

МОДЕЛИРОВАНИЕ СИГНАЛОВ И ПОЛЕЙ В СРЕДЕ Matlab

Лабораторная работа № 2

по дисциплине

«Математическое моделирование радиотехнических устройств и систем»

Цель работы: Изучение методов представления, характеристик и алгоритмов формирования полей сигналов и помех, присущих радиотехническим системам и устройствам.

1. Формирование объектов на изображениях

Рассмотрим несколько программ, позволяющих получить сигнальные изображения для полезных объектов, которые могут использоваться для моделирования сигналов радиотехнических систем наблюдения.

Первая программа `func1_rect` формирует объект прямоугольной формы с заданными размерами и расположением в пределах изображения. Аргументы функции включают размеры изображения (m,n) , размеры сигнального прямоугольника (M,N) , и значения (x,y) , задающие смещение центра прямоугольника относительно центра изображения:

```
function[S]=func1_rect(m,n,M,N,x,y)
%rectangular signal window S =(M,N) in
%rectangular image window (m,n)
%m,n-vertical and horizontal image sizes
%M,N-vertical and horizontal window sizes
%x,y - signal window shift wrt the center
%of the image
N1=floor(N/2)-1; M1=floor(M/2)-1;
m1=ceil(m/2)+1; n1=ceil(n/2)+1;
for i=1:m
    for j=1:n
        S(i,j)=0;
    end
end;
for i=m1-M1-1+y:m1+M1+y+1
    for j=n1-N1-1+x:n1+N1+x+1
        S(i,j)=1;
    end
end
```

Применение этой функции иллюстрируется следующей программой, а полученное изображение приведено на рис. 1.1 (слева):

```
clear %removes all variables from the workspace
close all % deletes all figures
STEP=0
m=256 %vertical image size
```

```

n=256 %horizontal image size
M=25  %vertical window size
N=49  %horizontal window size
x=0   %horizontal window shift
y=0   %vertical window shift
S=func1_rect(m,n,M,N,x,y);
indS=find(S); %array of nonzero elements
NS1=size(indS) %number of cells
figure(10), imshow(S,[]), title('Signal');

```

Вторая функция, которая будет использоваться далее, формирует изображение *s0*, разделенное на две части горизонтальной границей, так что верхняя часть будет черной, а нижняя – белой. Входной параметр *y* задает смещение границы, при *y=0* граница делит изображение пополам. Выходной параметр *m* дает абсолютный сдвиг границы относительно верха изображения. Функция *func1_one_edge* формирует еще три изображения *s90*, *s180* и *s270*, получаемые последовательным поворотом изображения *s0* на 90° против часовой стрелки:

```

function[S0,S90,S180,S270,M]=func1_one_edge(m,n,y)
%one edge in rectangular window (m,n)
%m-vertical, n-horizontal
%y - shift wrt the center of the window
m1=ceil(m/2)+1;
for i=1:m
    for j=1:n
        S(i,j)=0;
    end
end;
if y>m1-1, y=m1-2; end
M=m1+y;
for i=M:m
    for j=1:n
        S(i,j)=1;
    end
end;
S0=S; S90=rot90(S); S180=rot90(S90); S270=rot90(S180);

```

Применение данной функции в программе *test_one_edge* позволяет получить изображения, приведенные на рис. 1.1:

```

%test_one_edge
close all
STEP=0
m=256; n=256; y=0 %vertical shift
[S0,S90,S180,S270,M]=func1_one_edge(m,n,y);
%counterclockwise rotation
indS0=find(S0); NS0=size(indS0)
M
figure(10), imshow(S0,[]), title('HorizontalEdge');
figure(11), imshow(S90,[]), title('VerticalEdge');

```

```
figure(12), imshow(S180,[]), title('HorizontalUpsideEdge');
figure(13), imshow(S270,[]), title('VerticalUpsideEdge');
imwrite(S0, 'S0.jpg');
imwrite(S90, 'S90.jpg');
imwrite(S180, 'S180.jpg');
imwrite(S270, 'S270.jpg');
```

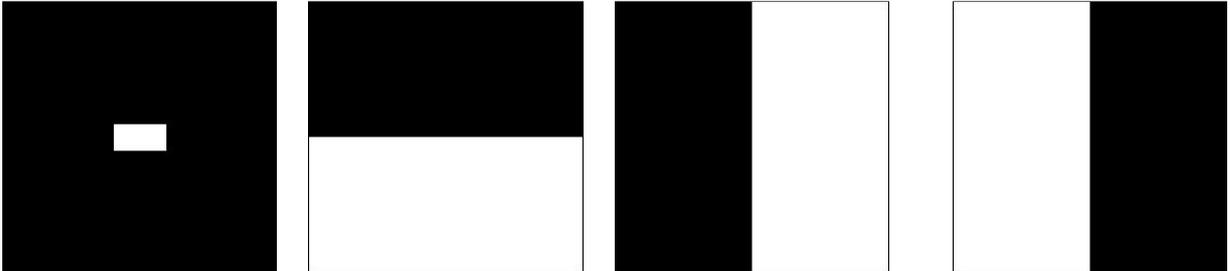


Рис. 1.1 Тестовые изображения прямоугольного объекта (слева), и различные положения границы на изображении S_0 , S_{90} и S_{180}

