

|  |  |  |  |
| --- | --- | --- | --- |
| **作 者:** | **芮涛** | **学 号：** | **913104210321** |
| **学 院:** | **电子工程与光电技术学院** | | |
| **专 业:** | **电子信息工程** | | |
| **课 程:** | **信号检测与估计** | | |
| **题 目:** | **伪相位编码连续波雷达信号处理** | | |
| **指导老师:** | **顾 红** | | |

（末尾1，8学号做（4人））仿真伪随机相位编码连续波雷达的信号处理。设码频为各学生学号末两位数，单位为MHz，伪码周期内码长为127，雷达载频为10GHz，输入噪声为高斯白噪声。目标模拟分单目标和双目标两种情况，目标回波输入信噪比可变（-35dB～10dB），目标速度可变（0～1000m/s），目标幅度可变（1～100），目标距离可变（0～10000m），相干积累总时宽不大于10ms。单目标时，给出回波视频表达式；脉压和FFT 后的表达式；仿真m序列的双值电平循环自相关函数，给出脉压后和FFT 后的输出图形；通过仿真说明各级处理的增益，与各级时宽和带宽的关系；仿真说明脉压时多卜勒敏感现象和多卜勒容限及其性能损失（脉压主旁比与多卜勒的曲线）。双目标时，仿真出大目标旁瓣盖掩盖小目标的情况；仿真出距离分辨和速度分辨的情况。

