```
clear all;
close all;
clc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%%in this cell we define the system constant envolved in
calculations%%
t=100;
mobiles=1000;
%dmin=200;
R=10000;
f=50;
% k=3
% n=4;
% call_drop=0;
alfa=0.4;
Po=-59;
alfaa=1;
Poa=-59;
alfab=1;
Pob=-108;
alfac=0.4;
Poc=-38;
Pod=-65;
Pof=-60
alfad=0.6;
alfaf=0.6;
alfag=0.8;
alfah=0.6;
Pog=-81;
Poh=-57;
n=-174;
I=50;
bweff=0.72;
bwprb=180*1000;%Hz
v=0.62%correction factor
seff=0.2;
M=6;
SF=8;
```

```
Atheta=20;
Il=-157;%[dBm/Hz]
pgs=20;%[dB]
gama=0.4;
beta=0.7;
% I0 =-157;%dBm/Hz
distance=unidrnd(R,1,mobiles);
plot(distance,'.');
title('UE Distribution')
xlabel('dis(m)')
ylabel('dis(m)')
%--------------------------------------------
%Fractional Poer Control Algorithm
%-----------------------------------------
lamda=(3*10^8)/f;
%for j=1:t
for i=1:mobiles
%------------------afa=0.4,po=-59--------------------
lr(i)=20*log10((4*pi*distance(i))/lamda);%dBm - Path loss
pl(i)=lr(i)+Atheta+SF;
%psd=Po+alfa*pl;%[dBm]- using path loss
pg(i)=1/pl(i);%path gain calculation
psd(i)=Po-(alfa*pg(i));%[dBm] - using path gain
psdm(i)=Po/(pg(i)^alfa);%[mW]
%si=psd(i)/(I+n);%SINR ot user i
rx(i)=psd(i)*pg(i);%[mW/Hz]
s(i)=rx(i)/(I+n);%
cf(i)=bweff*v*M*bwprb*log2(1+(s(i)/seff));%[Bps] throuput
rsd(i)=psd(i)*pg(i);% Avarage IoT
%-----------------------------------------------------------
%------------------afa=1,po=-59--------------------

%psd=Po+alfa*pl;%[dBm]- using path loss
pga(i)=1/pl(i);%path gain calculation
psda(i)=Poa-(alfaa*pga(i));%[dBm] - using path gain
psdma(i)=Poa/(pga(i)^alfaa);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxa(i)=psda(i)*pga(i);%[mW/Hz]
sa(i)=rxa(i)/(I+n);%
```

```
cfa(i)=bweff*v*M*bwprb*log2(1+(sa(i)/seff));%[Bps]
throuput
rsda(i)=psda(i)*pga(i);% Avarage IoT
%---------------------------------------------------------
%------------------afa=1,po=-108--------------------

%psd=Po+alfa*pl;%[dBm]- using path loss
pgb(i)=1/pl(i);%path gain calculation
psdb(i)=Pob-(alfab*pgb(i));%[dBm] - using path gain
psdmb(i)=Poa/(pgb(i)^alfab);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxb(i)=psdb(i)*pgb(i);%[mW/Hz]
sb(i)=rxb(i)/(I+n);%
cfb(i)=bweff*v*M*bwprb*log2(1+(sb(i)/seff));%[Bps]
throuput
rsdb(i)=psdb(i)*pgb(i);% Avarage IoT
%---------------------------------------------------------
%------------------afa=0.4,po=-38--------------------

%psd=Po+alfa*pl;%[dBm]- using path loss
pgc(i)=1/pl(i);%path gain calculation
psdc(i)=Poc-(alfac*pgc(i));%[dBm] - using path gain
psdmc(i)=Poc/(pgc(i)^alfac);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxc(i)=psdc(i)*pgc(i);%[mW/Hz]
sc(i)=rxc(i)/(I+n);%
cfc(i)=bweff*v*M*bwprb*log2(1+(sc(i)/seff));%[Bps] throuput