2. Моделирование шумов с различными распределениями вероятности

Рассмотрим программы моделирования шумовых помех, т. е. помех с независимыми значениями. Для исследования помехоустойчивости различных алгоритмов обработки применяются шумы с модельными распределениями вероятностей [1, 2].

Часто используются однопараметрические модели для плотностей вероятностей (например, экспоненциальная и релеевская плотности вероятностей), и двухпараметрические семейства плотностей (гауссовское, Вейбулла, гамма, логнормальное, равномерное и др.). Малое число входных параметров обеспечивает легкость управления моделью, хотя модели с малым числом параметров не всегда точно описывают реальные процессы в радиотехнических системах.

Экспоненциальное распределение $E(\beta)$ с параметром масштаба β играет важную роль при описании случайных процессов. Оно сильно отличается от гауссовского и входит в такие известные семейства распределений, как распределение Вейбулла и гамма-распределение. Таким образом, оно является некоторой связкой между этими двумя семействами.

Моделирование шума с экспоненциальным распределением дается функцией `func2_sum_exp`:

```
function[e,E]=func2_sum_exp(m,n,b,N)
%sum of N exp noises, scale b
e=zeros([m,n],'double');
for k=1:N
r1=rand(m,n); e1=-log(1-r1)*b; e=e1+e;
end;
me=max(max(e)); E=e/me;
```

Входной параметр b представляет масштаб распределения β . При $n=1$ функция генерирует матрицу независимых случайных чисел с экспоненциальной плотностью вероятностей. Выходная матрица E представляет случайные числа, нормированные к максимальному значению. При $n>1$ функция дает в каждой точке изображения сумму N независимых экспоненциально распределенных чисел.

Применение этой функции иллюстрируется программой:

```
%main_sum_exp_noise
clear all
close all
STEP=0
m=256; n=256; [m,n]
N=1 %number of noises
STEP=1 %ExpNoise
b=30 %scale parameter
[e,E]=func2_sum_exp(m,n,b,N);
figure(11), imshow(E,[]), title('SumExpNoise');
figure(110), imhist(E), title('SumExpHistogram');
t=input('STOP_1_ExpNoise','s');
STEP=2 %ExponentialPDF
me=max(max(e)) %max value
XM=floor(me); DXM=XM/200;%plot distribution
x=1:DXM:XM; p1 = pdf('exp',x,b);
figure(1000), plot(x,p1), title('Theoretical PDF');
t=input('STOP_2_ExpPDF','s');
STEP=3 %ParametersCalc1
[mean,var,sigma,gamma1,mn]=func3_mean_var_calc(e);
[mean,var,sigma,gamma1] % parameters output
t=input('STOP_3_ExpNoiseParameters_1','s');
STEP=4 %ParametersCalc1
[skew,gamma3,kurt,gamma4,mn]=func3_skew_kurt_calc(e);
[skew,gamma3,kurt,gamma4] % parameters output
t=input('STOP_4_ExpNoiseParameters_2','s');
```

Для исследования характеристик полученного шумового изображения вводятся еще две функции. Функция `func3_mean_var_calc` вычисляет выборочные среднее `mean` и дисперсию `var` по всему изображению:

```
function[mean,var,sigma,gamma1,mn]=func3_mean_var_calc(Y)
[m,n]=size(Y); mn=m*n;
mean=sum(sum(Y)); mean=mean/mn; Z=Y-mean;
var=sum(sum(Z.^2)); var=var/(mn-1); sigma=sqrt(var);
gamma1=mean/sigma;
```

Кроме того, функция вычисляет выборочное среднеквадратическое значение (стандартное отклонение) `sigma` и выборочный коэффициент `gamma1`, который является обратной величиной к коэффициенту вариации (отношение математического ожидания к СКО). Заметим, что вычисляемые

выборочные значения являются лишь оценками для параметров плотности, а не самими параметрами. Они случайны, и сами имеют плотности вероятностей, которые часто трудно точно установить. В частности, выходной параметр **mean** есть среднее арифметическое, он является случайной оценкой для неслучайной величины математического ожидания. В случае экспоненциальной плотности математическое ожидание совпадает с параметром масштаба **b**, и значение **mean** является достаточно хорошей несмещенной оценкой для него.