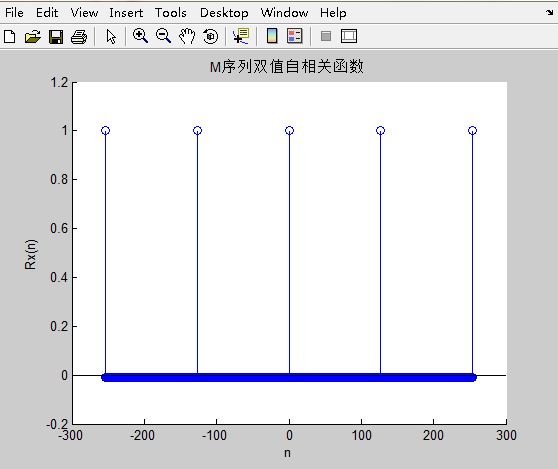
1.回波视频表达式：

其中为目标幅度，为延迟后的M序列，为载频，为多普勒频率

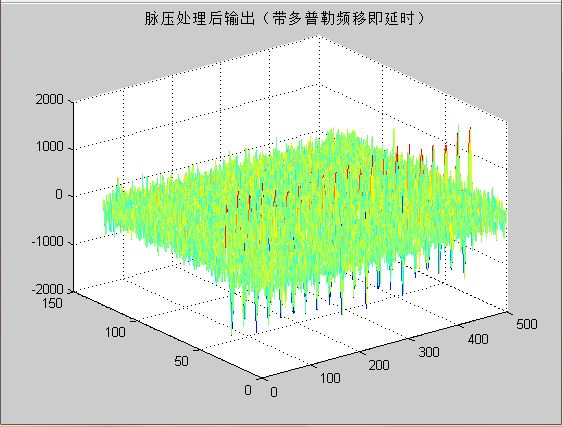
脉压表达式：

FFT后表达式：

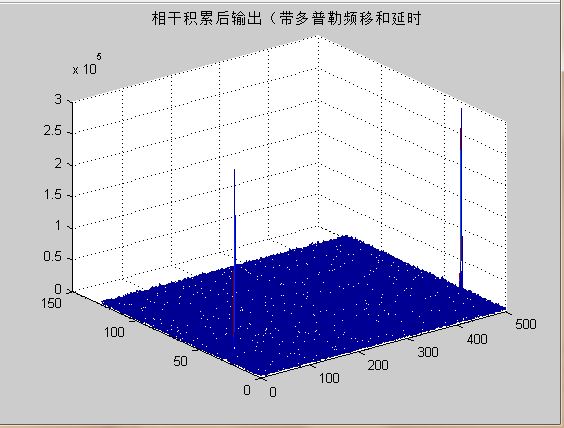
2.M序列双值电平循环自相关函数



3.回波脉压后图形（输入信噪比-5dB，目标速度100m/s，距离200m，相干累积长度500）



4.回波FFT后波形（输入信噪比-5dB，目标速度100m/s，距离200m，相干累积长度500）



5.对于M序列而言，其时宽为

带宽为

则时宽带宽积为P=127

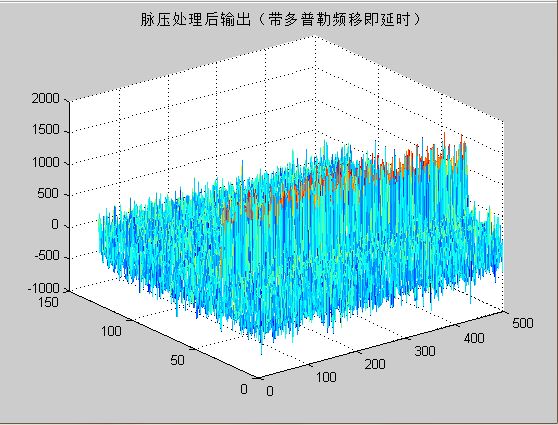
脉压后，信号的时宽变为一个码元周期T=0.0476us

时宽压缩比为p=127

而信号的带宽前后不变为

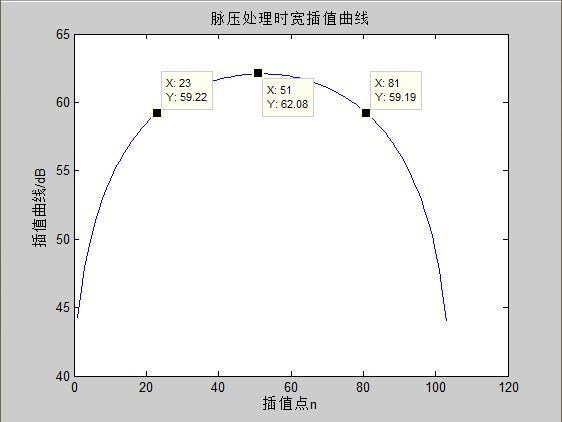
从而前后时宽带宽积比值为p=127.