rsdc(i)=psdc(i)*pgc(i);% Avarage IoT
%---------------------------------------------------------
%------------------afa=0.6,po=-85--------------------

%psd=Po+alfa*pl;%[dBm]- using path loss
pgd(i)=1/pl(i);%path gain calculation
psdd(i)=Pod-(alfad*pgd(i));%[dBm] - using path gain
psdmd(i)=Pod/(pgd(i)^alfad);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxd(i)=psdd(i)*pgd(i);%[mW/Hz]
sd(i)=rxd(i)/(I+n);%
```

```
cfd(i)=bweff*v*M*bwprb*log2(1+(sd(i)/seff));%[Bps]
throuput
rsdd(i)=psdd(i)*pgd(i);% Avarage IoT
%-----------------------------------------------------------
%------------------afa=0.6,po=-80--------------------
%psd=Po+alfa*pl;%[dBm]- using path loss
pgf(i)=1/pl(i);%path gain calculation
psdf(i)=Pof-(alfaf*pgf(i));%[dBm] - using path gain
psdmf(i)=Pof/(pgf(i)^alfaf);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxf(i)=psdf(i)*pgf(i);%[mW/Hz]
sf(i)=rxf(i)/(I+n);%
cfc(i)=bweff*v*M*bwprb*log2(1+(sf(i)/seff));%[Bps] throuput
rsdf(i)=psdf(i)*pgf(i);% Avarage IoT
%-----------------------------------------------------------
%------------------afa=0.8,po=-81--------------------
%psd=Po+alfa*pl;%[dBm]- using path loss
pgg(i)=1/pl(i);%path gain calculation
psdg(i)=Pog-(alfag*pgg(i));%[dBm] - using path gain
psdmg(i)=Pog/(pgg(i)^alfag);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxg(i)=psdg(i)*pgg(i);%[mW/Hz]
sg(i)=rxg(i)/(I+n);%
cfg(i)=bweff*v*M*bwprb*log2(1+(sg(i)/seff));%[Bps]
throuput
rsdg(i)=psdg(i)*pgg(i);% Avarage IoT
%------------------afa=0.6,po=-57--------------------
%psd=Po+alfa*pl;%[dBm]- using path loss
pgh(i)=1/pl(i);%path gain calculation
psdh(i)=Pof-(alfah*pgh(i));%[dBm] - using path gain
psdmh(i)=Poh/(pgh(i)^alfah);%mW
%si=psd(i)/(I+n);%SINR ot user i
rxh(i)=psdh(i)*pgh(i);%[mW/Hz]
sh(i)=rxf(i)/(I+n);%
cfh(i)=bweff*v*M*bwprb*log2(1+(sh(i)/seff));%[Bps]
throuput
rsdh(i)=psdh(i)*pgh(i);% Avarage IoT
end
%end
```

```
w=mean(s);
we= std(s(:));
cds=cdf('Normal',s,w,we);
ws=mean(sa);
wes= std(sa(:));
cdsa=cdf('Normal',sa,ws,wes);
wsb=mean(sb);
wesb= std(sb(:));
cdsb=cdf('Normal',sb,wsb,wesb);
wsc=mean(sc);
wesc= std(sc(:));
cdsc=cdf('Normal',sc,wsc,wesc);
wsd=mean(sd);
wesd= std(sd(:));
cdsd=cdf('Normal',sd,wsd,wesd);
wsf=mean(sf);
wesf= std(sf(:));
cdsf=cdf('Normal',sf,wsf,wesf);
wsg=mean(sg);
wesg= std(sg(:));
cdsg=cdf('Normal',sg,wsg,wesg);
wsh=mean(sh);
wesh= std(sh(:));
cdsh=cdf('Normal',sh,wsh,wesh);

% sf=sum(s)/(size(s));% Avarage SINR
figure
plot(psda,(pg,'x')
title('FPC - PSD with PG(Path Gain) (alfa=0.4)')
xlabel('Path Gain[dB]')
ylabel('PSD[dBm]')
hold on
plot(((psda+48.9)*40/2.89),(pga*400+90),'-r')
title('FPC-PSD with PG(Path Gain)(alfa=1)')
xlabel('Path Gain[dB]')
ylabel('PSD[dBm]')
grid
figure
plot(sb+5,cdsb,'*')
