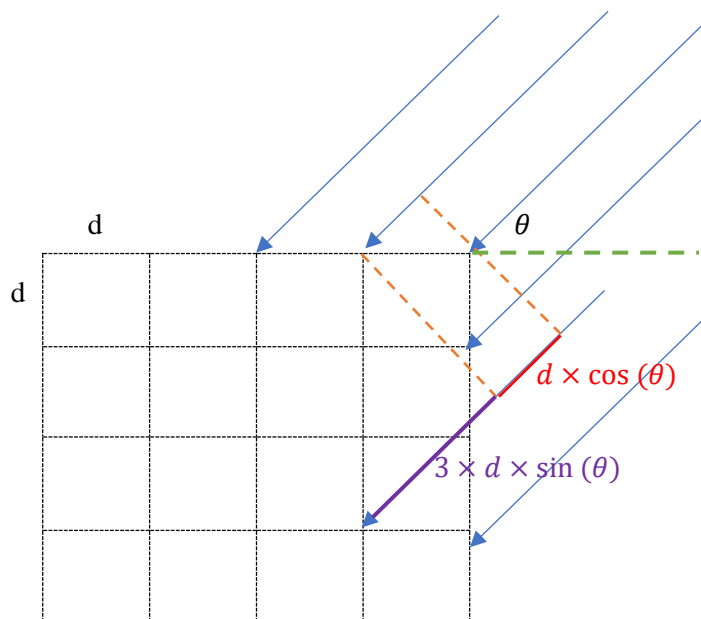


به نام خدا

پیاده سازی روش DAS

سعید اکبری 9817603

در این آزمایش علاوه بر آرایه خطی به آرایه صفحه ای نیز نیاز داریم. تاخیر سیگنال در آرایه صفحه ای برای میکروفونی که در سطر  $i$ ام و ستون  $j$ ام قرار دارد طبق هندسه مسطحه به صورت زیر محاسبه می شود:



$$\tau = \frac{(i - 1) \times d \times \cos(\theta) + (j - 1) \times d \times \sin(\theta)}{\text{Sound Velocity}}$$

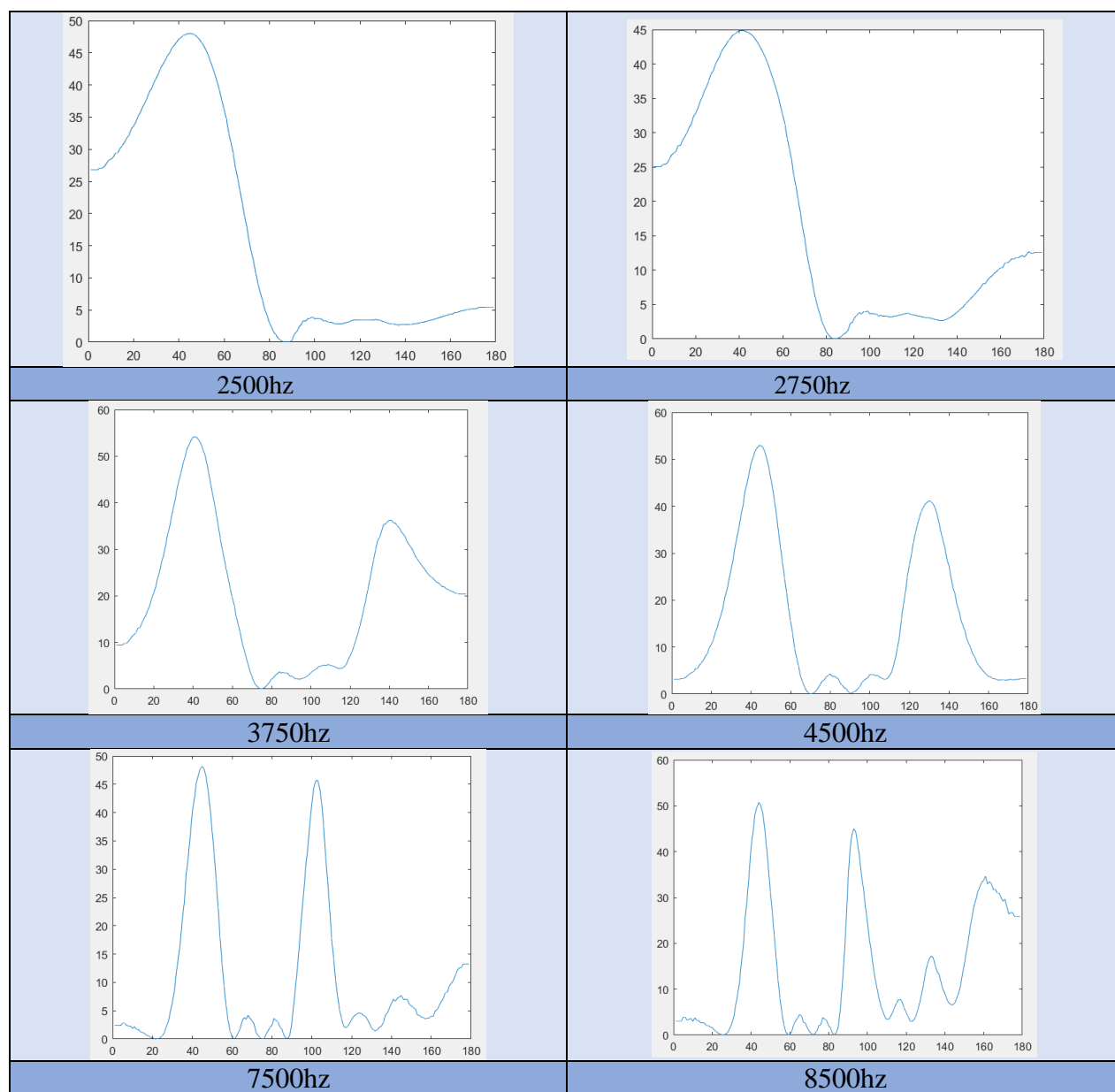
در نتیجه تعداد نمونه های تاخیر به صورت زیر است:

$$n = \tau \times Fs = \frac{(i - 1) \times d \times \cos(\theta) + (j - 1) \times d \times \sin(\theta)}{\text{Sound Velocity}} \times Fs$$