Для моментов и кумулянтов высших порядков моментные оценки не всегда обладают хорошими свойствами, и это следует учитывать при их практическом использовании. Выходной параметр **mn** дает общее число выборок на изображении.

Функция **func3_skew_kurt_calc** вычисляет выборочные значения для асимметрии **skew** и эксцесса **kurt**, а также выборочные значения кумулянтных коэффициентов третьего и четвертого порядков **gamma3** и **gamma4**:

```
function[skew,gamma3,kurt,gamma4,mn]=func3_skew_kurt_calc(Y)
[m,n]=size(Y); mn=m*n;
mean=sum(sum(Y)); mean=mean/mn; Z=Y-mean;
var=sum(sum(Z.^2)); var=var/(mn-1);
skew=sum(sum(Z.^3)); skew=skew/mn;
kurt=sum(sum(Z.^4)); kurt=kurt/mn; sigma=sqrt(var);
gamma3=skew/sigma.^3; gamma4=kurt/var.^2-3;
```

На рис. 2.1 показаны гистограммы, полученные для экспоненциальной плотности (слева), для суммы пяти независимых случайных чисел (в центре), и для десяти случайных чисел (справа). Очевидно, что плотности имеют тенденцию к увеличению симметрии слева направо, что отражает эффект «нормализации» плотностей при суммировании случайных чисел в соответствии с центральной предельной теоремой.

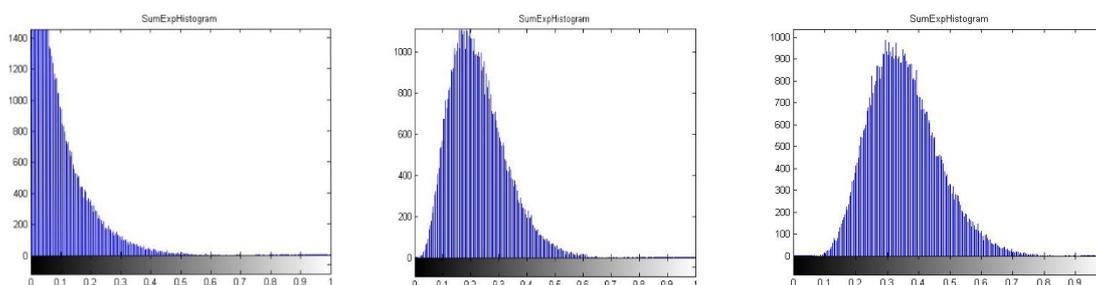


Рис. 2.1. Гистограммы для экспоненциального распределения (слева), и для сумм пяти и десяти экспоненциально распределенных чисел

Для генерирования шумов с другими модельными распределениями и исследования их характеристик в программе **main_sum_exp_noise** достаточно заменить два фрагмента с заголовками **STEP=1** и **STEP=2**.

В случае релеевского шума $R(\beta)$ с параметром масштаба β эти фрагменты будут такими:

```

STEP=1 %RayleighNoise
b=30;
[r,R]=func2_sum_Rayleigh(m,n,b,N);
figure(11), imshow(r,[]), title('SumRayleighNoise');
figure(110), imhist(R), title('SumRayleighHistogram');
t=input('STOP_1_RayleighNoise','s');
STEP=2 %RayleighPDF
mr=max(max(r)) %max value
XM=floor(mr); DXM=XM/200;%plot distribution
x=1:DXM:XM; p1 = pdf('rayl',x,b);
figure(1000), plot(x,p1), title('Theoretical PDF');
t=input('STOP_2_RayleighPDF','s');

```

Функция `func2_sum_Rayleigh` содержит параметр масштаба b , равный β для релеевского распределения $R(\beta)$:

```

function[r,R]=func2_sum_Rayleigh(m,n,b,N)
%sum of N Rayleigh noises, scale b
r=zeros([m,n], 'double');
for k=1:N
u1=rand(m,n); e1=-log(1-u1); r1=b*sqrt(e1); r=r1+r;
end;
mr=max(max(r)); R=r/mr;

```

Для гауссовского шума $N(\mu, \sigma)$ с математическим ожиданием (МО) μ и среднеквадратическим отклонением (СКО) σ фрагмент программы выглядит так:

```

STEP=1 %NormalNoise
a=30; b=10;
[g,G]=func2_sum_normal(m,n,a,b,N);
figure(11), imshow(G,[]), title('SumNormalNoise');
figure(110), imhist(G), title('SumNormalHistogram');
t=input('STOP_1_NormalNoise','s');
STEP=2 %NormalNoiseCalc1
mg=max(max(g)) %max value
XM=floor(mg); DXM=XM/200;%plot distribution
x=1:DXM:XM; p1 = pdf('norm',x,a,b);
figure(1000), plot(x,p1), title('Theoretical PDF');