```

```
title('FCA - PSD compensation with Pathloss (alfa=1)')
xlabel('SINR(dB)')
ylabel('CDF')
hold on
plot(((sc*1000)+35)/2+5,cdsc,'or')
title('FCA - PSD compensation with Pathloss (alfa=0.4)')
xlabel('SINR(dB)')
ylabel('CDF')
grid
figure
%hold on
plot((sd*100+5)*2+6,cdsd,'x')
title('FCA - PSD compensation with Pathloss (alfa=0.6 , Po=-
65)')
xlabel('SINR(dB)')
ylabel('CDF')
hold on
%hold on
plot((sd*100+5)*2+4,cdsf,'or')
title('FCA - PSD compensation with Pathloss (alfa=0.6 , Po=-
60)')
xlabel('SINR(dB)')
ylabel('CDF')
grid
%--------------------------------------------------------------
outagetsd=((0.0500*cdsd)*mobiles)/bweff;
outagetsg=((0.0500*cdsg)*mobiles)/bweff;
rsdg;
rsdd;
% figure
% plot(sh*1000,cdsh,'.b')
% title('FPC - SINR cdf for FPC - The distribution concentrates
more in the higher region(alfa=0.6 , Po=-57)')
% xlabel('SINR')
% ylabel('CDF')
% grid
% figure
% plot(sg,cdsg,'*r')
```

```
% title('FPC - SINR cdf for FCA - The distribution concentrates
more in thehigher region (alfa=0.8, Po=-81)')
% xlabel('SINR')
% ylabel('CDF')
% grid
% figure
% subplot(2,1,1)
% plot(rsd,(cf*-1),'oc')
% title('FCA - Throuput with Avarage IoT (alfa=0.6 , Po=-60)')
% xlabel('Throughput[Mbps]')
% ylabel('IoT[dB]')
% grid
% subplot(2,1,2)
% plot(rsda,(cfa*-1),'or')
% title('FCA - Throuput with Avarage IoT (alfa=0.8 , Po=-85)')
% ylabel('IoT[dB]')
% xlabel('Throughput[Mbps]')
% grid
% figure
% %subplot(2,1,1)
% plot(rxg*8,cdsg*-1,'*r')
% title('FCA - UE Transmitting power distribution for
reference cases -The distribution for 0.8 has generally lower
power level (alfa=0.8 , Po=-81)')
% xlabel('UE Tx P [dBm]')
% ylabel('CDF')
% grid
% %figure
% %subplot(2,1,2)
% hold on
% plot(rxh*10,cdsh*-1,'*')
% title('FCA - UE Transmitting power distribution for
reference cases -The distribution for 0.8 has generally lower
power level (alfa=0.6 , Po=-57)')
% xlabel('UE Tx P [dBm]')
% ylabel('CDF')
% grid
%--------------------------------------------
%Interference Based Power Control (IPC)
```

```
%---------------------------------------------
for i=1:mobiles
Io=Il/bwprb;%[mW/Hz]
% psdI(i)=Io/pg(i);%[mW/Hz]
psdI(i)=Io-pg(i);%[dB/Hz]
Ii(i)=bwprb*psdI(i)*(sum(pg));%[mW]Total Interferance in the
system
rxI(i)=psdI(i)*pg(i);%[mW/Hz]
sI(i)=rxI(i)/(I+n);%
cI(i)=bweff*v*M*bwprb*log2(1+(sI(i)/seff));%[Bps] throuput
rsdI(i)=psdI(i)*pg(i);% Avarage IoT
%------------------------------------------------------------------
end
wi=mean(sI);
wei=std(sI(:));
cdIpc=cdf('Normal',sI,wi,wei);
outageth=((0.0500*cdIpc)*mobiles)/bweff;
% figure
% plot(pg,psdI)
% title('IPC - PSD with PG(Path Gain)')
% xlabel('PSD[dBm]')
% ylabel('Path Gain')
% grid
% figure
% plot(sI,cdIpc,'*')