با توجه به این روابط و ایجاد سیگنال های تاخیر یافته و سپس در قسمت دریافت سیگنال، جبران تاخیر و جمع کردن سیگنال ها و میانگین گیری در نرم افزار متلب، الگوریتم DAS را پیاده سازی می کنیم.

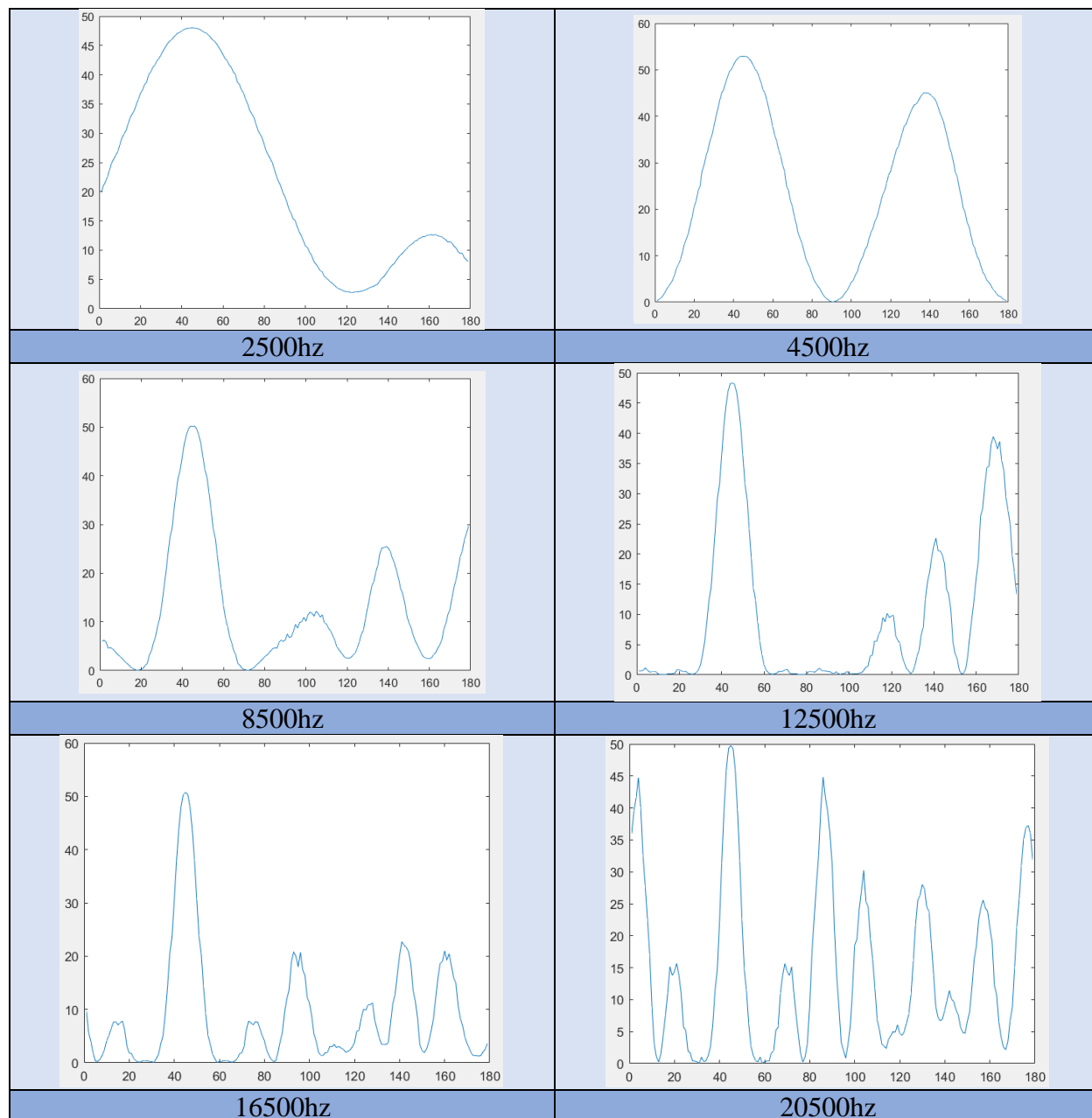
## محدود کردن فرکانس

تصاویر زیر در شرایطی که زاویه 45 درجه و  $d=5\text{cm}$  و با چهار میکروفون خطی بدست آمده است:



با توجه به تصاویر بدست آمده، هرچه فرکانس بیشتر می شود رزولوشن نیز افزایش می یابد و همچنین تعداد بالازدگی ها نیز بیشتر می شود.

