

LAMPIRAN A  
LISTING PROGRAM

```

clc;
clear;
close all;
clc;

co=3*1e8;
fo=9*1e8;
disp(['ea_g atau ea_g utk Gaussian atau ea_j atau ms_j utk Jakes']);
disp(' ');
METHOD=input('Masukkan jenis metode yang digunakan = ');
disp(' ');
disp(['default : N_i = 25']);
disp(' ');
N_i=input('Masukkan jumlah fungsi harmonik = ');

sigma_0_2=1;
disp(' ');

v=input('Masukkan kecepatan unit mobile = ');
disp(' ');
% fo ditentukan untuk 900 Mhz
% fo=input('Masukkan frekuensi pemancar = ');
% disp(' ');
f_max=(v./co).*fo;
PHASE='rand';
PLOT=1;
f_c=sqrt(log(2)).*f_max;

if (METHOD == 'ea_g')

[f_i_n,c_i_n,theta_i_n]=parameter_Gauss_ku(METHOD,N_i,sigma_0_2,f_max,...
    f_c,PHASE,PLOT);
elseif (METHOD == 'ea_g')

[f_i_n,c_i_n,theta_i_n]=parameter_Gauss_ku(METHOD,N_i,sigma_0_2,f_max,...
    f_c,PHASE,PLOT);
else
    [f_i_n,c_i_n,theta_i_n]=parameter_Jakes_ku(METHOD,N_i,...
        sigma_0_2,f_max,PHASE,PLOT);
end;

```

```

function [f_i_n,c_i_n,theta_i_n]=parameter_Jakes_ku(METHOD,N_i,...
          sigma_0_2,f_max,PHASE,PLOT)

%-----

% parameter_Jakes_ku.m -----
%
% Program untuk menghitung frekuensi-frekuensi Doppler diskrit,
% koefisien Doppler, dan fasa Doppler dengan rapat spektral daya Jakes.
%
% Program m-file yang digunakan : acf_mue.m

%-----
% [f_i_n,c_i_n,theta_i_n]=parameter_Jakes(METHOD,N_i,sigma_0_2,...
%          f_max,PHASE,PLOT)
%-----

% Penjelasan dari parameter-parameter input :
%
% METHOD:
% |-----|-----|
% | Metode-metode untuk menghitung frekuensi          | Input |
% | Doppler diskrit dan koefisien Doppler              |       |
% |-----|-----|
% | Method of equal areas (MEA)                        | 'ea_j' |
% |-----|-----|

% N_i: jumlah fungsi harmonik

% sigma_0_2: daya rata-rata dari proses Gaussian real deterministik mu_i(t)

% f_max: frekuensi Doppler maksimum
%
% PHASE:
% |-----|-----|
% | Metode untuk menghitung fasa Doppler              | Input |
% | |                                                  |       |
% |-----|-----|
% |-----|-----|
% | Random Doppler phases                             | 'rand' |
% |-----|-----|
%
% PLOT: plot dari Fungsi Autokorelasi dan rapat spektral daya dari mu_i(t),
% if PLOT==1

if nargin<6,
    error('Not enough input parameters');

```

```

end

sigma_0=sqrt(sigma_0_2);

% Method of equal areas (MEA)
if METHOD=='ea_j',
    n=(1:N_i)';
    f_i_n=f_max/(2*N_i)*(2*n-1);
    c_i_n=2*sigma_0/sqrt(pi)*(asin(n/N_i)-asin((n-1)/N_i)).^0.5;
else
    error('Method is unknown')
end

% Perhitungan fasa Doppler:
PHASE=='rand',
theta_i_n=rand(N_i,1)*2*pi;

if PLOT==1
    figure;
    % subplot(1,2,1);
    stem([-f_i_n(N_i:-1:1);f_i_n],1/4*[c_i_n(N_i:-1:1);c_i_n].^2);
    grid;
    xlabel('f(Hz)');
    ylabel('PSD');
    legend('Estimasi rapat spektral daya (psd) Jakes');
    tau_max=N_i/f_max;
    % tau_max=N_i/(K*f_max);
    tau=linspace(0,tau_max,500);
    r_mm=sigma_0^2*besselj(0,2*pi*f_max*tau);
    r_mm_tilde=acf_mue(f_i_n,c_i_n,tau);
    figure;
    % subplot(1,2,2);
    grid on;
    plot(tau,r_mm,'r-',tau,r_mm_tilde,'b--');
    grid;
    xlabel('tau(s)');
    ylabel('ACF');
    legend('Nilai autokorelasi sebenarnya (teoritis)', 'Nilai estimasi fungsi
autokorelasi (acf) Jakes');
end;

