

Aziz Amerul Faozi

18112042

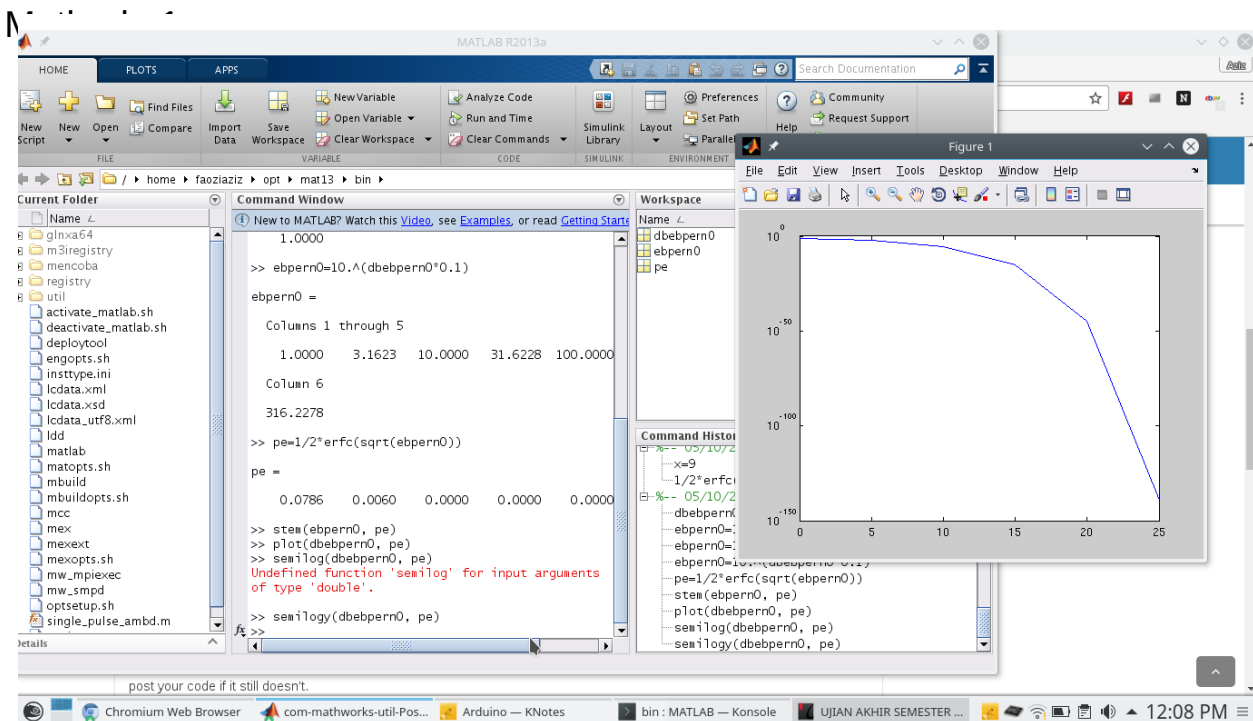
Nirkabel simulasi

Nirkabel

Bpk Adit Kurniawan

Nirkable Assigment

1. Membandingkan grafik fungsi ber dengan simulasi monte carlo.



Gambar 1: Grafik dengan menggunakan fungsi ber untuk BPSK.

Gambar 1 merupakan grafik dari plotting dengan menggunakan fungsi erfc untuk menghitung probabilitas error dengan fungsi standard.

Dengan menggunakan simulasi monte carlo saya mencoba menyimulasikan canal awgn bpsk tersebut.