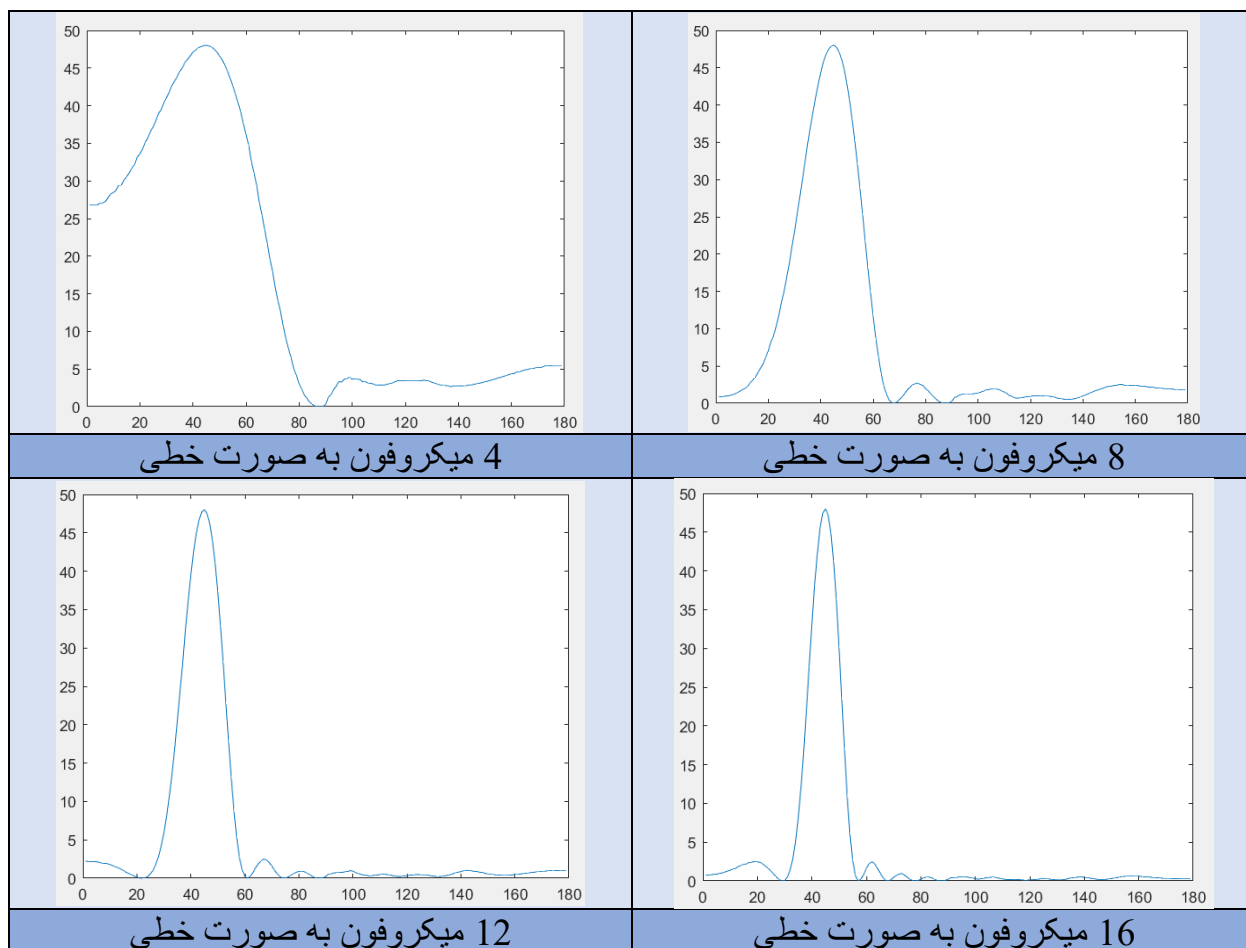
تصاویر زیر در شرایطی که زاویه 45 درجه و  $d=5\text{cm}$  و با چهار میکروفون که به صورت صفحه ای قرار گرفته اند بدست آمده است:



در این حالت نیز با افزایش فرکانس ضمن بهتر شدن رزولوشن، تعداد بالازدگی ها نیز افزایش می یابد؛ تفاوتی که این حالت با حالت آرایه خطی میکروفون ها دارد این است که در یک فرکانس مشخص رزولوشن آرایش خطی بیشتر از رزولوشن آرایه صفحه ای است.

