
$aux_CM_final = aux1_CM * aux2_CM;$
$CM_aux = aux_CM_final;$
$CM_vec = [CM_vecCM_aux];$
$CM_res = sum(CM_vec);$
$mat_CM(fil_CM, 1) = CM_res;$

end

CM$_vec = [];$

end

max$_C M = max(mat_C M);$
$index_m ax_C M = find(mat_C M == max(mat_C M));$

if(isequal(xp,Cm(index$_m ax_C M, :)) == 1)$

correct$_s oft = correct_s oft + 1;$

end
if(isequal(xp,Cm(index$_m ax_C M, :)) == 0)$

error$_s oft = error_s oft + 1;$

end

for c=1:n
if(vec$_{rh} d(c) < 0)$
$vec_{rh} d(c) = 0;$
$else$
$vec_{rh} d(c) = 1;$
end
end

cont$_h amming = 0;$

for fil$_h amming = 1 : 2^k$
$forcol_h amming = 1 : n$
$if(Cm(fil_h amming, col_h amming)\ = vec_{rh} d(1, col_h amming))$
$cont_h amming = cont_h amming + 1;$
$mat_h amming(fil_h amming, 1) = cont_h amming;$
end
end
$cont_h amming = 0;$
end

pos$_p alabra_d ecod_h d = find(mat_h amming == min(mat_h amming));$

if (isequal(Cm(pos$_p alabra_d ecod_h d(1), :), xp) == 1)$
$ok_h ard = ok_h ard + 1;$
end

if (isequal(Cm(pos$_p alabra_d ecod_h d(1), :), xp) == 0)$

$error_hard = error_hard + 1;$
end

vec$_r2_quantizate = vec_r2;$
$forc2 = 1 : 5$
$if(vec_r2_quantizate(c2) < 0)$
$vec_r2_quantizate(c2) = 0;$
$else$
$vec_r2_quantizate(c2) = 1;$
end
end

if(isequal(vec$_r2_quantizate, x) == 1)$
$ok_uncoded = ok_uncoded + 1;$
end

if(isequal(vec$_r2_quantizate, x) == 0)$
$error_uncoded = error_uncoded + 1;$
end

end

end

restante=restante-1;
clc
fprintf('Remain to process
end
disp('[Wrong Bits — Correct Bits]');
result$_decod_sof = [error_softcorrect_soft]$
$disp('[WrongBits|CorrectBits]');$
$result_decod_hard = [error_hardok_hard]$
$disp('[WrongBits|CorrectBits]');$
$result_uncoded = [error_uncodedok_uncoded]$
$disp('[BERSoft|BERHard|BERUncoded]');$
$ber = [error_soft/sampleerror_hard/sampleerror_uncoded/sample]$
$disp('SistematicGmatrix :');$
$disp(Gsys)$
$disp('TablewithCmwords :');$
$disp(Cm)$

xlabel('Eb/No (dB)')
ylabel('Bit Error Rate')

References

[1] Performance Analysis of Soft Decision Decoding and Hard Decision Decoding for Error Dependent Power Saving Viterbi Decoder for Mobile Devices, Offor Kennedy John, International Journal of Engineering Research and Technology (IJERT) Vol. 1 Issue 4, June 2012 .

[2] Maximum Likelihood Decoding on a Communication Channel,Cesar F. Caiafay , Nestor R. Barrazaz , Araceli N. Protoy ,XI Reunion de Trabajo en Procesamiento de la Information Control, 2007.

[3] Efficient Priority First Search Maximum Likelihood Soft-Decision Decoding of Linear Block Codes,Yunghsiang S. Han,IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 39, NO. 5, SEPTEMBER 1993 .

[4] Generalized Hamming Distance(Abraham Bookstein, Vladimir A. Kulyukin, Timo Raita)(Article in Information Retrieval · October 2002 DOI: 10.1023/A:1020499411651 · Source: OAI)

[5] Soft Decision and Quantised Soft Decision Decoding(M. Tomlinson et al., Error-Correction Coding and Decoding, Signals and Communication Technology, DOI 10.1007/978-3-319-51103-0-3)

[6] Comparison between Soft and Hard Decision Decoding using Quaternary Convolutional Encoders and the Decomposed CPM Model, Manjeet Singh,University of Cambridge

[7] Paul Metcalf (2020). Function To Generate Truth Table Condition Vectors (https://www.mathworks.com/matlabcentral/fileexchange/30008-function-to-generate-truth-table-condition-vectors), MATLAB Central File Exchange. Retrieved April 21, 2020.

[8] Lao, Yingjie, et al. "Reliable PUF-based local authentication with self-correction." IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 36.2 (2016): 201-213.

[9] Minimum- Weight Codewords in the (128,64) BCH Code. (L.D.Baumert- L.R Welch) 09-10.1977

[10] Analyses of error correction strategies for typical communication channels in watermarking Severine Baudry, Jean-François Delaigle, Bülent Sankur, Benoit Macq, Henri Maitre (Signal Processing 81 (2001) 12391250)