对于0多普勒率的回波而言，其脉压处理如下：



计算后其增益为21.0155dB=126.34127=时宽带宽积P

对于脉压信号的时宽，通过插值的方法作出其时宽曲线从而得到脉压信号的时宽，其时宽插值曲线如下：

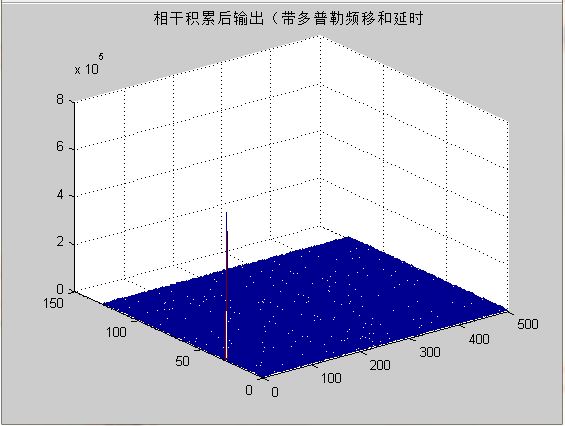


注：插值103点，每个插值点之间距离对应时间为

从图上可以的出，其3db的时宽为1.14T

对于相干累积（FFT处理）后，信号功率增强N2倍，而噪声增强N倍，故而FFT处理增益为N=500

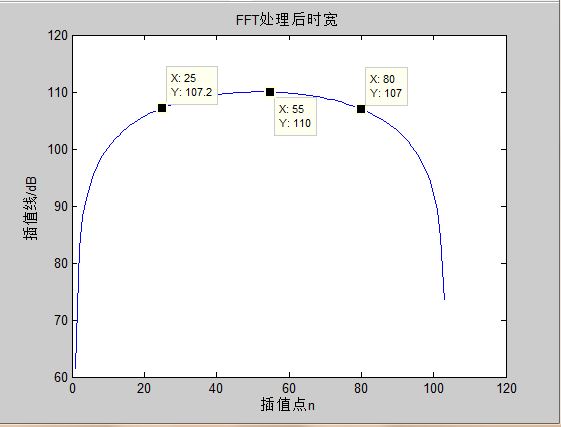
对于0多普勒率的回波而言，其FFT处理如下：



计算得到的增益26.9809=498.99500=N

对于FFT处理后的时宽同样通过对距离轴插值求得其时宽

插值曲线如下：

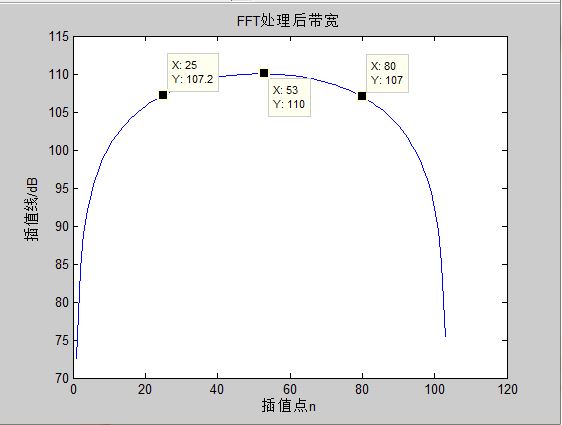


注：插值103点，每个插值点之间距离对应时间为

从图中可以看出3db时宽为1.08T

对于FFT处理后的带宽同样可以通过对速度轴插值求得其带宽

插值曲线如下：



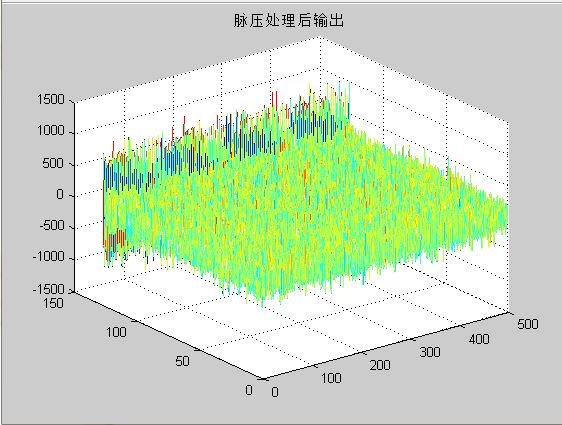
注：插值103点，每个插值点间距离对应频率

从中可以知道FFT信号处理后的带宽为356.64Hz=330.7Hz

6.考虑0延时下，脉压处理对多普勒的敏感问题

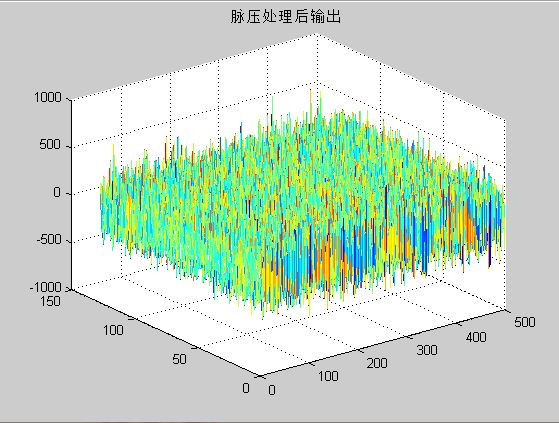
由（5）知，在0多普勒率下，脉压增益为126.4047=42dB

在考虑在多普勒率为M序列周期四分之一长度即=时，其脉压输出波形为：



计算得其增益为：20.1542dB，增益下降0.8573dB

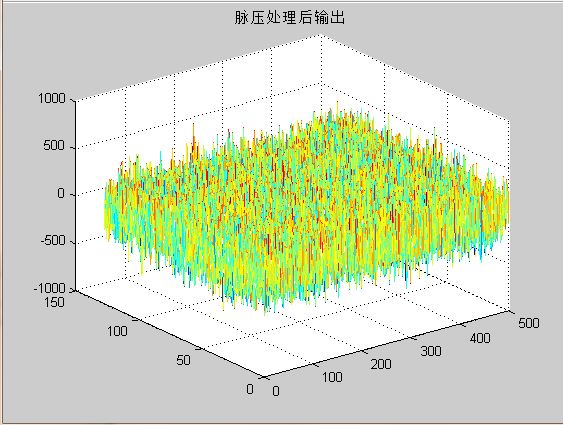
在考虑在多普勒率为M序列周期二分之一长度即=时，其脉压输出波形为：



从上图看脉压处理效果很差，几乎失效，根据计算：

得其增益为：8.5880dB，增益下降12.4235dB

当考虑在多普勒率为M序列周期长度即=时，其脉压输出波形为：



从图上看，其脉压处理失效，计算得增益比3.0674dB，下降17.9441dB,

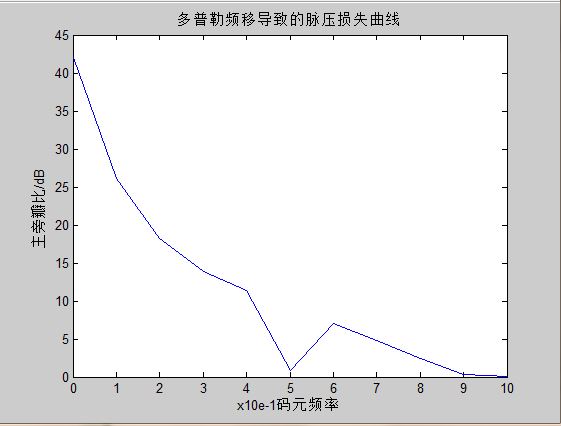
取这点作为多普勒容限的点，即多普勒容限为=

7.在不同多普勒频率下测算脉压输出后主瓣和旁瓣的比值从而绘制性能损失曲线，在测算时，不考虑高斯噪声的影响。

以1/10PT为间隔取多普勒频率，得到的主旁瓣比数据如下：

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 多普勒频率 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 主旁瓣比/dB | 42.05 | 26.15 | 18.23 | 13.82 | 11.43 | 0.83 | 6.99 | 4.75 | 2.45 | 0.30 | 0.0025 |