## اثر افزایش تعداد المان های گیرنده با فرض ثابت بودن فاصله گیرنده ها

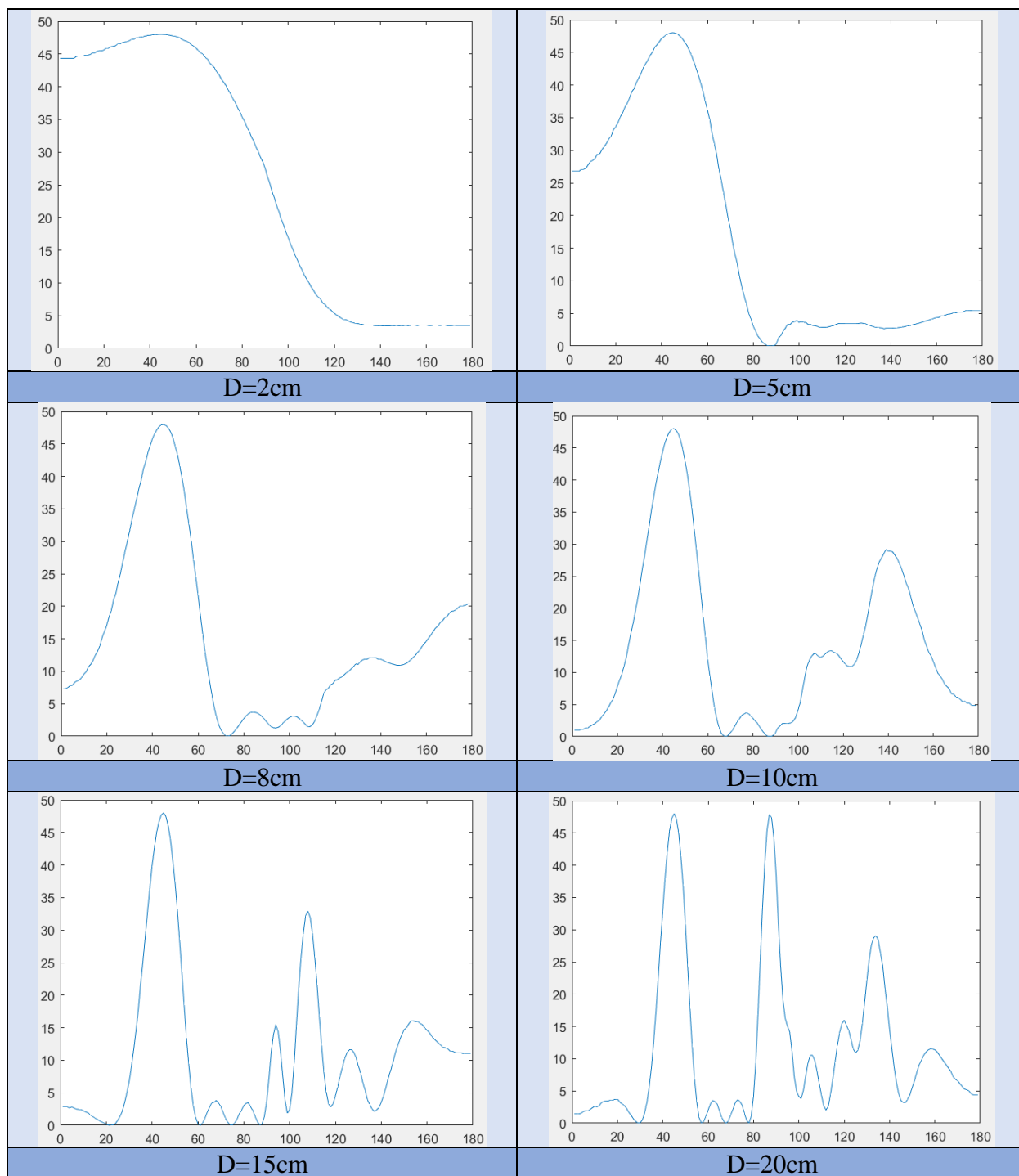
تصاویر زیر در شرایطی که زاویه 45 درجه و  $d=5\text{cm}$  و با  $f=4500\text{hz}$  بدست آمده است:



مشاهده می شود هرچه تعداد میکروفون ها افزایش می یابد، رزولوشن بهتر می شود. در قسمت های بعد زمانی که آرایش صفحه ای و خطی را مقایسه می کنیم مشاهده می شود که در حالت کلی با تعداد میکروفون برابر رزولوشن آرایه خطی بیشتر است.

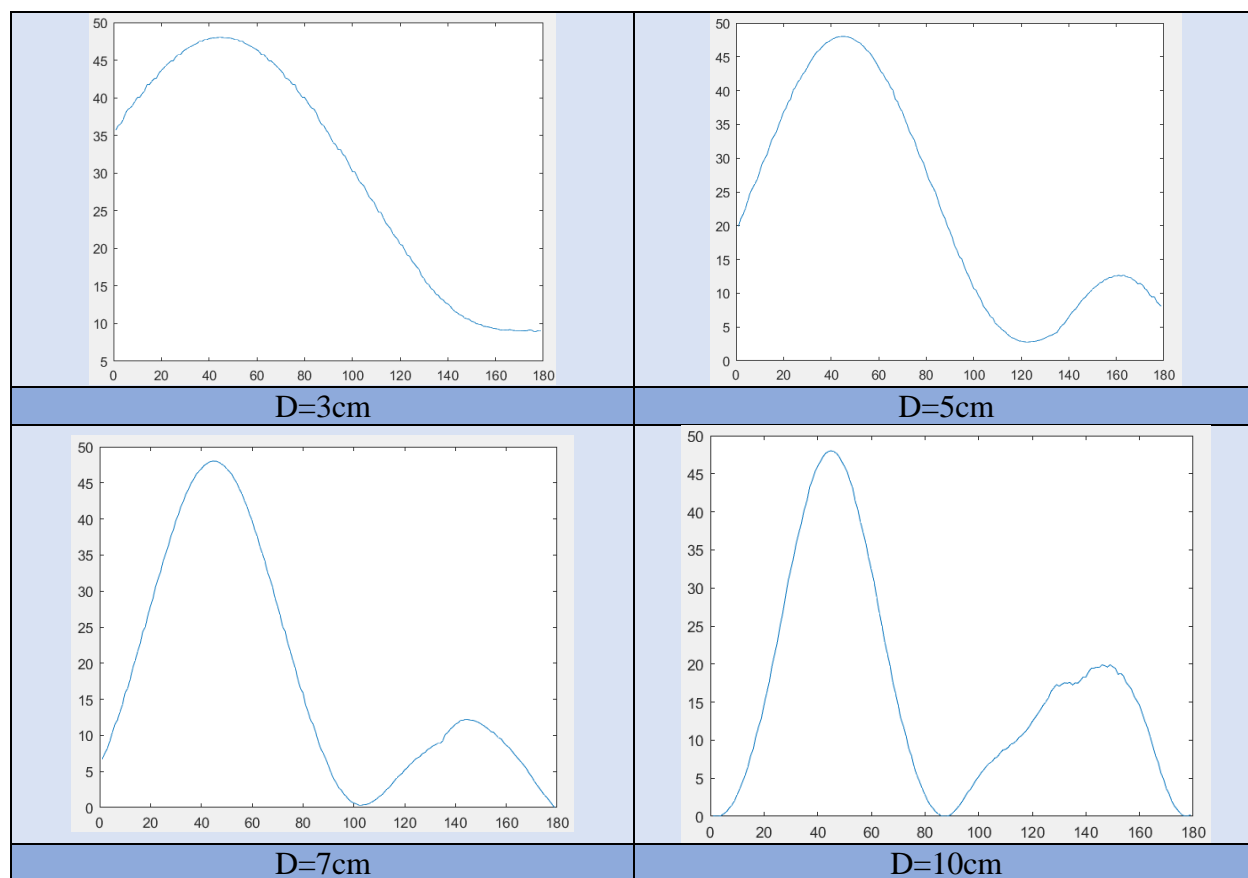
## اثر افزایش فاصله بین المان ها با فرض ثابت بودن تعداد المان ها