% file : simulaber.m

% desc : untuk menghitung nomer 1 pake montecarlo

%

%jumlah trial

N_sim=input('Masukkan jumlah simulasi');

N = input('Ukuran blok simulasi');

%variable N error akan menjelaskan bahwa untuk mendapatkan akurasi yang

%tinggi maka distribusi ini harus dianalisa dengan persamaan binomial

%yang berantai karena probabilitas nya akan lebih akurat dengan banyaknya

%jumlah yang disimulasikan

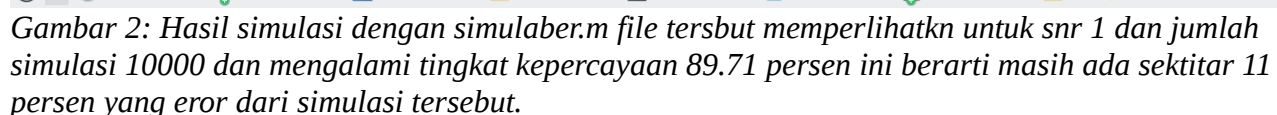
N_errors=input('Simulasi dari probabiliti yang terjadi lebih dari x error terjadi');

snr=input('Masukkan nilai SNRnya');

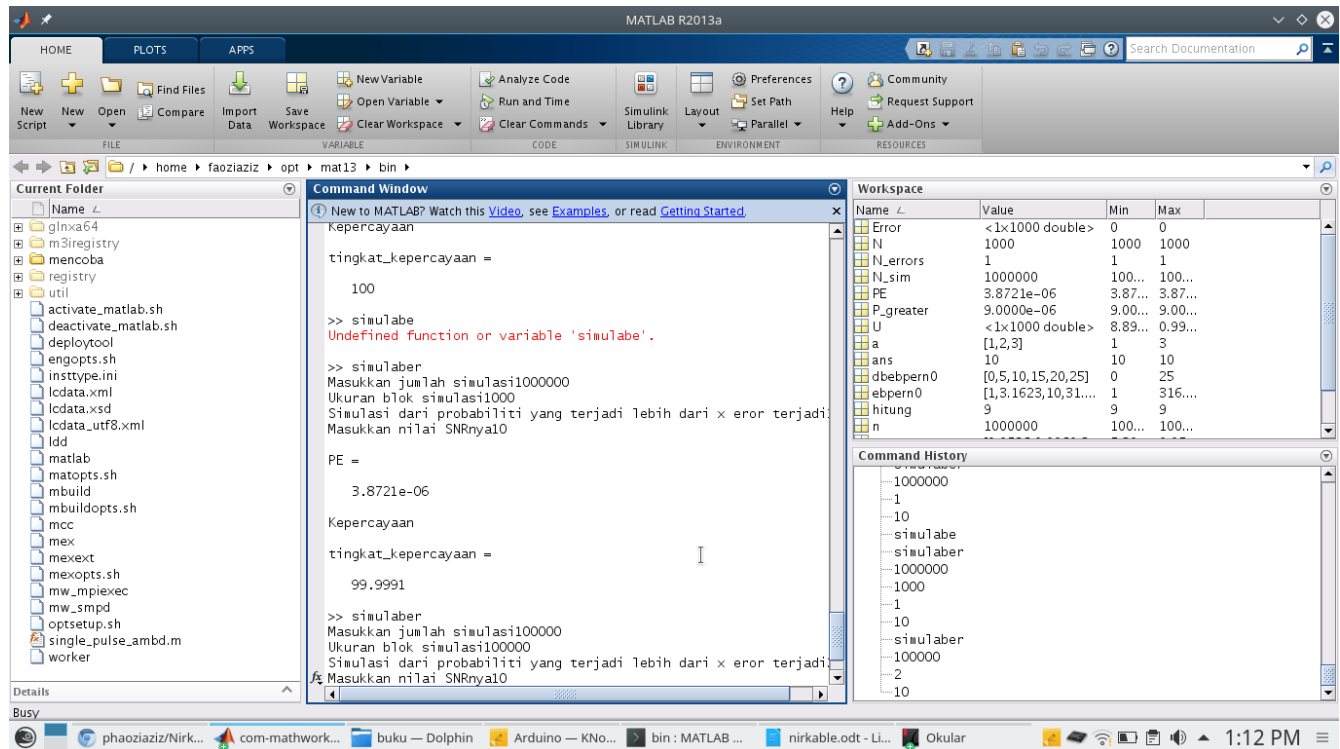
$$PE = 1/2 \cdot \text{erfc}(\sqrt{10^{(0.1 \cdot \text{snr})}})$$

```
U=rand(1,N);
```

```
display(tingkat_kepercayaan);
```