得到的主瓣旁瓣比的比值曲线如下：

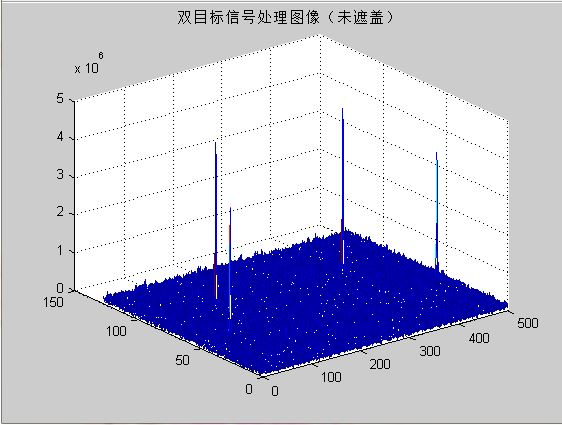


8.双目标时，当两个目标回波的强度不同时，会导致大目标覆盖小目标的现象

取大目标回波幅度为300，速度200m/s，距离1200m时

小目标回波幅度也为300，速度600m/s，距离600m时

信号处理结果为：

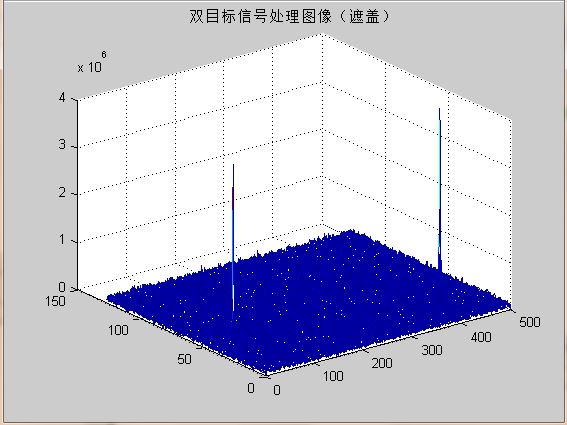


此时可以清楚的区分出两个目标

取大目标回波幅度为300，速度200m/s，距离1200m时

小目标回波幅度为20，速度600m/s，距离600m时

信号处理结果为：



此时在图上只能看到一个大目标的信号，而小目标则被大目标的旁瓣遮盖

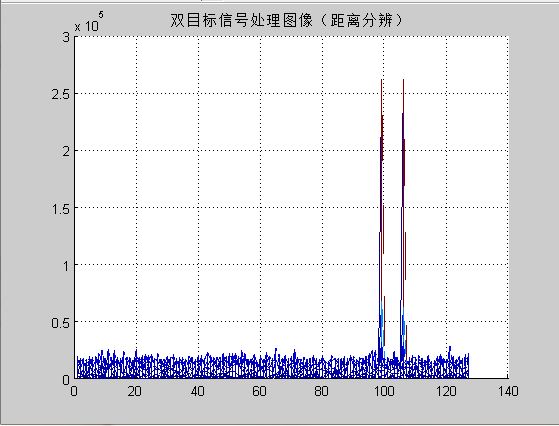
9.当两个目标的距离差在一定范围内时，雷达将无法分辨两个目标，此外当两个目标的速度差在一定范围内时，雷达将无法分辨两个目标在速度上的差别。

9.1 从理论上分析距离的分辨力应当是延迟一个码元时间所表示目标的距离，根据延迟时间t与距离的关系,其中R为距离，而后得到距离分辨力为

=7.1468m

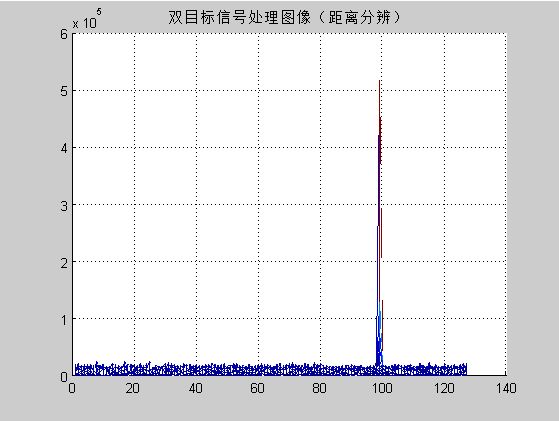
其中为一个码元的频率，此处为21Mhz

当两个目标的速度同为200m/s时，一个目标的距离为700m，而另一个目标的距离为750m时，信号处理在距离轴上的结果为：



从图上看，可以分辨出二者的距离

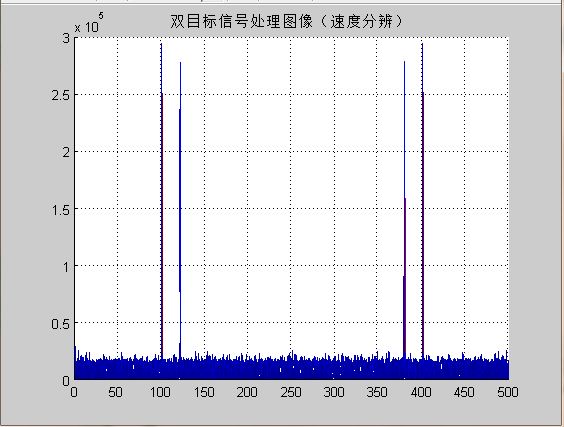
当两个目标的速度同为200m/s时，一个目标的距离为700m，而另一个目标的距离为705m时，信号处理在距离轴上的结果为：



从图上看，二者的距离差距表现不出来。

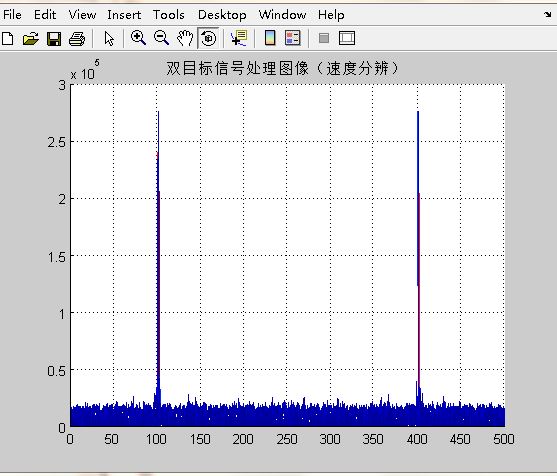
9.2 从理论上分析，速度的分辨率应当是在速度轴上一个单位所代表的速度差，由多普勒频率与目标在距离轴位置的关系，其中为采样频率，此处为，k为目标在距离轴上的位置点；从而得到双目标的速度分辨率为： =4.96m/s