تصاویر زیر در شرایطی که زاویه 45 درجه و  $f=2500\text{hz}$  و با چهار میکروفون خطی بدست آمده است:



مشاهده می شود با افزایش فاصله رزولوشن بهتر شده اما تعداد قله ها افزایش می یابد.

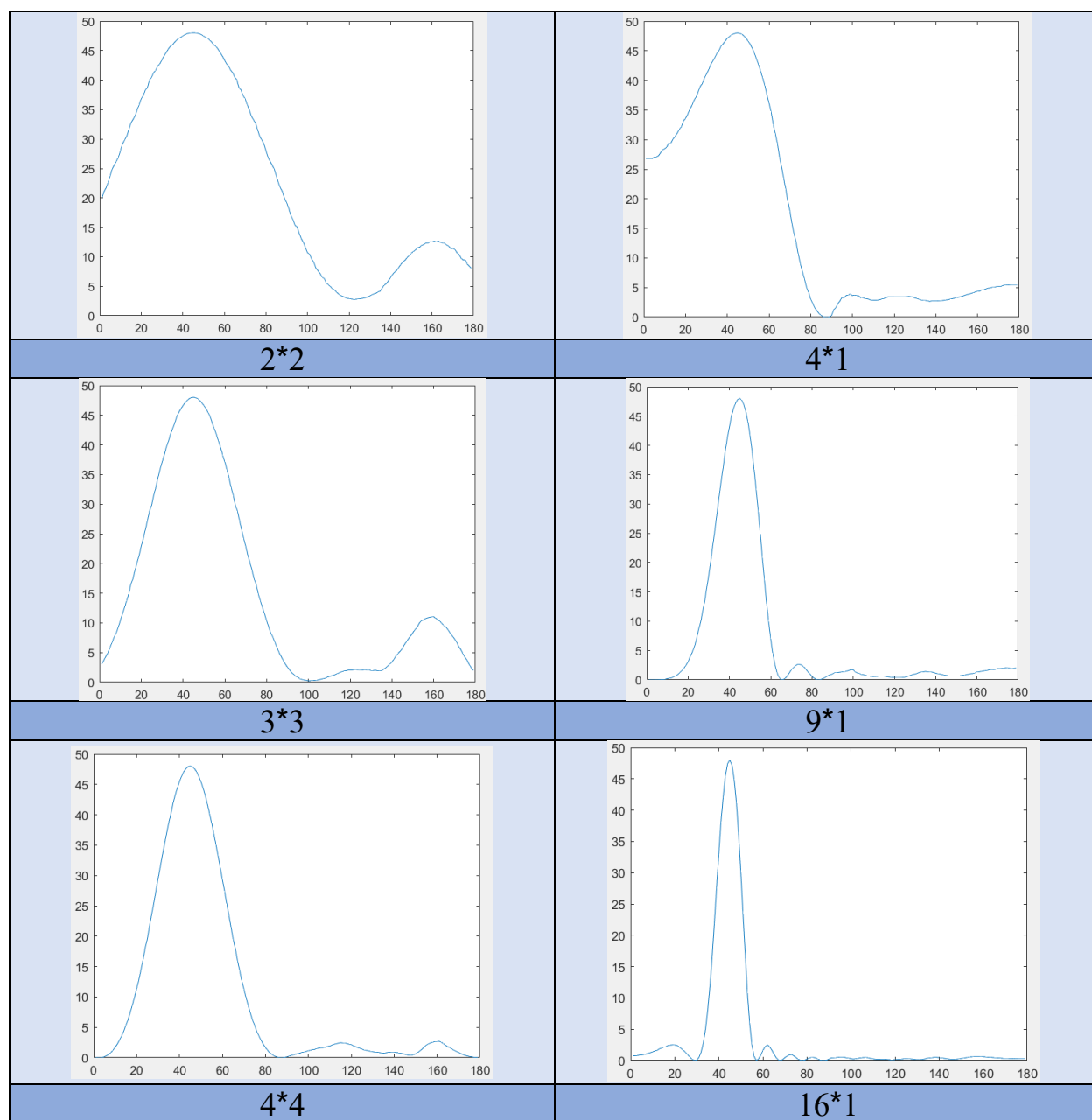
اکنون برای چینش صفحه ای با چهار میکروفون نیز این آزمایش را انجام می دهیم:



با توجه به تصاویر این دو آزمایش و آزمایش مربوط به افزایش فرکانس، می بینیم که افزایش فاصله تاثیری مشابه افزایش فرکانس دارد.