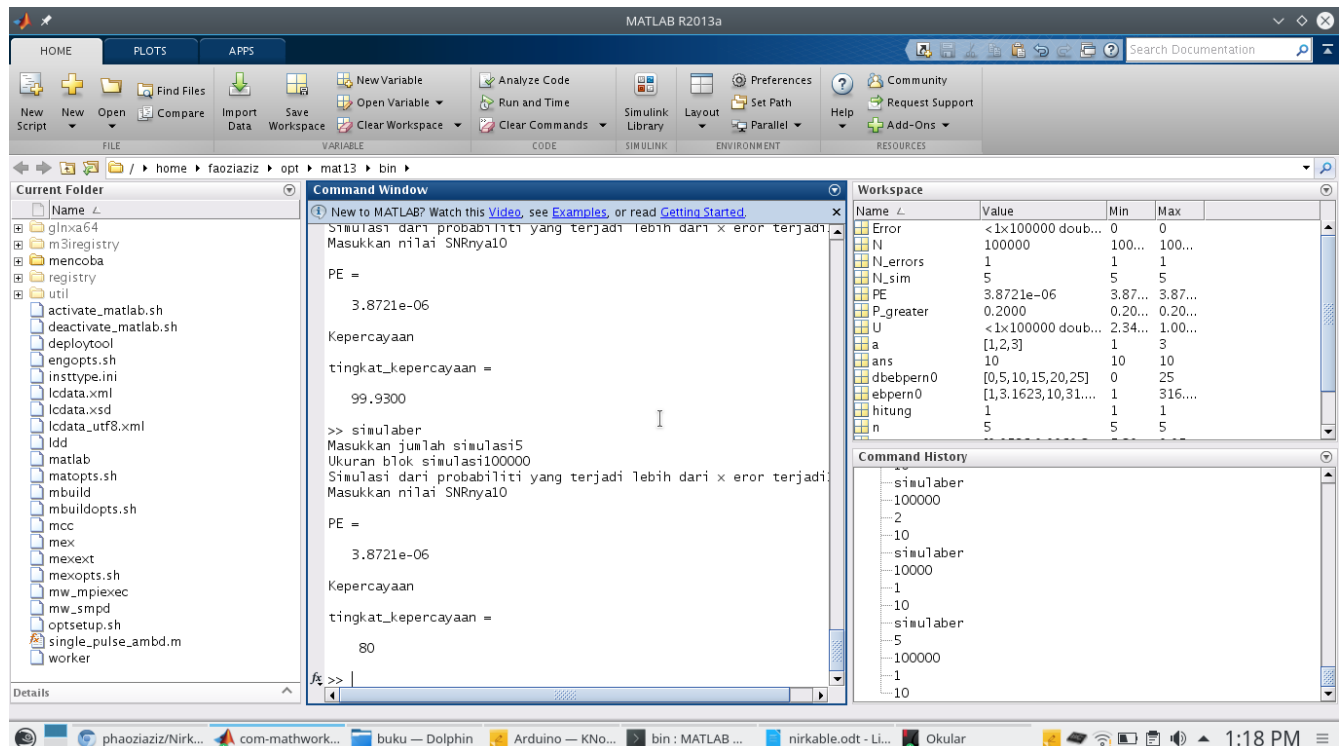


Dari gambar 2 terlihat bahwa simulasi dan probabilitas yang masih bisa terbaca di layar kaca. Sayangnya untuk simulasi yang dilakukan dengan menggunakan SNR lebih tinggi tidak mampu terbaca dengan baik dengan simulasi ini.