当两个目标的距离同为700m时，一个目标的速度为496m/s，而另一个目标的速度为600m/s时，信号处理在速度轴上的结果为：



从图上看，可以看出二者在速度上的差别。

当两个目标的距离同为700m时，一个目标的速度为496m/s，而另一个目标的速度为500m/s时，信号处理在速度轴上的结果为：



从图上看，无法分辨出二者在速度上的差别

附MATLAB程序：

1.M序列产生对应M\_creater.m：

%求127点M序列,并绘制自相关函数%

clear all;

register=[1,0,1,0,1,0,1];%寄存器初始状态

n=length(register);

m=2^n-1;

model=[1,0,0,0,0,0,1];%反馈系数

for i=1:m

m\_line(i)=register(n);

n\_register(1)=mod(sum(register.\*model),2);

for j=2:n

n\_register(j)=register(j-1);

end

register=n\_register;

end

m\_dline=m\_line\*2-1;

2.绘制出M自相关函数，对应plotM\_rx.m

%绘制M序列双值自相关函数%

M\_creater;

t=-2\*m:1:2\*m;

for i=1:4\*m+1

m\_dline\_r=m\_sh(m\_dline,t(i));

rx(i)=sum(m\_dline\_r.\*m\_dline);

rx(i)=rx(i)/m;

end

stem(t,rx)

title('M序列双值自相关函数');

xlabel('n');

ylabel('Rx(n)');

**其中m\_sh 函数定义如下：**

%完成M序列自相关移位函数%

function y=m\_sh(m\_dline,t)

m=length(m\_dline);

if (t==0)

y=m\_dline;

elseif (t>0)

t=rem(t,m);

a\_0=[ones(1,m-t),zeros(1,t)];

b\_0=[zeros(1,m-t),ones(1,t)];

a=m\_dline.\*a\_0;

b=m\_dline.\*b\_0;

a\_1=find (a==0);

b\_1=find(b==0);

a(a\_1)=[];

b(b\_1)=[];

y=[b,a];

else (t<0)

t=rem(t,m);

a\_0=[ones(1,-t),zeros(1,m+t)];

b\_0=[zeros(1,-t),ones(1,m+t)];

a=m\_dline.\*a\_0;

b=m\_dline.\*b\_0;

a\_l=find (a==0);

b\_1=find(b==0);

a(a\_l)=[];

b(b\_1)=[];

y=[b,a];

end

3. 单目标信号处理仿真脉压，FFT图像并验证时宽带宽 对应single\_process.m

%单目标信号处理仿真验证时宽带宽

M\_creater;%产生127点M序列m\_dline

%回波参数

fe=21e6;%码频

p=127;%码元数

fg=10e9;%载频

t=10e-3;%相干累积最大时长

N=500;%相干累积长度

c=3e8;%光速

snr=-5;%输入端信噪比

%目标参数

Am=20;%目标幅度 范围1-100

%v=0;

v=0;%目标速度 范围0-1000

destine=200;%目标距离 范围 0-10000

%给定目标的速度和距离下N段信号

delay=2\*destine/c; %由距离产生回波信号延迟

delay=fix(delay\*fe);

s\_in=Am.\*repmat(m\_dline,1,N)./2; %0延迟0多普率回波

s\_in=[zeros(1,delay+1),s\_in]; %加入延迟后的回波

s\_in=awgn(s\_in,snr,'measured'); %加入高斯噪声后的回波

fd=2\*v\*fg/c; %多普勒频移

for i=delay+1:N\*p+delay+1;

s\_in(i)=s\_in(i)\*cos(2\*pi\*fd/fe\*(i-1)); %加入多普勒频移后的回波信号

end

%脉压处理—匹配滤波器

ht=fliplr(m\_dline);

s\_out=conv(s\_in,ht); %时域卷积完成匹配滤波

%距离门重排

for j=1:N

for i=1:p

s\_out\_r(i,j)=s\_out((j-1)\*p+i);

end

end

%求脉压时宽插值

k=100;

y=0;

x=1:k+3;

for j=1:N

a\_line=(abs(s\_out\_r(:,j)))';

a\_max=max(a\_line);

a\_label=find(a\_line==a\_max);

a\_label=[a\_label-1,a\_label,a\_label+1];

y=y+a\_line(a\_label);

end

y=y/N;

yi=inter\_edit(y,k);

yi\_db=20\*log10(yi);

figure(3);

plot(x,yi\_db)

title('脉压处理时宽插值曲线');

xlabel('插值点n');

ylabel('插值曲线/dB');

%相干累加—FFT处理

for i=1:p

s\_out\_r\_f(i,:)=abs(fft(s\_out\_r(i,:)));

end

%求相干累加时宽带宽

if (v>0)

k=100;

p\_max=max(max(abs(s\_out\_r\_f)));

[x,y]=find(s\_out\_r\_f==p\_max);

f\_array=abs((s\_out\_r\_f(:,y(1)))');%求时宽

f\_line=abs(s\_out\_r\_f(x(1),:));%求带宽

yi\_t=f\_array([x(1)-1,x(1),x(1)+1]);

yi=inter\_edit(yi\_t,k);

figure(4);

yi=20\*log10(yi);

x=1:k+3;

plot(x,yi)

title('FFT处理后时宽');

xlabel('插值点n');

ylabel('插值线/dB');

yi\_t=f\_line([y(1)-1,y(1),y(1)+1]);

yi=inter\_edit(yi\_t,k);

figure(5);

yi=20\*log10(yi);

x=1:k+3;

plot(x,yi)

title('FFT处理后带宽');

xlabel('插值点n');

ylabel('插值线/dB');

end

%计算脉压和FFT增益比

true=ADD\_count(snr,fd,destine,m\_dline,2)