## بررسی اثر چینش آرایه

تصاویر زیر در شرایطی که زاویه 45 درجه و  $d=5\text{cm}$  و با  $f=4500\text{Hz}$  بدست آمده است:

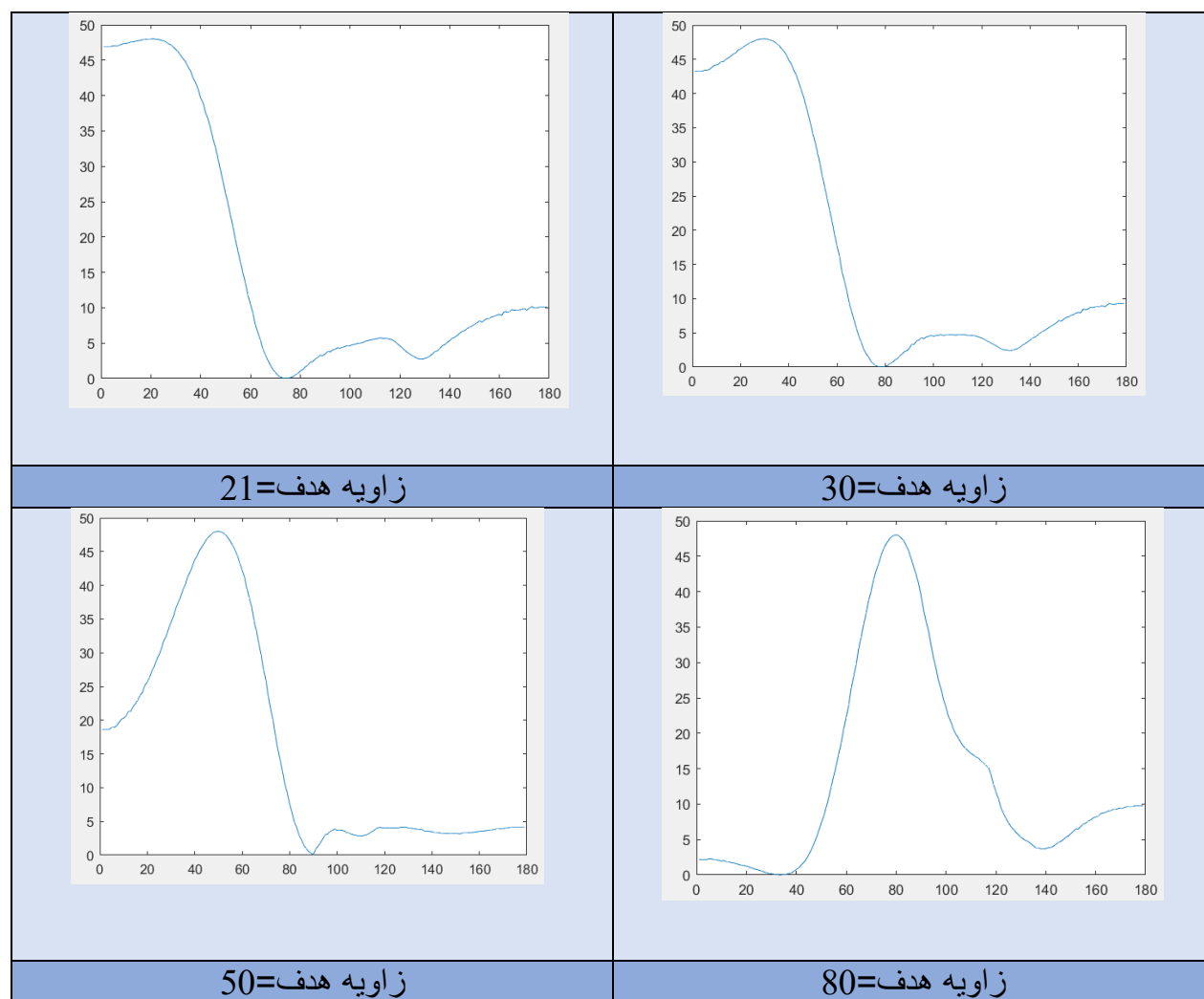


با توجه به تصاویر بالا با تعداد میکروفون برابر آرایه صفحه ای رزولوشن کمتری نسبت به آرایه خطی می دهد؛ اما باید به این نکته توجه کرد که در آرایه صفحه ای چیزی که بدست می آوریم محدوده تشخیص زاویه بیشتر است.