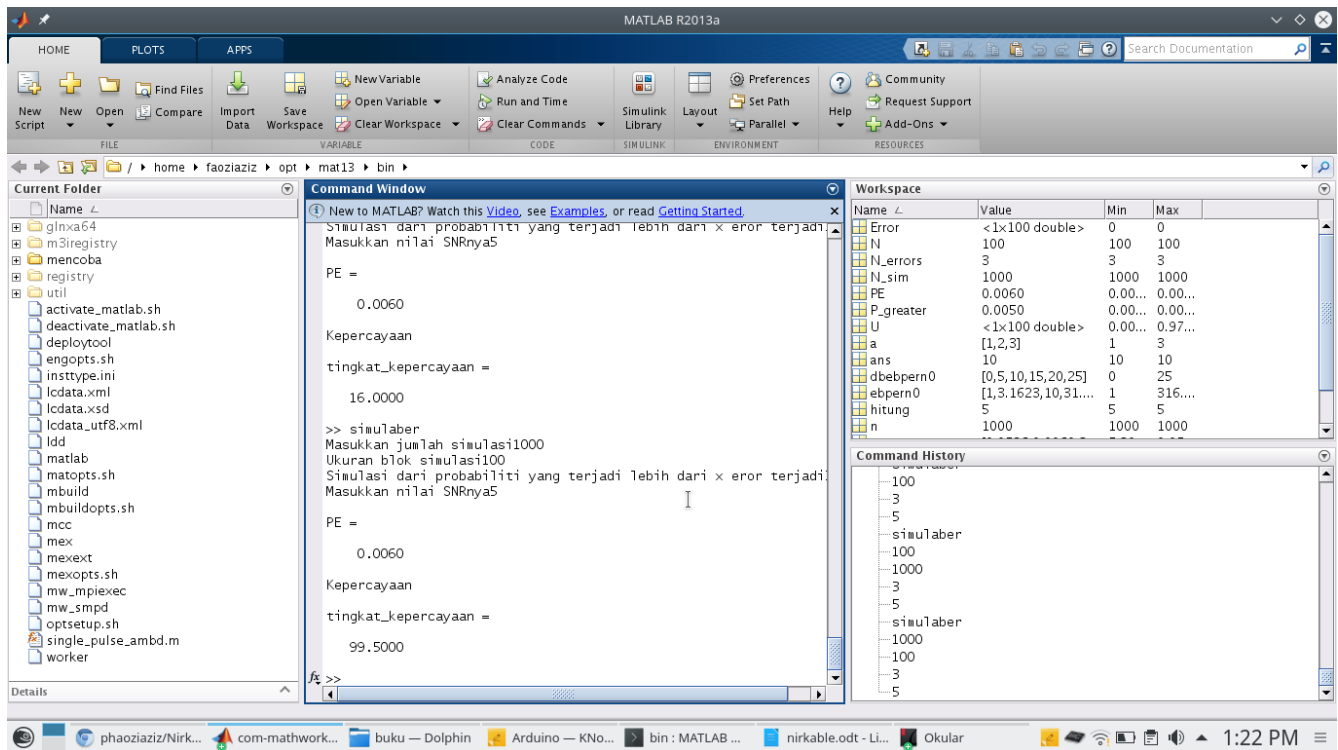
Gambar 3: Dalam simulasi yang dilakukan 1 juta kali terlihat error yang muncul