%绘制脉压后图像

figure(1);

mesh(1:N,1:p,s\_out\_r)

title('脉压处理后输出（带多普勒频移即延时）');

%绘制相干累积后图形

figure(2);

mesh(1:N,1:p,s\_out\_r\_f)

title('相干积累后输出（带多普勒频移和延时');

4.时宽带宽的差值函数定义 对应inter\_edit.m

function yi=inter\_edit(y,N)

x=[1,2+N/2,N+3];

xi=1:N+3;

xi(x)=[];

yi\_r=interp1(x,y,xi,'cubic');

for i=1:2

for j=1:N/2

yi\_r\_t(i,j)=yi\_r((i-1)\*N/2+j);

end

end

yi=[y(1),yi\_r\_t(1,:),y(2),yi\_r\_t(2,:),y(3)];

5.计算脉压和FFT对应两级处理的增益函数，对应文件ADD\_count.m

% 计算脉压和FFT增益

function true=ADD\_count(snr,fd,destine,m\_dline,flag)

%回波参数

fe=21e6;%码频

p=127;%码元数

fg=10e9;%载频

t=10e-3;%相干累积最大时长

N=500;%相干累积长度

c=3e8;%光速

%目标参数

Am=20;%目标幅度 范围1-100

%不含噪声的纯信号

delay=2\*destine/c; %由距离产生回波信号延迟

delay=fix(delay\*fe);

s\_in\_1=Am.\*repmat(m\_dline,1,N)./2; %0延迟0多普率回波

s\_in\_1=[zeros(1,delay+1),s\_in\_1]; %加入延迟后的回波

length\_1=length(s\_in\_1);

for i=delay+1:N\*p+delay+1;

s\_in\_1(i)=s\_in\_1(i)\*cos(2\*pi\*fd/fe\*(i-1)); %加入多普勒频移后的回波信号

end

%脉压处理—匹配滤波器（带信号）

ht=fliplr(m\_dline);

s\_out\_1=conv(s\_in\_1,ht); %时域卷积完成匹配滤波

%距离门重排

for j=1:N

for i=1:p

s\_out\_r\_1(i,j)=s\_out\_1((j-1)\*p+i);

end

end

%相干累加

for i=1:p

s\_out\_r\_f\_1(i,:)=abs(fft(s\_out\_r\_1(i,:)));

end

%没有信号的纯噪声输入

s\_in\_2=awgn(s\_in\_1,snr,'measured'); %加入高斯噪声后的回波

s\_in\_2=s\_in\_2-s\_in\_1;%纯噪声信号

%脉压处理—匹配滤波器（不带信号）

ht=fliplr(m\_dline);

s\_out\_2=conv(s\_in\_2,ht); %时域卷积完成匹配滤波

%距离门重排

for j=1:N

for i=1:p

s\_out\_r\_2(i,j)=s\_out\_2((j-1)\*p+i);

end

end

%相干累加

for i=1:p

s\_out\_r\_f\_2(i,:)=abs(fft(s\_out\_r\_2(i,:)));

end

%计算脉压增益

enery\_s\_1=0;

for j=1:N

p\_max(j)=max(abs(s\_out\_r\_1(:,j)));

enery\_s\_1=enery\_s\_1+p\_max(j)^2;

end

enery\_s\_1=enery\_s\_1/N;

enery\_n\_1=0;

for j=1:N

for i=1:p

enery\_n\_1=enery\_n\_1+ s\_out\_r\_2(i,j)^2;

end

end

enery\_n\_1=enery\_n\_1/N/p;%计算噪声功率

snr\_out\_1=10\*log10(enery\_s\_1/enery\_n\_1);

snr\_out\_1\_add=snr\_out\_1-snr

%计算FFT增益

if (flag==2)

p\_max=max(max(abs(s\_out\_r\_f\_1)));%找到信号峰值点

[x,y]=find(s\_out\_r\_f\_1==p\_max);

enery\_s\_2=0;

if (length(x)==1)

enery\_s\_2=enery\_s\_2+s\_out\_r\_f\_1(x,y)^2;

enery\_s\_2=enery\_s\_2;

else

for i=1:2

enery\_s\_2=enery\_s\_2+s\_out\_r\_f\_1(x(i),y(i))^2;

end

enery\_s\_2=enery\_s\_2/2;

end

enery\_n\_2=0;

for j=1:N

for i=1:p

enery\_n\_2=enery\_n\_2+ s\_out\_r\_f\_2(i,j)^2;

end

end

enery\_n\_2=enery\_n\_2/N/p;%计算噪声功率

snr\_out\_2=10\*log10(enery\_s\_2/enery\_n\_2);%计算脉压输出端信噪比

snr\_out\_2\_add=snr\_out\_2-snr\_out\_1

end

true=1;

