Digital Communication Lab

**E.C.E. DEPARTMENT**

**NIT CALICUT**

BCH ENCODING AND DECODING



**Submitted by-**

M Thanuja - B130531EC

M Jahnavi – B130619EC

M Supraja - B13382EC

N Kaushik – B130044EC

# 1 Objectives

Simulate a digital communication system that uses coherent BPSK with antipodal signaling and BCH code in the presence of Rayleigh fading channel and white noise and plot BER curves.

* Implementation of BPSK modulation with Error Control Coding
* Synthesizing a frequency flat Rayleigh fading channel and plot its PDF.
* Understanding the BER performance of uncoded digital modulation scheme in the presence of Rayleigh fading channel and AWGN channel
* Implementation of (15,7,5) BCH encoding.
* Analysis of BER performance of the modulation scheme with BCH code in the presence of Rayleigh Fading Channel
* Plotting of Ideal and Practical BER curves
* Find the *Eb/N*0 required for achieving a BER performance of 10−3 in all comparisons

# 2 Theory

**2.1 Rayleigh Fading Channel:**

The delays associated with different signal paths in a multipath fading channel change in an unpredictable manner and can only be characterized statistically. When the number of paths is very large, Central Limit Theorem can be invoked in order to model the time - varying impulse response of the channel as a circularly symmetric complex valued Gaussian random process. The channel is said to be **Rayleigh fading** channel if the impulse response is modeled as a zero - mean complex Gaussian process.

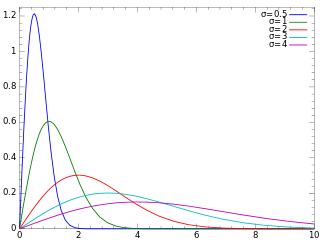


Figure 1: pdf of Rayleigh Fading Channel

• **Channel Model**

A circularly symmetric complex Gaussian random variable of the form,

*Z* = *X* + *jY* (1)

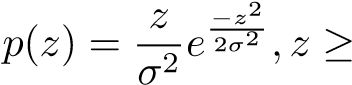
where *X* and *Y* are zero - mean independent and identically distributed (iid) Gaussian random variable. For a circularly symmetric complex random variable Z, we have:

*E*[*Z*] = *E*[*ejθZ*] = *ejθE*[*Z*] (2)

The statistics of a circularly symmetric complex Gaussian random variable is completely specified by its variance.

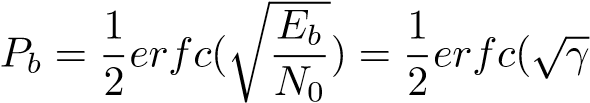
*σ*2 = *E*[*Z*2] (3)

The probability density of the magnitude |Z| is given by:

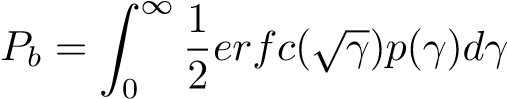
0 (4)

This is the *Rayleigh random variable*. This model is reasonable for an environment where there are a large number of reflectors.

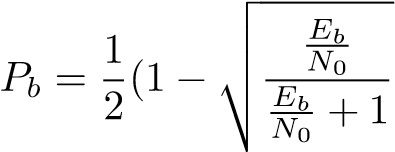
**BER for BPSK modulation in a Rayleigh Fading Channel:** The probability of error in BPSK modulation scheme in the presence of AWGN channel is given by:

) (5)

In the presence of the Rayleigh Fading Channel, the effective bit energy to noise ratio is . Hence, to find the bit error probability is determined by evaluating the conditional probability density function *Pb*|*h* over the probability density function of *γ*:

 (6)

This can be reduced to yield the following result:

) (7)

## 2.3 BCH Code

The BCH codes form a class of cyclic error-correcting codes that are constructed using finite fields.

There is a precise control over the number of symbol errors correctable by the code. In particular, it is possible to design binary BCH codes that can correct multiple bit errors.

They are easy to decode, using the algebraic Syndrome method.For any positive integers *m >*= 3 and *t <* 2*m*−1, there exists a binary code with block length *n* = 2*m* −1, number of parity-check digits *n*−*k <*= *mt* with a minimum distance *dmin >*= 2*t* + 1, called the t-error correcting BCH codes. The generator polynomial of this code is specified in terms of its roots from the Galois field GF (2*m*).

