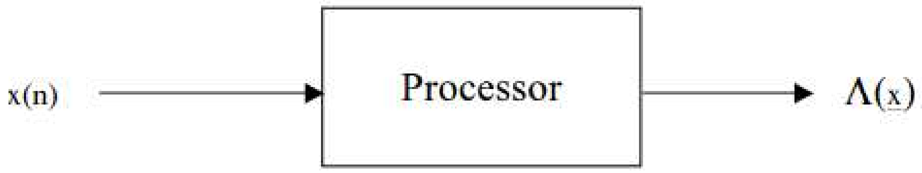
ECE 254 Homework 2

SKE and SKEP Processor Performance

Name: Mingxuan Wang

Date: 2015/10/10

- Title: SKE and SKEP Processor Performance
- Objective:



Our goal is to decide presence or absence of a sinusoid buried in white Gaussian noise

where: $x(n) = s(n) + n(n)$, $n = 0,1,\ldots, N-1$ and $N = 16$

$s(n) = A \sin(2\pi f_c n + \phi)$, $f_c = 1/16$

$n(n)$ is an uncorrelated, Gaussian noise sequence ~$N(0,1)$

$SNR = A^2/2\sigma^2$, $\sigma^2 = 1$.

1. Select values of A such that:

   a) $\frac{2E}{N_0} = 4$

   b) $\frac{2E}{N_0} = 9$

   And what are the corresponding SNR's?

2. Determine the ROC performance of the following processors both theoretically and via Monte Carlo simulation:

   a) Signal Known Exactly(SKE)

   b) Signal Known Except for Phase(SKEP)

When carrying out the simulations, generate estimates of $p(\Lambda|H_0)$ and $p(\Lambda|H_1)$ (or sufficient statistics for Λ), plot them, and from these (or their corresponding cumulative distribution functions) compute estimates of P_D and P_F. Compare your theoretical and simulation results.

Plot your ROC curves both on linear axes and on normal probability paper. See Chap. 2 in [1].

- Approach:

1. For the first problem:

$$\frac{2E}{N_0} = \frac{E}{N_0/2} = \frac{\frac{1}{2w}\sum_{n=0}^{N-1} s^2(n)}{N_0/2} = \frac{\sum_{n=0}^{N-1} s^2(n)}{\sigma^2} = N * SNR$$

While we know:

$$SNR = \frac{A^2/2}{\sigma^2}$$

Where N = 16, $\sigma^2 = 1$

According to the equations above, we have:

$$A = \sqrt{\frac{4E\sigma^2}{N * N_0}}$$

Then we can calculate:

a) When $\frac{2E}{N_0} = 4,\ A = \frac{\sqrt{2}}{2}$, SNR = 1/4

b) When $\frac{2E}{N_0} = 9,\ A = \frac{3\sqrt{2}}{4}$, SNR = 9/16

2. For the second problem, there are 2 parts:

a) For SKE problem:

i. Theoretically, we use PDF of Gaussian distribution to plot the ideal or theoretical distribution of p(T(x)|H0) and p(T(x)|H1). Notice that T(x) is sufficient data. In matlab, there is a function called 'normpdf' to generate this kind of Gaussian distribution.

For H0, mean value is 0 and variance is $\frac{\sigma^2 N}{2}$. For H1, mean value is NA/2 and variance is the same as H0.

Then we got the black solid line curve in first two plots of figure (1) for theoretically PDF of SKE.

ii. For Monte Carlo simulation: We first generate sufficient data (M=1000000). Then we calculate the probability by:

1. Finding the length of period which data is in a certain range.
2. Calculating length/M, this is the probability we want
3. Finding the length of period which data is above a threshold.
4. Calculating length/M, this is the Q function we want.

Note that the probability is the histogram in figure (1) and the Q function is the red/blue curve in third and fourth plots in figure (1)

iii. For pd and pf:

1. Theoretically, we use the Q function provided by the book to calculate value.
2. For M-C simulation, we calculate pd and pf by finding the data period above threshold of H0 and H1, then just like before, calculating length (period)/M.