```

Функция `func2_sum_normal` имеет в качестве входных параметров МО и СКО значения (a, b) , а N есть число суммируемых случайных величин при генерировании случайного значения в каждой точке поля:

```

function[g,G]=func2_sum_normal(m,n,a,b,N)
%sum of N Gauss noises, scale b
g=zeros([m,n], 'double');

```

```

for k=1:N
    g1=randn(m,n); g1=g1*b+a; g=g1+g;
end;
mg=max(max(g)); G=g/mg;

```

Логнормальное распределение $LN(x_{med}, kv)$ генерируется с помощью следующих фрагментов:

```

STEP=1 %LognormalNoise
med=30; kv=0.5; %med>0 kv>0
[lg,LG,ro,mul,sigmal,E,V]=func2_sum_Lognormal(m,n,med,kv,N);
figure(11), imshow(LG,[]), title('SumLognormalNoise');
figure(110), imhist(LG), title('SumLognormalHistogram');
ro %ro=E/med
%ro_check=E/med
mul %N(mul,sigmal)-generic normal distribution
sigmal % sigmal=sqrt(2*ln(ro))
t=input('STOP_1_LognormalNoise','s');
STEP=2 %LognormalPDF
mlg=max(max(lg)) %max value
XM=floor(mlg); DXM=XM/200;%plot distribution
x=1:DXM:XM; p1 = pdf('logn',x,mul,sigmal.^2);
figure(1000), plot(x,p1), title('Theoretical PDF');
t=input('STOP_2_LognormalPDF','s');

```

Функция `func2_sum_Lognormal` имеет в качестве входных параметров медиану `med` и коэффициент вариации `kv` плотности вероятности:

```

function[lg,LG,ro,mul,sigmal,E,V]
=func2_sum_Lognormal(m,n,med,kv,N)
%sum of N Lognormal noises, med-median, kv- coeff. var
lg=zeros([m,n],'double');
ro=sqrt(kv.^2+1); mul=log(med); E=ro*med;
V=E.^2*kv.^2; sigmal=sqrt(2*log(ro));
for k=1:N
    g1=randn(m,n); g1=g1*sigmal+mul;
    lg1=exp(g1); lg=lg1+lg;
end;
mlg=max(max(lg));
LG=lg/mlg;

```

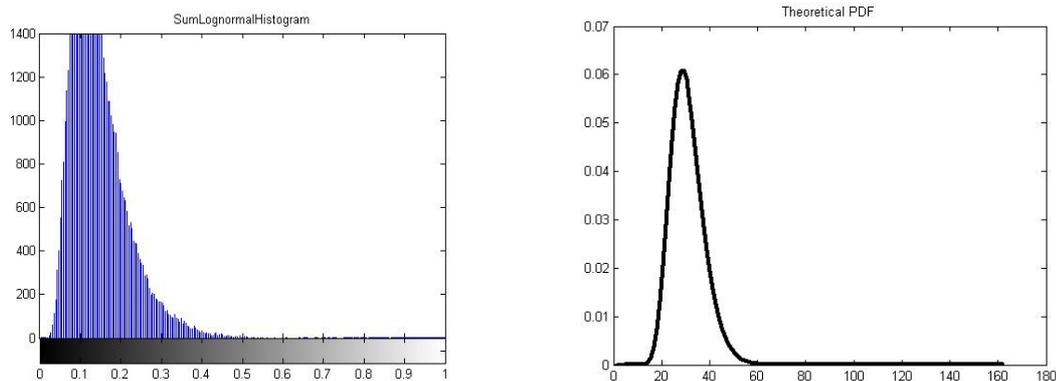


Рис. 2.2. Гистограмма логнормального распределения (слева) и теоретическая плотность вероятности (справа)

На рис. 2.2 представлены гистограмма логнормального распределения, полученная для $med=30$, $kv=0,5$ (слева), и теоретическая плотность вероятности (справа).

СПИСОК ЛИТЕРАТУРЫ

1. Волков, В. Ю. Обнаружение и различение сигналов в радиотехнических задачах: учеб. пособие / В. Ю. Волков ; СПбГУТ. – СПб., 2000.
2. Волков, В. Ю. Адаптивные, инвариантные и робастные методы обнаружения и различения сигналов: учеб. пособие / В. Ю. Волков ; СПбГУТ. – СПб., 2005.
3. Волков, В. Ю. Адаптивные, инвариантные и робастные методы обнаружения и различения сигналов : учеб. пособие : часть 2 / В. Ю. Волков ; СПбГУТ. – СПб., 2008.
4. Волков, В. Ю. Адаптивные, инвариантные и робастные методы обнаружения и различения сигналов : учеб. пособие : часть 3 / В. Ю. Волков ; СПбГУТ. – СПб., 2012