# 3 Implementation

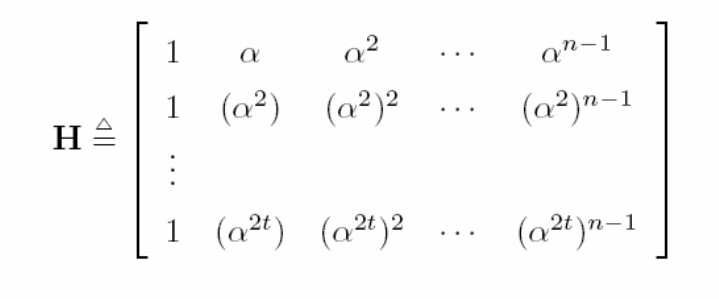
## 3.1 BCH Code

We are required to implement a (15, 7, 5) BCH code,which is 2- error correcting.

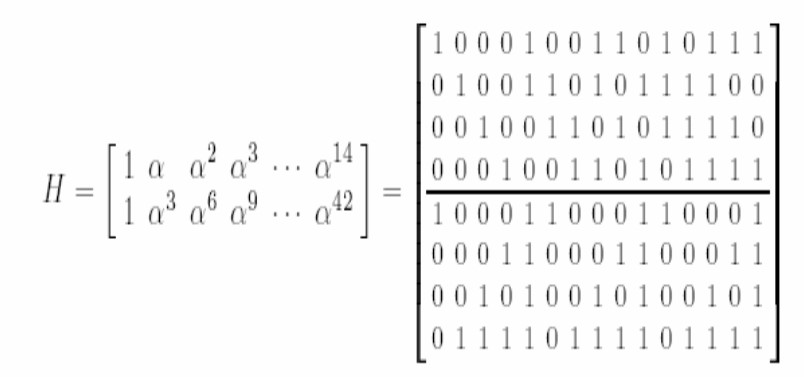
We first construct the generator polynomial for this code. Let *α* be a primitive element of GF(2*m*). The generator polynomial *g*(*x*) of the terror-correcting code of length 2*m*−1 is the lowest degree polynomial over GF(2) which has *α* , *α*2 ,*α*3, ...*α*2*t* and their conjugates as its roots.

If *φi*(*x*) is the minimal polynomial of *αi* , *g*(*x*) = *LCM*(*φ*1(*x*)*,φ*3(*x*)*,....,φ*2*t*−1(*x*))

We have the **H** matrix as



The entries of H are elements from *GF*(2*m*). Each element in *GF*(2*m*) can be represented by a m-tuple over *GF*(2). Then we get the binary parity check matrix for the code. Therefore **H** for our above problem is :

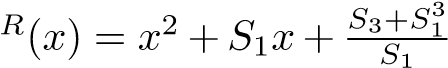


**BCH decoding**

For a t error correcting BCH code, the syndrome is a 2t-tuple, *S* = (*S*1*,S*2*,...,S*2*t*) = *rHT*, where r is the received code vector.

The *ith* component of the syndrome is *Si* = *r*(*αi*).

The decoding procedure for 2-error correcting code is as follows:

* + Compute *S*1 and *S*3.
  + If *S*1 = 0 and *S*3 = 0 , then assume no error.
  + If , then assume 1 error. Error location is *i*1 satisfying *X*1 = *αi*1.
  + If *S*1 6= 0 and, then assume 2 errors. Error location numbers are the zeros in *GF*(2*m*) of Λ*R*(*x*), where Λ Error locations are *i* satisfying *X* = *αi*.
  + For all other conditions, more than two errors have occurred and these are uncorrectable.

# 4 MATLAB Code

clear all;ctr=0; n=15; k=7; t=2; N=k\*1000;

ebyn0db = 0 : 2 : 20;

ebyn0 = 10.^(ebyn0db/10);

ip=rand(1,N)>0.5; si=2\*ip-1;

g=[1 0 0 0 1 0 1 1 1 0 0 0 0 0 0];

gen=[];

for i=0:k-1

gg= circshift(g,i); gen=[gen gg];

end

gen=gen'; a=0:15; b=gf(a,4); alpha=b(3); A=[];

for i= 0: 15

aa= alpha^i;

A=[A aa];

end

s=reshape(ip,N/k,k);

coded = mod(s.\*gen,2);

code = reshape(coded',1,(N/k)\*n);