3. We use original ‘plot’ function and ‘probpaper’ function provided by the book to plot pd and pf both on linear axes and on normal probability paper.
4. Note: In figure 2, red solid line is the theoretically calculated ROC curve and cyan dash line is the M-C simulation ROC curve. They are nearly the same.

b) For SKEP problem:

i. The method to solve this problem is nearly the same as before. We do not get into details here.

ii. The difference is the function to calculate the probability. We cannot use ‘normpdf’ anymore. Here we choose to type and calculate the pdf by ourselves. We need to calculate the non-central parameter λ of PDF and the ν degrees of freedom is 2.

iii. Please see Appendix code for details.

- Results(including plots):

1. In figure 1 and 3, there are plots of pdf theoretically and M-C simulation. According to my result, they are nearly the same.
2. For SKE an SKEP problem, the PDF is in different shape because they are different distributions.

   In figure 1 and 3, the blue histogram and blue curve, which is the pdf and Q function curve for H1 condition, shift to right a bit when SNR increase. The mean is related to SNR. And Q function is the integration from a certain value to infinity. So when SNR increase, the mean of the distribution shift to right and Q function also increase.
3. Figure 2 and 4 are ROC curves where x axis is PD and y axis is PFA. In each of the figure, left plot is on normal probability paper and right plot is on linear axis. Red solid line is the theoretically calculated ROC curve and cyan dash line is the M-C simulation ROC curve. The below one is for H0 and the above one is for H1.
4. I tested M = 10000 and M = 1000000, it appeared that when M becomes larger, the M-C simulation is more accurate.
5. The higher the SNR is the better the processor performance is.

Plots:

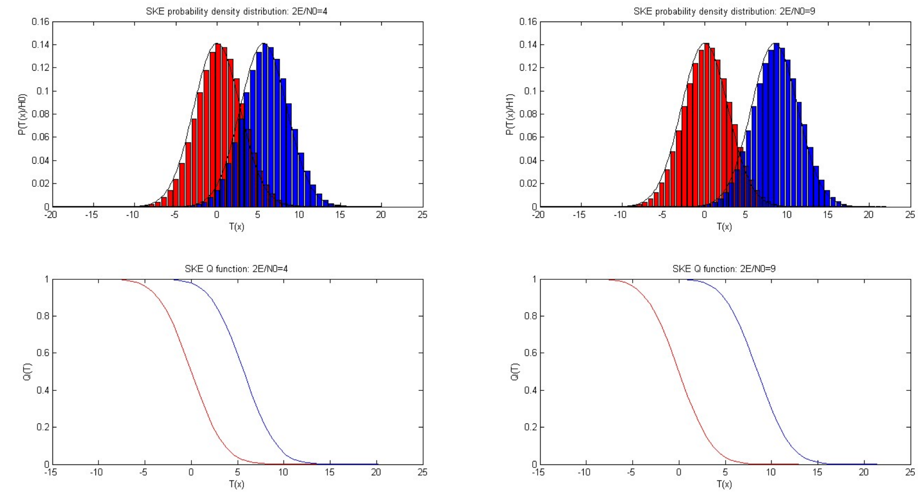


Figure 1 SKE probability density distribution and Q function

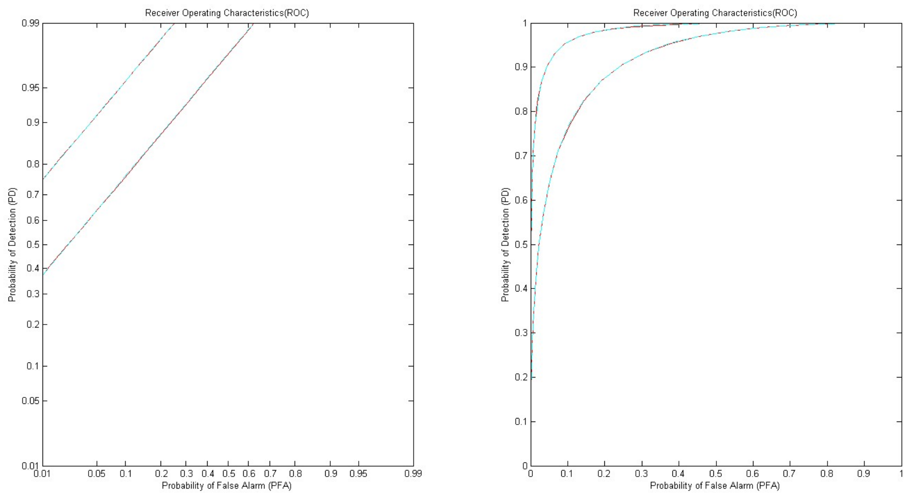


Figure 2 SKE ROC curve on different axes

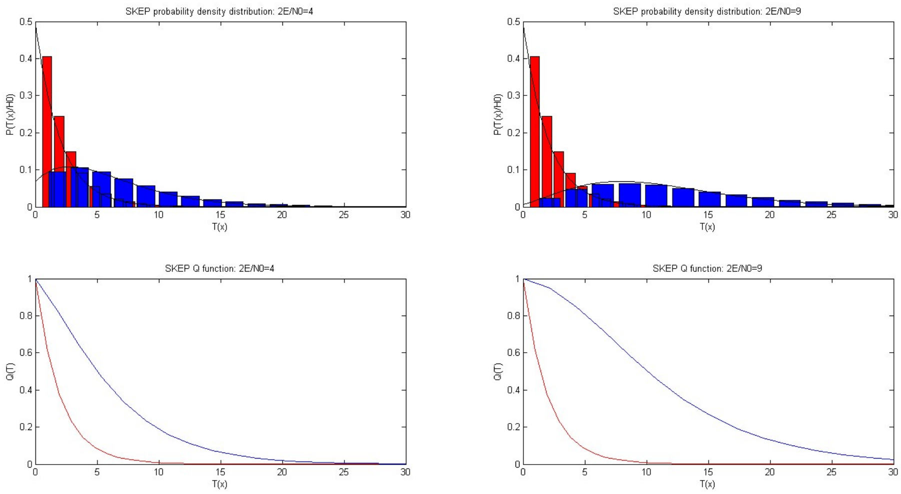


Figure 3 SKEP probability density distribution and Q function

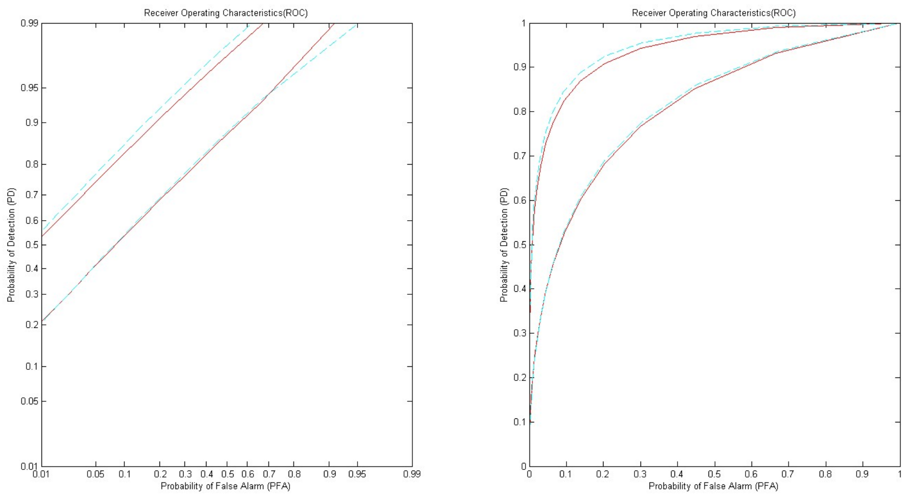


Figure 4 SKEP ROC curve on different axes

- Appendix:

  Hw2.m:

```
% Define constance

%

% 2E/N0 = 4

N = 16;

SNR1 = 1/4;

A1 = (1/2)^(1/2);

meanh01 = 0;

meanh11 = A1 * N/2;

varh01 = N/2;

varh11 = N/2;


% 2E/N0 = 9

SNR2 = 9/16;

A2 = (9/8)^(1/2);

meanh02 = 0;

meanh12 = A2 * N/2;

varh02 = N/2;

varh12 = N/2;

%

n = 0:1:15;
```

```
fc = 1/16;

M = 1000000;

sn1 = A1*sin(2*pi*fc*n);

sn2 = A2*sin(2*pi*fc*n);


%% ske problem


% case 1: 2E/N0 = 4

% generate data

T0 = zeros(1, M);

T1 = zeros(1, M);

for i = 1: M

    nn = randn(1, 16);

    xn0 = nn;

    xn1 = sn1+nn;

    t0 = xn0.*sn1/A1;

    t1 = xn1.*sn1/A1;

    T0(i) = sum(t0);

    T1(i) = sum(t1);

end


% calculate parameters for plot
```

```
ngam=40;
gmin0=min(T0);
gmax0=max(T0);
gmin1=min(T1);
gmax1=max(T1);
step0=(gmax0-gmin0)/ngam;
step1=(gmax1-gmin1)/ngam;
gama0= gmin0:step0:gmax0;
gama1= gmin1:step1:gmax1;
Gama0=[0,gama0];
Gama1=[0,gama1];

p0=zeros(1,length(gama0));
p1=zeros(1,length(gama1));

% calculate p
for i=1:length(Gama0)-1
    clear temp0;
    temp0=find(T0>Gama0(i)&T0<Gama0(i+1));
    p0(i)=length(temp0)/M;
end
for i=1:length(Gama1)-1
```

```
    clear temp1;

    temp1=find(T1>Gama1(i)&T1<Gama1(i+1));

    p1(i)=length(temp1)/M;

end


% theoratically p

x=-20:0.1:20;

y0=normpdf(x,meanh01,(varh01)^(1/2));

y1=normpdf(x,meanh11,(varh11)^(1/2));


% plot

figure(1)

subplot(2,2,1)

bar(gama0,p0/step0,'r')

hold on

figure(1)

subplot(2,2,1)

bar(gama1,p1/step1,'b')

figure(1)

subplot(2,2,1)

plot(x,y0,'k')

figure(1)
```

```
subplot(2,2,1)

plot(x,y1,'k')

xlabel('T(x)')

ylabel('P(T(x)/H0)');

title('SKE probability density distribution: 2E/N0=4');


for i=1:length(gama0)

    clear temp0;

    temp0=find(T0>gama0(i));

    p0(i)=length(temp0)/M;

end

for i=1:length(gama1)

    clear temp1;

    temp1=find(T1>gama1(i));

    p1(i)=length(temp1)/M;

end

figure(1)

subplot(2,2,3)

plot(gama0,p0,'r')

hold on

figure(1)

subplot(2,2,3)
```

```
plot(gama1,p1,'b')

xlabel('T(x)')

ylabel('Q(T)');

title('SKE Q function: 2E/N0=4');


% case 2: 2E/N0 = 9

T2=zeros(1,M);

T3=zeros(1,M);

for i=1:M

  nn1=randn(1,16);

  xn2=nn1;

  xn3=sn2+nn1;

  t2=xn2.*sn2/A2;

  t3=xn3.*sn2/A2;

  T2(i)=sum(t2);

  T3(i)=sum(t3);

end

gmin2=min(T2);

gmax2=max(T2);

gmin3=min(T3);

gmax3=max(T3);

step2=(gmax2-gmin2)/ngam;
```

```
step3=(gmax3-gmin3)/ngam;

gama2=gmin2:step2:gmax2;

gama3=gmin3:step3:gmax3;


p2=zeros(1,length(gama2));

p3=zeros(1,length(gama3));


Gama2=[0,gama2];

Gama3=[0,gama3];