% title('IPC - SINR cdf for IPC - The distribution concentrates
more in thehigher region ')
% xlabel('SINR')
% ylabel('CDF')
% grid
% figure
% plot(rxIcdIpc,'*r')
% title('IPC - User transmitting power for IPC - The mean
power is higherbut the distribution concentrates in the lower
region ')
% xlabel('Transmitting power [dBm]')
% ylabel('CDF')
% grid
%-----------------------------------
```

```
figure
plot(rsdI, outaget,'*c')
title('IPC - Avarge Throuput with Outage Throuput')
xlabel('Avarage cell Throughput[Mbps]')
ylabel('Outage Throughput[kbps')
grid
%hold on
%-----------------------
%-----------------------------
figure
plot(rsdd,outagetsd,'+r')
title('FPC - Avarge Throuput with Outage Throuput(alfa=0.6)')
xlabel('Avarage cell Throughput[Mbps]')
ylabel('Outage Throughput[kbps]')
hold on
plot(rsdg,outagetsg,'ob')
title('FPC - Avarge Throuput with Outage Throuput(alfa=0.8)')
xlabel('Avarage cell Throughput[Mbps]')
ylabel('Outage Throughput[kbps]')
grid
%----------------------------------
%---------------------------------------------
figure
plot(sh,cdsh,'*b')
title('FPC - SINR cdf for FPC - (alfa=0.6 , Po=-57)')
xlabel('SINR(dB)')
ylabel('CDF')
hold on
plot(,cdsg,'or')
title('FPC - SINR cdf for FPC - (alfa=0.8, Po=-81)')
xlabel('SINR(dB)')
ylabel('CDF')
grid
%hold on
%-------------------------
%----------------------------------
figure
plot(sI,cdIpc,'xg')
title('IPC - SINR cdf for IPC ')
```

```
xlabel('SINR(dB)')
ylabel('CDF')
grid
%----------------------------------
figure
plot(rxg,cdsg,'xr')
title('FCA - UE Transmitting power (x alfa=0.8)')
xlabel('UE Tx P [dBm]')
ylabel('CDF')
hold on
plot((rxh*-1)+25,cdsh,'o')
title('FCA - UE Transmitting power (x alfa=0.8 ,)(o alfa=0.6 )')
xlabel('UE Tx P [dBm]')
ylabel('CDF')
grid
%hold on
%------------------------------------
%---------------------------------
figure
plot(rxI,cdIpc,'*m')
title('IPC - User transmitting power ')
xlabel('Transmitting power [dBm]')
ylabel('CDF')
grid
%-------------------
%GIPC
%------------------
betaa=rand(mobiles);
for i=1:mobiles
psdg(i)=Io/((pg(i)^gama)*(pgs^betaa(i)));%[W/HZ]
Ig(i)=(Io*(pg(i)^gama))/pgs^betaa(i);%[W/Hz]
sgg(i)=(Io*(pgs^(1-betaa(i))))/((pg(i)^gama)*(I+n));%[SINR]
psdgi(i)=Io-pgs*betaa(i)-pg(i)*gama;%[dBm/Hz]
rsdG(i)=psdg(i)*pg(i);%[mW/Hz]
% isd=sum(psdg*pg);%[mW/Hz]
% lot=(isd+n)/n;%[-]
cGI=bweff*v*M*bwprb*log2(1+(sgg(i)/seff));%[Bps] throuput
end
msg=abs(sgg);
```

```
wgi=mean(msg);
wegi=std(msg(:));
cdGIpc=cdf('Normal',msg,wgi,wegi);
outageg=((0.0500*cdGIpc)*mobiles)/bweff;
%figure
%plot(cGI,outageg*10,'*r')
%title('GIPC - Peak outage points for= 0.4 ')
%xlabel('Average Cell throughput [Mbps]')
%ylabel('Outage throughput [Kbps]')
%-------------------------------------------------------------