BER = zeros(1,length(ebyn0db));

nn = randn(1,N) + 1i\*randn(1,N);

nn1 = randn(N/k,n) + 1i\*randn(N/k,n);

for ii=1: length(ebyn0db)

rxd1 = si + 10^(-ebyn0db(ii)/20)\*nn; rxd = real(rxd1)>0; coded\_n = (2\*coded-1) + 10^(-ebyn0db(ii)/20)\*nn1; coded\_n = real(coded\_n) >0;

rcv=[];

for i=1:N/k pos=find(coded\_n(i,:));

for m=1:length(pos)

if m==1

s1= A(pos(m));s3= A(pos(m))^3;

else s1=s1+A(pos(m));s3=s3+(A(pos(m)))^3;

end

end

s1d=double(s1.x); s11=de2bi(s1d,2\*t); s3d=double(s3.x);

if (s1==b(1)) && (s3==b(1))

rcv=[rcv coded\_n(i,:)];

elseif (s1 ~= b(1)) && (s3 == s1^3)

e=s11;

im = find(A(1:15) == s1);

rr=xor(coded\_n(i,:),[zeros(1,im-1) 1 zeros(1,n-im)]);

rcv=[rcv rr];

elseif (s1 ~= b(1)) && (s3 ~= s1^3)

pr=(s3+s1^3)/s1; er=[];

for f=1:length(b)

if (b(f))^2 + s1\*b(f) + pr == b(1)

er = [er b(f)];

end

end

if isempty(er)

rcv=[rcv coded\_n(i,:)];

ctr=ctr+1;

else

e1 = find(A(1:15) == er(1));e2 = find(A(1:15) == er(2));

if(e1>e2)

xx= [zeros(1,e2-1) 1 zeros(1,e1-1-e2) 1 zeros(1,n-e1)];

elseif(e1<e2)

xx= [zeros(1,e1-1) 1 zeros(1,e2-1-e1) 1 zeros(1,n-e2)];

else

xx= [zeros(1,e1-1) 1 zeros(1,n-e1-1)];

end

rr= xor(coded\_n(i,:),xx);

rcv=[rcv rr];

end

else

rcv=[rcv coded\_n(i,:)];

end

end

rcv=reshape(rcv, n, N/k); y= length(find(xor(rcv,coded')));

BER(ii)= y/(n\*(N/k));

y1= length(find(xor(rxd,ip))); thber(ii) = y1/N;

end

%thber = 0.5\*erfc(sqrt(10.^(ebyn0db/10)));

%Rayleigh channel variance=0.2;

X = randn(1, N)+1i\*randn(1,N);

Y = randn(1, N)+1i\*randn(1,N);

R = sqrt(variance\*(real(X).^2 + real(Y).^2)); range = 0:0.1:3;

%Get histogram values and approximate it to get the pdf curve h = hist(R, range); approxPDF = h/(0.1\*sum(h)); %Simulated PDF from the x and y samples

%Theoritical PDF from the Rayleigh Fading equation theoretical = (range/variance).\*exp(-range.^2/(2\*variance)); figure; plot(range, approxPDF,’b\*’, range, theoretical,’r’); title(’Simulated and Theoretical Rayleigh PDF for variance = 0.5’) legend(’Simulated PDF’,’Theoretical PDF’) xlabel(’r’); ylabel(’P(r)’); grid;

BERr=[]; BERu=[];

nn = randn(N/k,n) + 1i\*randn(N/k,n); hn = randn(N/k,n) + 1i\*randn(N/k,n);

for ii=1:length(ebyn0db)

rxd1 = Y.\*si + 10^(-ebyn0db(ii)/20)\*X; rxd11 = rxd1./Y;

rxd = real(rxd11)>0;

coded\_nr1 = hn.\*(2\*coded-1) + 10^(-ebyn0db(ii)/20)\*nn;

coded\_nr11 = coded\_nr1./hn; coded\_nr = real(coded\_nr11) >0; rcv=[];

for i=1:N/k pos=find(coded\_nr(i,:))

for m=1:length(pos)

if m==1

s1= A(pos(m));s3= A(pos(m))^3

else

s1=s1+A(pos(m));s3=s3+(A(pos(m)))^3;

end

end

s1d=double(s1.x); s11=de2bi(s1d,2\*t); s3d=double(s3.x);

if (s1==b(1)) && (s3==b(1))

rcv=[rcv coded\_n(i,:)];

elseif (s1 ~= b(1)) && (s3 == s1^3)

e=s11; im = find(A(1:15) == s1);

rr=xor(coded\_nr(i,:),[zeros(1,im-1) 1 zeros(1,n-im)]);

rcv=[rcv rr];

elseif (s1 ~= b(1)) && (s3 ~= s1^3)

pr=(s3+s1^3)/s1; er=[];