for i=1:length(Gama2)-1

    clear temp2;

    temp2=find(T2>Gama2(i)&T2<Gama2(i+1));

    p2(i)=length(temp2)/M;

end

for i=1:length(Gama3)-1

    clear temp3;

    temp3=find(T3>Gama3(i)&T3<Gama3(i+1));

    p3(i)=length(temp3)/M;

end

x=-20:0.1:20;

y2=normpdf(x,meanh02,(varh02)^(1/2));

y3=normpdf(x,meanh12,(varh12)^(1/2));
```

```
figure(1)

subplot(2,2,2)

bar(gama2,p2/step2,'r')

hold on

figure(1)

subplot(2,2,2)

bar(gama3,p3/step3,'b')

figure(1)

subplot(2,2,2)

plot(x,y2,'k')

figure(1)

subplot(2,2,2)

plot(x,y3,'k')

xlabel('T(x)')

ylabel('P(T(x)/H1)');

title('SKE probability density distribution: 2E/N0=9');


for i=1:length(gama2)

    clear temp2;

    temp2=find(T2>gama2(i));

    p2(i)=length(temp2)/M;

end
```

```
for i=1:length(gama3)

        clear temp3;

        temp3=find(T3>gama3(i));

        p3(i)=length(temp3)/M;

end

figure(1)

subplot(2,2,4)

plot(gama2,p2,'r')

hold on

figure(1)

subplot(2,2,4)

plot(gama3,p3,'b')

xlabel('T(x)')

ylabel('Q(T)');

title('SKE Q function: 2E/N0=9');


% calculate pd and pf

b1max = gmax0;

b1min = gmin1;

b1step = (b1max - b1min)/ngam;

b1 = b1min:b1step:b1max;

pf1 = zeros(1, length(b1));
```

```
pd1 = zeros(1, length(b1));

p0 = Q(b1/(varh01)^(1/2));

p1 = Q((b1-meanh11)/(varh11)^(1/2));


for i=1:length(b1)

    clear temp0;

    clear temp1;

    temp0=find(T0>b1(i));

    pf1(i)=length(temp0)/M;

    temp1=find(T1>b1(i));

    pd1(i)=length(temp1)/M;

end


b2max=gmax2;

b2min=gmin3;

b2step=(b2max-b2min)/ngam;

b2=b2min:b2step:b2max;

pf2=zeros(1,length(b2));

pd2=zeros(1,length(b2));

p2=Q(b2/(varh02)^(1/2));

p3=Q((b2-meanh12)/(varh12)^(1/2));

for i=1:length(b2)
```

```
    clear temp2;
    clear temp3;
    temp2=find(T2>b2(i));
    pf2(i)=length(temp2)/M;
    temp3=find(T3>b2(i));
    pd2(i)=length(temp3)/M;
end

figure(2)
subplot(1,2,1)
probpaper(pf1, pd1, 'r')
hold on
figure(2)
subplot(1,2,1)
probpaper(pf2, pd2, 'r')
figure(2)
subplot(1,2,1)
probpaper(p0,p1, 'c--')
figure(2)
subplot(1,2,1)
probpaper(p2,p3, 'c--')
```

```
figure(2)

subplot(1,2,2)

plot(pf1,pd1,'r')

hold on

figure(2)

subplot(1,2,2)

plot(pf2,pd2,'r')

figure(2)

subplot(1,2,2)

plot(p0,p1,'c--')

figure(2)

subplot(1,2,2)

plot(p2,p3,'c--')

 xlabel('Probability of False Alarm (PFA)')

  ylabel('Probability of Detection (PD)');

  title('Receiver Operating Characteristics(ROC)');



%% skep problem:

v = 2;

lambda1 = A1^2*16/2;

lambda2 = A2^2*16/2;
```

```
T0=zeros(1,M);

T1=zeros(1,M);