## بررسی حساسیت رزولوشن به زاویه هدف

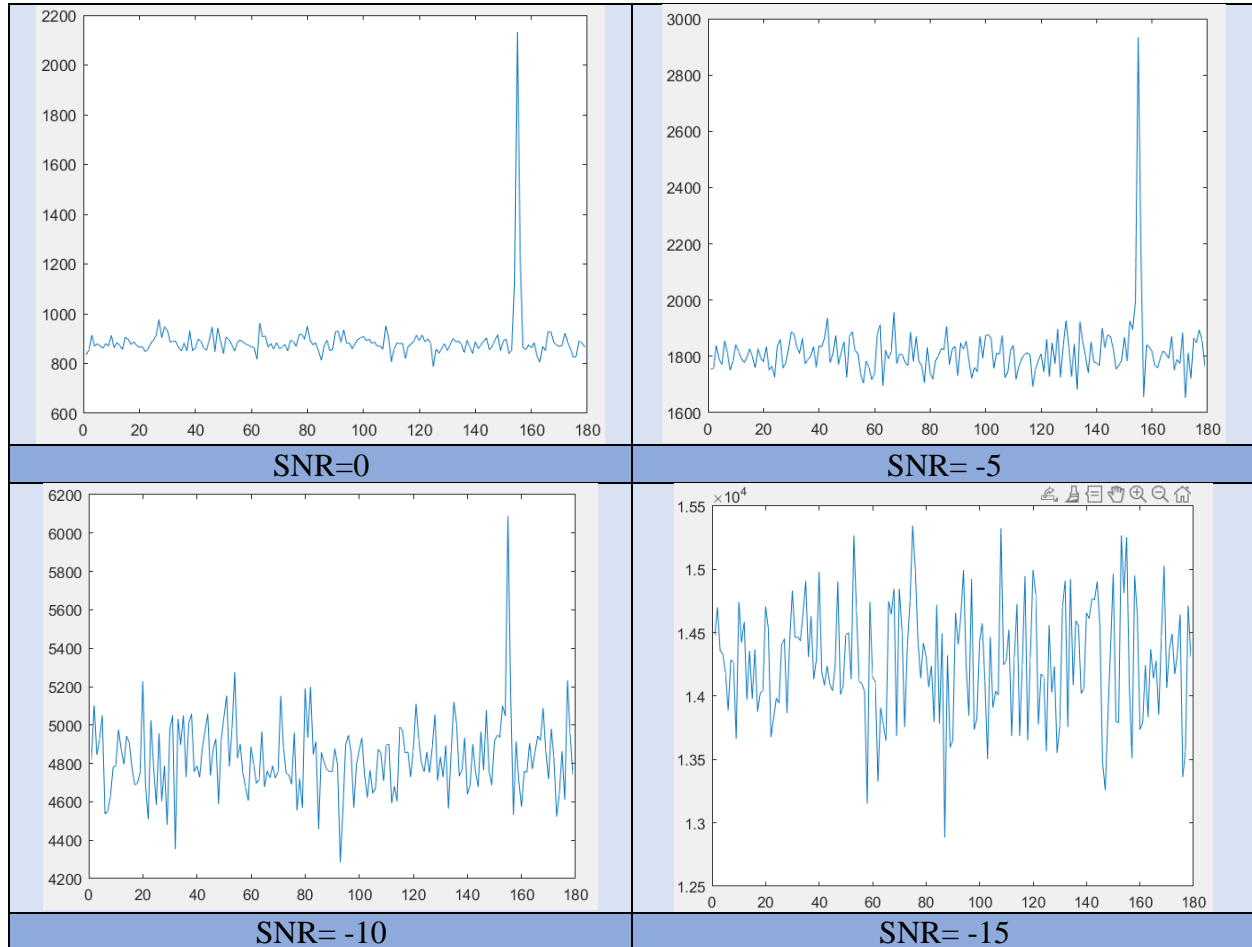
تصاویر زیر در شرایط  $f=2500\text{hz}$  و  $d=5\text{cm}$  و با چهار میکروفون خطی بدست آمده است:



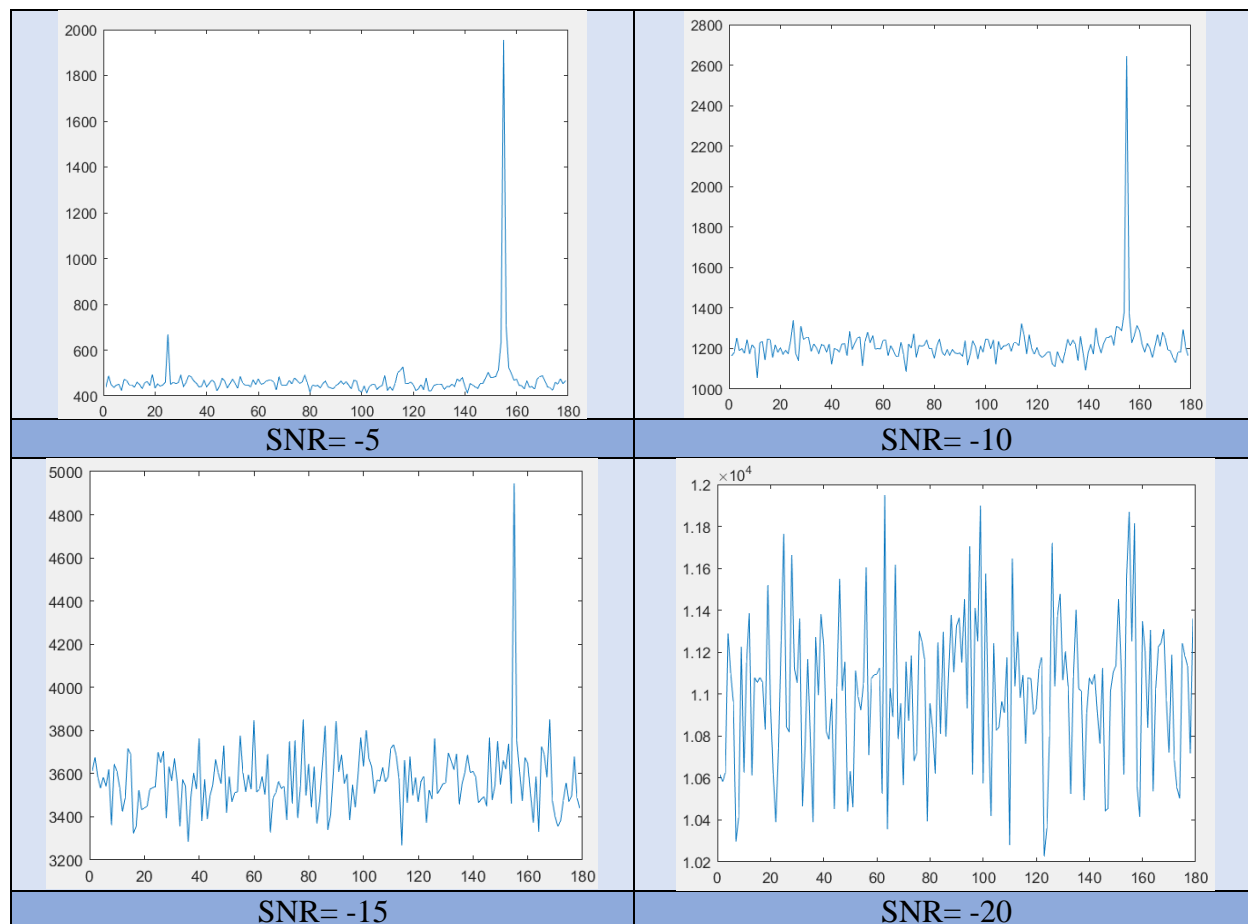
واضح است که به ازای زوایای هدفی که به 90 درجه نزدیک تر اند، رزولوشن بیشتری داریم و این به دلیل زاویه دیدی است که سیگنال مورد نظر نسبت به میکروفون ها دارد.

## روش شبیه سازی شده تا چه میزان SNR جواب می دهد؟

با کمی جست و جو می توان دید که به ازای  $SNR = -13$  دیگر این روش پاسخ گو نیست. تصاویر زیر مربوط به زاویه 45 درجه می باشد برای چهار میکروفون که به صورت خطی قرار گرفته اند:



با کمی جست و جو می توان دید که به ازای  $SNR = -20$  دیگر این روش برای 16 میکروفون که به صورت صفحه ای قرار گرفته اند پاسخگو نیست. تصاویر زیر مربوط به زاویه 45 درجه می باشد:



در ابتدا باید پارامتر های استفاده شده را مقدار دهی کنیم:

```
clc;
clear all;
%%Initialize parameters
Fs=48000;
SoundVelocity=330;
MicDistance=0.05;
t=1:1/Fs:10;
AngleRange=179;
AngleOfRadiationDegree=155;
AngleOfRadiationRad=AngleOfRadiationDegree*pi/180;
f1=5;
f2=3500;
NumOfMicInRow=4;
NumOfMicInCol=4;
InputSignal=randn(1,numel(t));
%InputSignal=sin(2*pi*f2*t);
% figure(1)
% plot(InputSignal);
```

سپس سیگنال های مورد نظر را با تاخیر مناسب می سازیم:

```
%% Signal generation
for i=1:NumOfMicInRow
    for j=1:NumOfMicInCol
        TempDelaySample=round(((i-1)*MicDistance*cos(AngleOfRadiationRad)/SoundVelocity+(j-1)*MicDistance*sin(AngleOfRadiationRad)/SoundVelocity)*Fs);
        TempInputSignal=circshift(InputSignal,TempDelaySample);
        RescivedSignal(i,j,1:numel(t))=TempInputSignal;
    end
end

%% Delay computing...
for Angle=1:AngleRange
    AngleR=Angle*pi/180;
    for i=1:NumOfMicInRow
        for j=1:NumOfMicInCol
            AngleMicDelay(i,j,Angle)=round(Fs*(((i-1)*MicDistance*cos(AngleR)+(j-1)*MicDistance*sin(AngleR))/SoundVelocity));
        end
    end
end
```

در نهایت باید تشخیص دهیم که سیگنال های مورد نظر با چه زاویه ای تابیده شده اند:

```
% Signal correction
Win1=2000;
Win2=1750;
for Angle=1:AngleRange
    SumOfSignals=zeros(1,Win2);
    for i=1:NumOfMicInRow
        for j=1:NumOfMicInCol
            for k=1:Win1
                TempSignal(k)=RescivedSignal(i,j,k);
            end
            CorrectSignal=circshift(TempSignal,-AngleMicDelay(i,j,Angle));
            Signal(1:Win2)=awgn(CorrectSignal(1:Win2),-5);
            SumOfSignals=Signal+SumOfSignals;
        end
    end
    SignalAverage=SumOfSignals/(NumOfMicInCol*NumOfMicInRow);
    SignalPower(Angle)=dot(SignalAverage,SignalAverage);
end
```

کاری که در آخر باید انجام دهیم، رسم نمودار توان برحسب زاویه و بدست آوردن زاویه مورد نظر است:

```
% Finding angle
for i=1:AngleRange
    if max(SignalPower)==SignalPower(i)
        disp(i);
    end
end
figure(2)
plot(SignalPower);
```