```

```

function
[f_i_n,c_i_n,theta_i_n]=parameter_Gauss_ku(METHOD,N_i,sigma_0_2,f_max,...
    f_c,PHASE,PLOT)

%-----
% parameter_Gauss_ku.m -----
%
% Program untuk menghitung frekuensi-frekuensi Doppler diskrit,
% koefisien Doppler, dan fasa Doppler dengan rapat spektral daya Gaussian
%
% Program m-file yang digunakan : acf_mue.m

%-----
% [f_i_n,c_i_n,theta_i_n]=parameter_Gauss_ku(METHOD,N_i,sigma_0_2,...
%         f_max,f_c,PHASE,PLOT)
%-----
% Penjelasan dari parameter-parameter input :
%
% METHOD:
% |-----|-----|
% | Metode-metode untuk menghitung frekuensi      | Input |
% | Doppler diskrit dan koefisien Doppler          |       |
% |-----|-----|
% | Method of equal areas (MEA)                    | 'ea_g' |
% |-----|-----|
% N_i: jumlah fungsi harmonik
% sigma_0_2: daya rata-rata dari proses Gaussian real deterministik mu_i(t)
%
% f_max: frekuensi Doppler maksimum
% f_c: frekuensi cutoff 3-dB
%
% PHASE:
% |-----|-----|
% | Metode untuk menghitung fasa Doppler           | Input |
% |-----|-----|
% |-----|-----|
% | Fasa Doppler yang bersifat random              | 'rand' |
% |-----|-----|
%
% PLOT: plot dari Fungsi Autokorelasi dan rapat spektral daya dari mu_i(t),
% if PLOT==1

if nargin<7,
    error('Not enough input parameters');
end

```

```

sigma_0=sqrt(sigma_0_2);
% kappa_c=f_max/f_c;
kappa_c=sqrt(2./log(2)).*2; % edit tgl 180210
% Method of equal areas (MEA)
if METHOD=='ea_g',
    n=(1:N_i)';
    f_i_n=kappa_c*f_c/(2*N_i)*(2*n-1);
    c_i_n=sigma_0*sqrt(2)*sqrt(erf(n*kappa_c*...
        sqrt(log(2))/N_i)-erf((n-1)*kappa_c*...
        sqrt(log(2))/N_i));
%    K=1;

% Perhitungan fasa Doppler :
PHASE=='rand',
theta_i_n=rand(N_i,1)*2*pi;

if PLOT==1,
    figure;
% subplot(1,2,1)
    stem([-f_i_n(N_i:-1:1);f_i_n],...
        1/4*[c_i_n(N_i:-1:1);c_i_n].^2)
    grid
    xlabel('f (Hz)');
    ylabel('PSD');
    legend('Estimasi rapat spektral daya (psd) Gaussian');
    tau_max=N_i/(kappa_c*f_c);
%    tau_max=N_i/(K*kappa_c*f_c);
    tau=linspace(0,tau_max,500);
    r_mm=sigma_0_2*exp(-(pi*f_c/sqrt(log(2))*tau).^2);
    r_mm_tilde=acf_mue(f_i_n,c_i_n,tau);
    figure;
% subplot(1,2,2)
    plot(tau,r_mm,'r-',tau,r_mm_tilde,'b--')
    grid
    xlabel('tau(s)');
    legend('Nilai autokorelasi sebenarnya (teoritis)','Nilai estimasi autokorelasi
(acf) Gaussian');
    ylabel('ACF');
end;

%-----
% acf_mue.m -----
%
% program untuk menghitung ACF proses Gaussian deterministik mu_i(t)
%
```

```
%-----  
% r_mm=acf_mue(f,c,tau)  
%-----  
% Keterangan parameter input:  
%  
% f: frekuensi Doppler diskrit  
% c: koefisien Doppler  
% tau: time separation variable  
  
function r_mm=acf_mue(f,c,tau)  
  
r_mm=0;  
for n=1:length(c),  
    r_mm=r_mm+0.5*c(n)^2*cos(2*pi*f(n)*tau);  
end
```