6.仿真脉压处理对多普勒敏感的现象，对应douple.m

%仿真单目标时脉压处理对多普勒频移的敏感关系，在仿真中不考虑延迟的问题（匹配滤波器对时延自适应）

%包括：1.多普勒敏感现象，在这里取无多普勒（fd=0），小多普勒（fd=1/4个发射波周期），多普勒容限时（fd=发射波周期）

% 2.绘制性能损失曲线（脉压主旁瓣与多普勒频移关系曲线）

%单目标信号处理仿真

M\_creater;%产生127点M序列m\_dline

%回波参数

fe=21e6;%码频

p=127;%码元数

fg=10e9;%载频

t=10e-3;%相干累积最大时长

N=500;%相干累积长度

c=3e8;%光速

snr=-5;%输入端信噪比

Am=20;%回波幅度

destine=0;

%0延迟多普勒回波信号

s\_in=repmat(m\_dline,1,N); %0延迟0多普率回波

s\_in=Am/2.\*awgn(s\_in,snr,'measured'); %加入高斯噪声后的回波

%fd=0; %多普勒频移1

%fd=fe/p/4;%多普勒频移2

fd=fe/p/2;

%fd=fe/p;%多普勒频移3；

i=1:p\*N;

s\_in=s\_in.\*cos(2\*pi\*fd/fe\*i); %加入多普勒频移后的回波信号

%脉压处理—匹配滤波器

ht=fliplr(m\_dline);

s\_out=conv(s\_in,ht); %时域卷积完成匹配滤波

%距离门重排

for j=1:N

for i=1:p

s\_out\_r(i,j)=s\_out((j-1)\*p+i);

end

end

true=ADD\_count(snr,fd,destine,m\_dline,1)

%绘制脉压后图像

figure(1);

mesh(1:N,1:p,s\_out\_r)

title('脉压处理后输出');

7.绘制脉压性能损失曲线，对应文件douple\_2.m

%绘制性能损失曲线

clear all;

M\_creater;

%回波参数

Am=20;

fe=21e6;

p=127;

snr=-10;

N=500;

%绘制性能损失曲线

for k=1:11

%0延迟多普勒回波信号

s\_in=repmat(m\_dline,1,N); %0延迟0多普率回波

%s\_in=awgn(s\_in,snr,'measured');%加入高斯噪声后的回波

fd=(k-1)/10\*fe/p;

i=0:p\*N-1;

s\_in=s\_in.\*cos(2\*pi\*fd/fe\*i); %加入多普勒频移后的回波信号

%脉压处理—匹配滤波器

ht=fliplr(m\_dline);

s\_out=conv(s\_in,ht); %时域卷积完成匹配滤波

%距离门重排

for j=1:N

for i=1:p

s\_out\_r(i,j)=s\_out((j-1)\*p+i);

end

end

for j=1:N

a\_line=(s\_out\_r(:,j))';

a\_line=abs(a\_line);

t\_max=max(a\_line);

a\_label=find(a\_line==t\_max);

a\_line(a\_label)=[];

p\_max=max(a\_line);

d\_p(j)=t\_max/p\_max;

end

s\_p(k)=sum(d\_p)/N;

end

s\_p=20\*log10(s\_p);

k=0:10;

plot(k,s\_p)

title('多普勒频移导致的脉压损失曲线');

xlabel('x10e-1码元频率');

ylabel('主旁瓣比/dB');

8.验证双目标时，大目标旁瓣掩盖小目标旁瓣的现象，对应double\_process\_1.m

%双目标仿真1 大旁瓣掩盖小旁瓣

%仿真：1.仿真出大目标旁瓣覆盖小目标旁瓣的现象

% 2.仿真出距离分辨率和速度分辨率的大小

M\_creater;%产生127点M序列m\_dline

%回波参数

fe=21e6;%码频

p=127;%码元数

fg=10e9;%载频

t=10e-3;%相干累积最大时长

N=500;%相干累积长度

c=3e8;%光速

snr=-10;%输入端信噪比

%大目标参数

Am\_1=300;%目标幅度 范围1-100

v\_1=200;%目标速度 范围0-1000

destine\_1=1200;%目标距离 范围 0-10000

%给定目标的速度和距离下N段信号

delay\_1=2\*destine\_1/c; %由距离产生回波信号延迟

delay\_1=fix(delay\_1\*fe);

s\_in\_1=Am\_1.\*repmat(m\_dline,1,N)./2; %0延迟0多普率回波

s\_in\_1=[zeros(1,delay\_1+1),s\_in\_1]; %加入延迟后的回波

s\_in\_1=awgn(s\_in\_1,snr,'measured'); %加入高斯噪声后的回波

fd\_1=2\*v\_1\*fg/c; %多普勒频移

%fd=fe/p/4;

for i=delay\_1+1:N\*p+delay\_1+1;

s\_in\_1(i)=s\_in\_1(i)\*cos(2\*pi\*fd\_1/fe\*(i-1)); %加入多普勒频移后的回波信号

end

%小目标参数

%Am\_2=300;%目标幅度 范围1-100

Am\_2=20;

v\_2=600;%目标速度 范围0-1000

destine\_2=600;%目标距离 范围 0-10000

%给定目标的速度和距离下N段信号

delay\_2=2\*destine\_2/c; %由距离产生回波信号延迟

delay\_2=fix(delay\_2\*fe);

s\_in\_2=Am\_2.\*repmat(m\_dline,1,N)./2; %0延迟0多普率回波

s\_in\_2=[zeros(1,delay\_2+1),s\_in\_2]; %加入延迟后的回波

s\_in\_2=awgn(s\_in\_2,snr,'measured'); %加入高斯噪声后的回波

fd\_2=2\*v\_2\*fg/c; %多普勒频移

%fd=fe/p/4;

for i=delay\_2+1:N\*p+delay\_2+1;

s\_in\_2(i)=s\_in\_2(i)\*cos(2\*pi\*fd\_2/fe\*(i-1)); %加入多普勒频移后的回波信号

end

%脉压处理—匹配滤波器

ht=fliplr(m\_dline);

s\_out\_1=conv(s\_in\_1,ht); %时域卷积完成匹配滤波

s\_out\_2=conv(s\_in\_2,ht);