for f=1:length(b)

if (b(f))^2 + s1\*b(f) + pr == b(1) er = [er b(f)];

end

end

if isempty(er)

rcv=[rcv coded\_nr(i,:)];ctr=ctr+1;

else

e1 = find(A(1:15) == er(1));e2 = find(A(1:15) == er(2));

if(e1>e2)

xx= [zeros(1,e2-1) 1 zeros(1,e1-1-e2) 1 zeros(1,n-e1)];

elseif(e1<e2)

xx= [zeros(1,e1-1) 1 zeros(1,e2-1-e1) 1 zeros(1,n-e2)];

else

xx= [zeros(1,e1-1) 1 zeros(1,n-e1-1)];

end

rr= xor(coded\_nr(i,:),xx); rcv=[rcv rr];

end

else

rcv=[rcv coded\_nr(i,:)];

end

end

rcv=reshape(rcv, n, N/k); y= length(find(xor(rcv,coded'))); BERr(ii)= y/(n\*(N/k)); y1= length(find(xor(rxd,ip)));

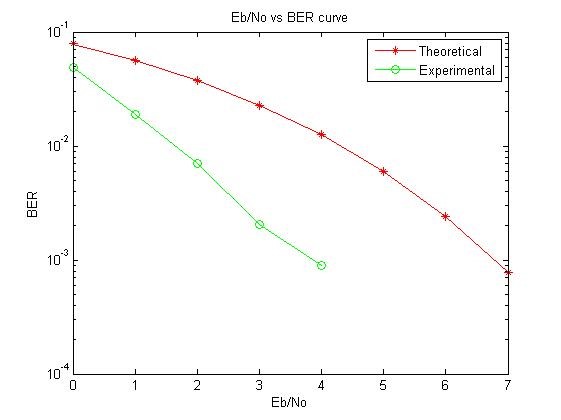
BERu(ii) = y1/N;

end

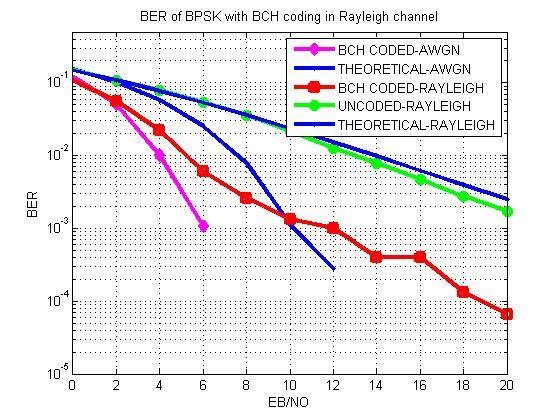
thray = 0.5.\*(1-sqrt(ebyn0./(ebyn0+1))); figure; semilogy(ebyn0db,BER,'md-','Linewidth',3);hold on; semilogy(ebyn0db,thber,'b.-','Linewidth',3);hold on; semilogy(ebyn0db,BERr,'rs-','Linewidth',3);hold on; semilogy(ebyn0db,BERu,'go-','Linewidth',3);hold on; semilogy(ebyn0db,thray,'b.-','Linewidth',3);grid on; axis([0 20 10^-5 0.5]); title('BER of BPSK with BCH coding in Rayleigh channel'); legend( 'BCH CODED-AWGN','THEORETI' );ylabel('BER');

# 5 MATLAB Simulation

The PDF of AWGN channel was plotted as in the figure below:



The BER plots of BPSK modulation scheme- theoretical and practical scenarios with and without BCH error coding and Rayleigh fading channel were obtained as follows:



# 6 Results

* BPSK modulation scheme was implemented in the presence of AWGN noise and Rayleigh fading channel
* BCH code was implemented for error detection and correction of the BPSK modulated signal.
* BER plots for ideal and practical scenarios were obtained for BPSK modulation scheme, BPSK modulation in the presence of Rayleigh fading channel, and also when implementing error detection and correction using BCH code.
* The performance of the various schemes were analyzed.
* The required SNR for BER of 10−3 was obtained as follows:
  + For the theoretical Rayleigh channel as well as transmission without error control coding for the Rayleigh channel, the required SNR is over 20dB.
  + The theoretical AWGN channel requires an SNR of about 10dB. In this simulation, the BCH coded Rayleigh channel signal also requires a similar SNR.
  + There is a significant reduction in the SNR required for the same

BER rate for the BCH coded AWGN channel, an SNR of only around 6dB. This results in a tremendous improvement in performance of the system.