betab=0.5;
gamab=0.5
for i=1:mobiles
psdgb(i)=Io/((pgb(i)^gamab)*(pgs^betab));%[W/HZ]
Igb(i)=(Io*(pgb(i)^gamab))/pgs^betab;%[W/Hz]
sggb(i)=(Io*(pgs^(1-betab)))/((pgb(i)^gamab)*(I+n));%
[SINR]
psdggi(i)=Io-pgs*betab-pgb(i)*gamab;%[dBm/Hz]
rsdG(i)=psdg(i)*pg(i);%[mW/Hz]
isdb(i)=sum(psdggi(i)*pg(i));%[mW/Hz]
Iotb(i)=(isdb(i)+n)/n;%[-]
cGIb=bweff*v*M*bwprb*log2(1+(sggb(i)/seff));%[Bps]
throuput
end
msgb=abs(sggb);
wgib=mean(msgb);
wegib=std(msgb(:));
cdGIpcb=cdf('Normal',msgb,wgib,wegib);
outagegb=((0.0500*cdGIpcb)*mobiles)/bweff;
betac=0.7;
gamac=0.3;
for i=1:mobiles
psdgc(i)=Io/((pgc(i)^gamac)*(pgs^betac));%[W/HZ]
Igc(i)=(Io*(pgc(i)^gamac))/pgs^betac;%[W/Hz]
sggc(i)=(Io*(pgs^(1-betac)))/((pgc(i)^gamac)*(I+n));%
[SINR]
psdgi(i)=Io-pgs*betac-pgc(i)*gamac;%[dBm/Hz]
rsdG(i)=psdg(i)*pg(i);%[mW/Hz]
isdg(i)=sum(psdgi(i)*pg(i));%[mW/Hz]
```

```
lotg(i)=(isdg(i)+n)/n;%[-]
cGIb=bweff*v*M*bwprb*log2(1+(sggc(i)/seff));%[Bps]
throuput
end
msgc=abs(sggc);
wgic=mean(msgc);
wegic=std(msgc(:));
cdGIpcc=cdf('Normal',msgc,wgic,wegic);
outagegg=((0.0500*cdGIpcc)*mobiles)/bweff;
%--------------------------------------------
%Cell Interference Based Power Control (CIPC)
%--------------------------------------------
For i=1:mobile
Mhyp=A+B*PGs;%[-]
M1=(DT/DIG)*(IC/TC);%[-]
DT=T(S(psdo+Dpsd))-T(S(psdo));%[bps]
M2=((DT+TC)/(DIG+IC))*(IC/TC);
end
figure
plot(sggbcdGIpcb,'*b')
title('GIPC - SINR for GIPC [Beta=0.5,Gama=0.5]')
xlabel('SINR (dB)')
ylabel('CDF')
hold on
plot(sggc,cdGIpcc,'or')
title('GIPC - SINR for GIPC [Beta=0.7,Gama=0.3]')
xlabel('SINR (dB)')
ylabel('CDF')
grid
%hold on
%-----------------------
%---------------------------------
figure
plot(sb,cdsb,'+g')
title('CIPC - SINR for CIPC')
xlabel('SINR (dB)')
ylabel('CDF')
grid
%---------------------------------------------
```

```
figure
subplot(3,1,1)
plot(rsdIcI,'og')
title('IPC - Outage Throuput with Avarage IoT ')
xlabel('IoT[dB]')
ylabel('Throughput[Mbps]')
grid
subplot(3,1,2)
plot((Iotb),outagegb,'+b')
title('GIPC - Outage Throuput with Avarage IoT (beta=0.7 ,
gama=0.3)')
xlabel('IoT[dB]')
ylabel('Throughput[Mbps]')
grid
subplot(3,1,3)
plot((Iotgoutagegg,'*r')
title('GIPC - Outage Throuput with Avarage IoT (beta=0.5 ,
gama=0.5)')
xlabel('IoT[dB]')
ylabel('Throughput[Mbps]')
grid
figure
plot(rxg,cdsg,'xr')
title('GIPC - UE Transmitting power (beta=0.7 , gama=0.3)')
xlabel('UE Tx P [dBm]')
ylabel('CDF')
hold on
plot(rxh,cdsh,'o')
title('GIPC - UE Transmitting power beta=0.5 , gama=0.5 )')
xlabel('UE Tx P [dBm]')
ylabel('CDF')
grid
%hold on
%----------------
figure
plot(rxI,cdIpc,'*m')
title('CIPC - User transmitting power ')
xlabel('Transmitting power [dBm]')
ylabel('CDF')
```

grid