Gambar 3 terlihat bahwa mulai ada error yang muncul

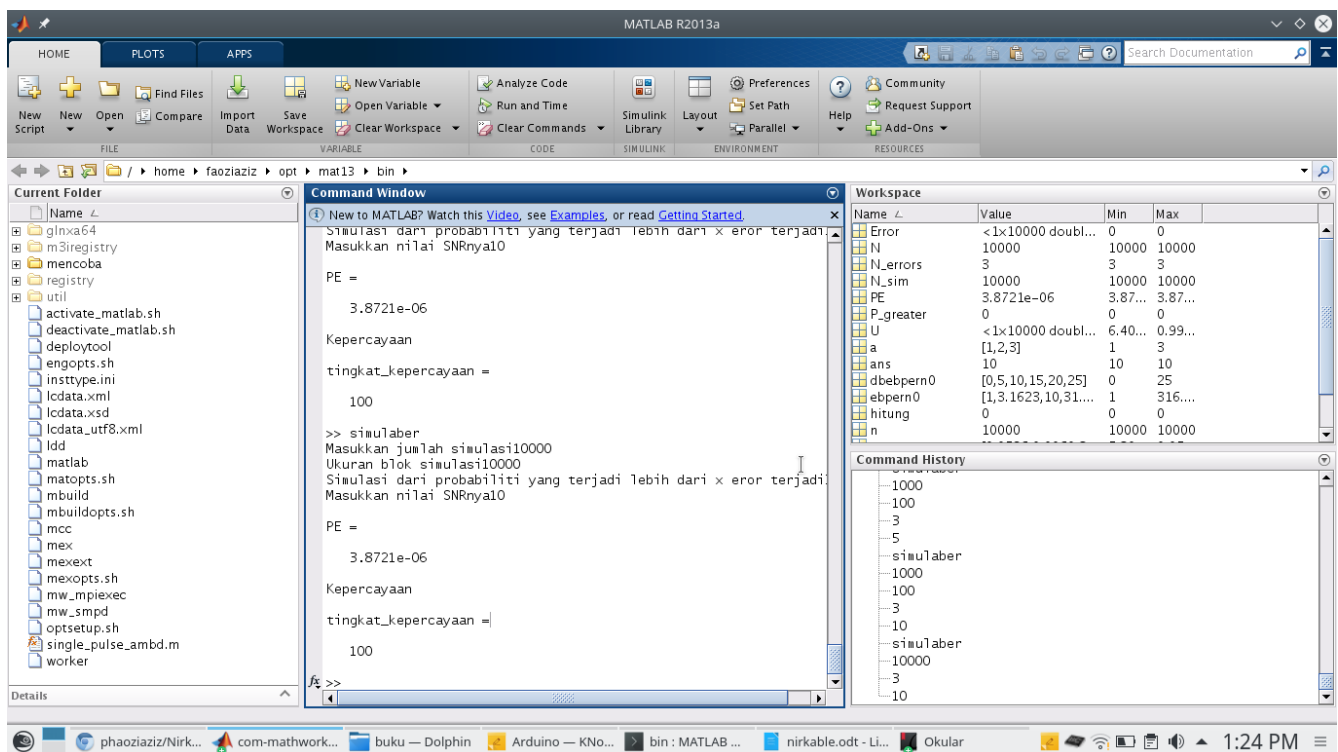


Gambar 4: Menambah blok simulasi 10000 dan melakukan simulasi selama 5 kali

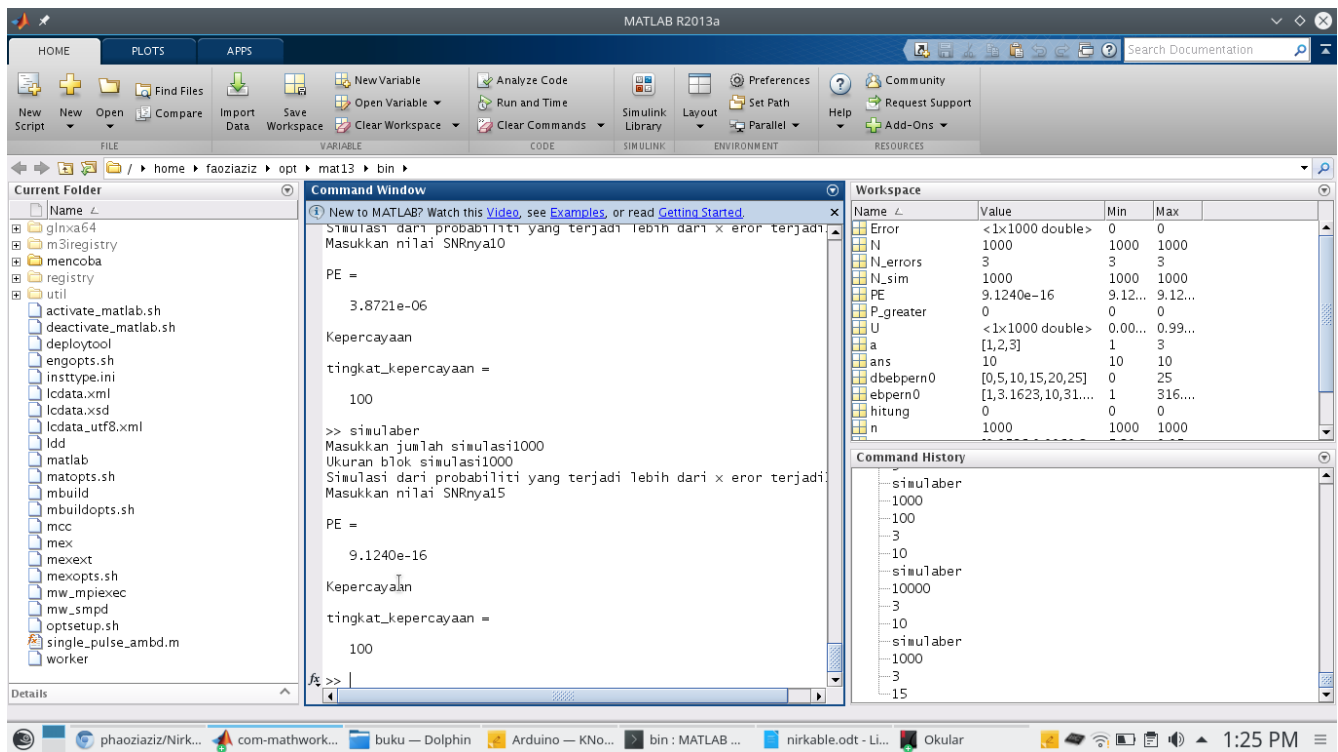
Dengan meningkatkan jumlah blok simulasi maka eror yang terjadi semakin sering, dan nilai ini akan meningkatkan akurasi yang lebih tinggi.