m=zeros(1,M*16);


% case 1:

% generate data

for i=1:M

    nn=randn(1,16);

    phi=(rand(1,16)*2-1)*pi;

    sn=A1*sin(2*pi*fc*n+phi);

    cn=A1*cos(2*pi*fc*n+phi);

    xn0=nn;

    xn1=sn+nn;

    a0=xn0.*cn/A1;

    b0=xn0.*sn/A1;

    a1=xn1.*cn/A1;

    b1=xn1.*sn/A1;

    A0=sum(a0);

    B0=sum(b0);

    A1_p=sum(a1);

    B1_p=sum(b1);

    T0(i)=(1/N)*(A0^2+B0^2)*2;
```

```
    T1(i)=(1/N)*(A1_p^2+B1_p^2)*2;
end

ngam=40;
gmin0=min(T0);
gmax0=max(T0);
gmin1=min(T1);
gmax1=max(T1);
step0=(gmax0-gmin0)/ngam;
step1=(gmax1-gmin1)/ngam;
gama0= gmin0:step0:gmax0;
gama1= gmin1:step1:gmax1;
Gama0=[0,gama0];
Gama1=[0,gama1];

p0=zeros(1,length(gama0));
p1=zeros(1,length(gama1));

% calculate p
for i=1:length(Gama0)-1
   clear temp0;
   temp0=find(T0>Gama0(i)&T0<Gama0(i+1));
```

```
        p0(i)=length(temp0)/M;
end
for i=1:length(Gama1)-1
        clear temp1;
        temp1=find(T1>Gama1(i)&T1<Gama1(i+1));
        p1(i)=length(temp1)/M;
end

x=0:0.1:30;
y0=(1/(2^(v/2)*gamma(v/2)))*exp(-1/2*x);
y1=(1/2)*exp(-(1/2)*(x+lambda1)).*besseli(0,(lambda1*x).^(1/2));

figure(3)
subplot(2,2,1)
axis([0,30,0,0.5]);
bar(gama0,p0/step0,'r')
hold on
figure(3)
subplot(2,2,1)
axis([0,30,0,0.5]);
bar(gama1,p1/step1,'b')
figure(3)
```

```
subplot(2,2,1)

axis([0,30,0,0.5]);

plot(x,y0,'k')

figure(3)

subplot(2,2,1)

axis([0,30,0,0.5]);

plot(x,y1,'k')

xlabel('T(x)')

ylabel('P(T(x)/H0)');

title('SKEP probability density distribution: 2E/N0=4');


for i=1:length(gama0)

  clear temp0;

  temp0=find(T0>gama0(i));

  p0(i)=length(temp0)/M;

end

for i=1:length(gama1)

  clear temp1;

  temp1=find(T1>gama1(i));

  p1(i)=length(temp1)/M;

end

figure(3)
```

```
subplot(2,2,3)

axis([0,30,0,1]);

plot(gama0,p0,'r')

hold on

figure(3)

subplot(2,2,3)

axis([0,30,0,1])

plot(gama1,p1,'b')

xlabel('T(x)')

ylabel('Q(T)');

title('SKEP Q function: 2E/N0=4');


% case 2:

T2=zeros(1,M);

T3=zeros(1,M);


for i=1:M

  nn1=randn(1,16);

  phi1=(rand(1,16)*2-1)*pi;

  sn1=A2*sin(2*pi*(1/16)*n+phi1);

  xn2=nn1;

  xn3=sn1+nn1;
```

```
  a2=xn2.*cos(2*pi*(1/16)*n+phi1);
  b2=xn2.*sin(2*pi*(1/16)*n+phi1);
  a3=xn3.*cos(2*pi*(1/16)*n+phi1);
  b3=xn3.*sin(2*pi*(1/16)*n+phi1);
  A2_p=sum(a2);
  B2_p=sum(b2);
  A3=sum(a3);
  B3=sum(b3);
  T2(i)=T0(i);
  T3(i)=(1/N)*(A3^2+B3^2)*2;
end

gmin2=min(T2);
gmax2=max(T2);
gmin3=min(T3);
gmax3=max(T3);
step2=(gmax2-gmin2)/ngam;
step3=(gmax3-gmin3)/ngam;
gama2=gmin2:step2:gmax2;
gama3=gmin3:step3:gmax3;

p2=zeros(1,length(gama2));