%距离门重排

for j=1:N

for i=1:p

s\_out\_r\_1(i,j)=s\_out\_1((j-1)\*p+i);

end

end

for j=1:N

for i=1:p

s\_out\_r\_2(i,j)=s\_out\_2((j-1)\*p+i);

end

end

s\_out\_r=s\_out\_r\_1+s\_out\_r\_2;

%相干累加—FFT处理

for i=1:p

s\_out\_r\_f(i,:)=abs(fft(s\_out\_r(i,:)));

end

%绘制处理后图形

figure(1);

mesh(1:N,1:p,s\_out\_r\_f)

%title('双目标信号处理图像（未遮盖）');

title('双目标信号处理图像（遮盖）');

9.仿真双目标距离分辨与速度分辨的情况 对应文件double\_process\_2.m

%双目标仿真2 仿真出距离分辨和速度分辨的大小

%仿真：1.仿真出大目标旁瓣覆盖小目标旁瓣的现象

% 2.仿真出距离分辨率和速度分辨率的大小

M\_creater;%产生127点M序列m\_dline

%回波参数

fe=21e6;%码频

p=127;%码元数

fg=10e9;%载频

t=10e-3;%相干累积最大时长

N=500;%相干累积长度

c=3e8;%光速

snr=-10;%输入端信噪比

%大目标参数

Am\_1=20;%目标幅度 范围1-100

%距离检测

v\_1=200;%目标速度 范围0-1000

destine\_1=700;%目标距离 范围 0-10000

destine\_1=700.1;

%速度检测

%destine\_1=700;

%v\_1=496;

%给定目标的速度和距离下N段信号

delay\_1=2\*destine\_1/c; %由距离产生回波信号延迟

delay\_1=fix(delay\_1\*fe);

s\_in\_1=Am\_1.\*repmat(m\_dline,1,N)./2; %0延迟0多普率回波

s\_in\_1=[zeros(1,delay\_1+1),s\_in\_1]; %加入延迟后的回波

s\_in\_1=awgn(s\_in\_1,snr,'measured'); %加入高斯噪声后的回波

fd\_1=2\*v\_1\*fg/c; %多普勒频移

for i=delay\_1+1:N\*p+delay\_1+1;

s\_in\_1(i)=s\_in\_1(i)\*cos(2\*pi\*fd\_1/fe\*(i-1)); %加入多普勒频移后的回波信号

end

%小目标参数

Am\_2=20;%目标幅度 范围1-100

%距离检测

v\_2=200;%目标速度 范围0-1000

%destine\_2=710;%目标距离 范围 0-10000

destine\_2=705;

%速度检测

%destine\_2=700;

%v\_2=500;

%给定目标的速度和距离下N段信号

delay\_2=2\*destine\_2/c; %由距离产生回波信号延迟

delay\_2=fix(delay\_2\*fe);

s\_in\_2=Am\_2.\*repmat(m\_dline,1,N)./2; %0延迟0多普率回波

s\_in\_2=[zeros(1,delay\_2+1),s\_in\_2]; %加入延迟后的回波

s\_in\_2=awgn(s\_in\_2,snr,'measured'); %加入高斯噪声后的回波

fd\_2=2\*v\_2\*fg/c; %多普勒频移

for i=delay\_2+1:N\*p+delay\_2+1;

s\_in\_2(i)=s\_in\_2(i)\*cos(2\*pi\*fd\_2/fe\*(i-1)); %加入多普勒频移后的回波信号

end

%脉压处理—匹配滤波器

ht=fliplr(m\_dline);

s\_out\_1=conv(s\_in\_1,ht); %时域卷积完成匹配滤波

s\_out\_2=conv(s\_in\_2,ht);

%距离门重排

for j=1:N

for i=1:p

s\_out\_r\_1(i,j)=s\_out\_1((j-1)\*p+i);

end

end

for j=1:N

for i=1:p

s\_out\_r\_2(i,j)=s\_out\_2((j-1)\*p+i);

end

end

s\_out\_r=s\_out\_r\_1+s\_out\_r\_2;

%相干累加—FFT处理

for i=1:p

s\_out\_r\_f(i,:)=abs(fft(s\_out\_r(i,:)));

end

%绘制处理后图形

figure(1);

mesh(1:N,1:p,s\_out\_r\_f)

title('双目标信号处理图像（距离分辨）');

%检测程序

delta\_r=c/2/fe

p\_max=max(max(s\_out\_r\_f));

[x,y]=find(s\_out\_r\_f>(p\_max/1.05))

%title('双目标信号处理图像（速度分辨）');

%delta\_v=fe\*c/2/p/N/fg

%p\_max=max(max(s\_out\_r\_f));

%[x,y]=find(s\_out\_r\_f>(p\_max/1.05))