Gambar 5: Simulasi dengan SNR 5 db



Gambar 6: Simulasi untuk 10 db



Gambar 7: Simulasi untuk 15 db

Daftar Pustaka :

Ziemer, Rodger. E. 2014. “*Principle Communicartion 7th edition* page 290.”

Methode 2

Dengan menambahkan sinyal awgn noise

SNR_dB=5;

t=1:1:1000;

x=round(0.75*rand(1,1000));

rng('default');

y_costum=add_awgn_noise(x,SNR_dB);

rng('default');

y_inbuilt=awgn(x, SNR_dB, 'measured');

z=round(y_costum);

figure; subplot(1,2,1);

plot(t,x,'b', t,y_costum,'r');

legend('signal', 'signal with noise');

title('costum add awgn noise ');

Nomer 2

```
function y =fading(len, fd, T)
%jumlah
N =34;
N0=(N/2-1)/2;
alpha=pi/4;
xc=zeros(len,1);
xs=zeros(len,1);
sc=sqrt(2)*cos(alpha);
ss=sqrt(2)*sin(alpha);
ts=0:len-1;
ts=ts'.*T+round(rand(1,1)*10000)*T;
wd=2*pi*fd;
xc=sc.*cos(wd.*ts);
xs=ss.*cos(wd.*ts);
for lx=1:N0
    wn=wd*cos(2*pi*lx/N0);
    xc=xc+(2*cos(pi*lx/N0).*cos(wn.*ts));
    xs=xs+(2*sin(pi*lx/N).*cos(wn.*ts));
end;
y=(xc+i.*xs./sqrt(N0+1));

asa

%sumber
%http://www.gaussianwaves.com/gaussianwaves/wp-content/uploads/2015/06/How\_to\_generate\_AWGN\_noise.pdf
%
function [y, n]=add_awgn_noise(x, SNR_dB)
L=length(x);
SNR=10^(SNR_dB/10);
Esym=sum(abs(x).^2)/(L);
N0=Esym/SNR;
if(isreal(x))
    noiseSigma=sqrt(N0);
    n=noiseSigma*rand(1,L);
else
    noiseSigma=sqrt(N0/2);
    n=noiseSigma*(randn(1,L)+1i*randn(1,L));
end
y=x+n;
end
```

Maaf tidak saya tidak tahu diri dan terima kasih atas bantuannya

Aziz Faozi