```

```
p3=zeros(1,length(gama3));


Gama2=[0,gama2];
Gama3=[0,gama3];
for i=1:length(Gama2)-1
    clear temp2;
    temp2=find(T2>Gama2(i)&T2<Gama2(i+1));
    p2(i)=length(temp2)/M;
end
for i=1:length(Gama3)-1
    clear temp3;
    temp3=find(T3>Gama3(i)&T3<Gama3(i+1));
    p3(i)=length(temp3)/M;
end
x=0:0.1:30;
y2=(1/(2^(v/2)*gamma(v/2)))*exp(-1/2*x);
y3=(1/2)*exp(-(1/2)*(x+lambda2)).*besseli(0,(lambda2*x).^(1/2));


figure(3)
subplot(2,2,2)
axis([0,30,0,0.5]);
bar(gama2,p2/step2,'r')
```

```
hold on

figure(3)

subplot(2,2,2)

axis([0,30,0,0.5]);

bar(gama3,p3/step3,'b')

figure(3)

subplot(2,2,2)

axis([0,30,0,0.5]);

plot(x,y2,'k')

figure(3)

subplot(2,2,2)

axis([0,30,0,0.5]);

plot(x,y3,'k')

xlabel('T(x)')

ylabel('P(T(x)/H0)');

title('SKEP probability density distribution: 2E/N0=9');



for i=1:length(gama2)

  clear temp2;

  temp2=find(T2>gama2(i));

  p2(i)=length(temp2)/M;
```

```
end
for i=1:length(gama3)
  clear temp3;
  temp3=find(T3>gama3(i));
  p3(i)=length(temp3)/M;
end
figure(3)
subplot(2,2,4)
axis([0,30,0,1]);
plot(gama2,p2,'r')
hold on
figure(3)
subplot(2,2,4)
axis([0,30,0,1])
plot(gama3,p3,'b')
xlabel('T(x)')
ylabel('Q(T)');
title('SKEP Q function: 2E/N0=9');


I1max=gmax0;
I1min=gmin1;
```

```
l1step=(l1max-l1min)/ngam;

l1=l1min:l1step:l1max;

pf1=zeros(1,length(l1));

pd1=zeros(1,length(l1));

p0=Qchipr2(2,0,l1,0.0001);

p1=Qchipr2(2,lambda1,l1,0.0001);

for i=1:length(l1)

  clear temp0;

  clear temp1;

  temp0=find(T0>l1(i));

  pf1(i)=length(temp0)/M;

  temp1=find(T1>l1(i));

  pd1(i)=length(temp1)/M;

end

l2max=gmax2;

l2min=gmin3;

l2_step=(l2max-l2min)/ngam;

l2=l2min:l2_step:l2max;

pf2=zeros(1,length(l2));

pd2=zeros(1,length(l2));



p2=Qchipr2(2,0,l2,0.0001);
```

```
p3=Qchipr2(2,lambda2,l2,0.0001);
for i=1:length(l2)
  clear temp2;
  clear temp3;
  temp2=find(T2>l2(i));
  pf2(i)=length(temp2)/M;
  temp3=find(T3>l2(i));
  pd2(i)=length(temp3)/M;
end
figure(4)
subplot(1,2,1)
probpaper(pf1,pd1, 'r')
hold on
figure(4)
subplot(1,2,1)
probpaper(pf2,pd2, 'r')
figure(4)
subplot(1,2,1)
probpaper(p0,p1, 'c--')
figure(4)
subplot(1,2,1)
probpaper(p2,p3, 'c--')
```

```
figure(4)

subplot(1,2,2)

plot(pf1,pd1, 'r')

hold on

figure(4)

subplot(1,2,2)

plot(pf2,pd2,'r')

figure(4)

subplot(1,2,2)

plot(p0,p1, 'c--')

figure(4)

subplot(1,2,2)

plot(p2,p3, 'c--')

 xlabel('Probability of False Alarm (PFA)')

  ylabel('Probability of Detection (PD)');

  title('Receiver Operating Characteristics(ROC)');
```