

6. ノイズのパワースペクトラム

次にノイズの場合について同様に計算してみます。MaTXを使います。

正規乱数は平均値が0, σ (標準偏差) が1なので、これをそのままノイズとして扱えば平均パワーが 1Vrms^2 のホワイトノイズになります。

ノイズの場合振幅がランダムで、短い期間で区切ると 区間毎にバラツキを生じ、最初の N 個のサンプルと次の N 個のサンプルで結果にかなり差が出ます。FFT アナライザでは多数回測定し平均して表示しています。そこで、ここでは N を 1024 に増やし 16 回のパワースペクトラムの平均を計算します。

```
Fs=6400.0; // sampling frequency
N=1024; //1回の FFT のサンプル数
M=16*N; //FFT を 16 回繰り返して平均を求めるため
Freq_step=Fs/N;
t=[0:M]/Fs; //時刻列(配列)
tm=t(1:M);
randn("seed",1);
xn=randn(t); //M 個のガウシアンノイズ(平均パワー1Vrms^2)
//平均パワーを確認
Pgaus=sum(xn(1:M)*xn(1:M))/M
```

Pgaus = 0.999327 でした。ノイズパワーは確かに 1Vrms^2 です。

```
q=[0:M/2];
fm=Fs/M*q;
/*乱数で作ったノイズをそのままフィルタを通さずに使っているので、
矩形窓でもよいが、ENBW_cfなどを考慮に入れた計算を試すため
ここでは Hanning 窓を使う。*/
k=[1:N];
w=0.5*(1-cos(2*PI*k/(N+1))); //Hanning 窓
sum_sq_w=sum(w*w);
sq_sum_w=(sum(w))^2;
Amp_cf=sum(w)/N;
ENBW_cf=sum_sq_w/sq_sum_w*N
```

補正係数は $N=1024$ なので、 $N=256$ の場合とわずかに異なります。

```
Amp_cf = 0.500488
ENBW_cf = 1.49854
```

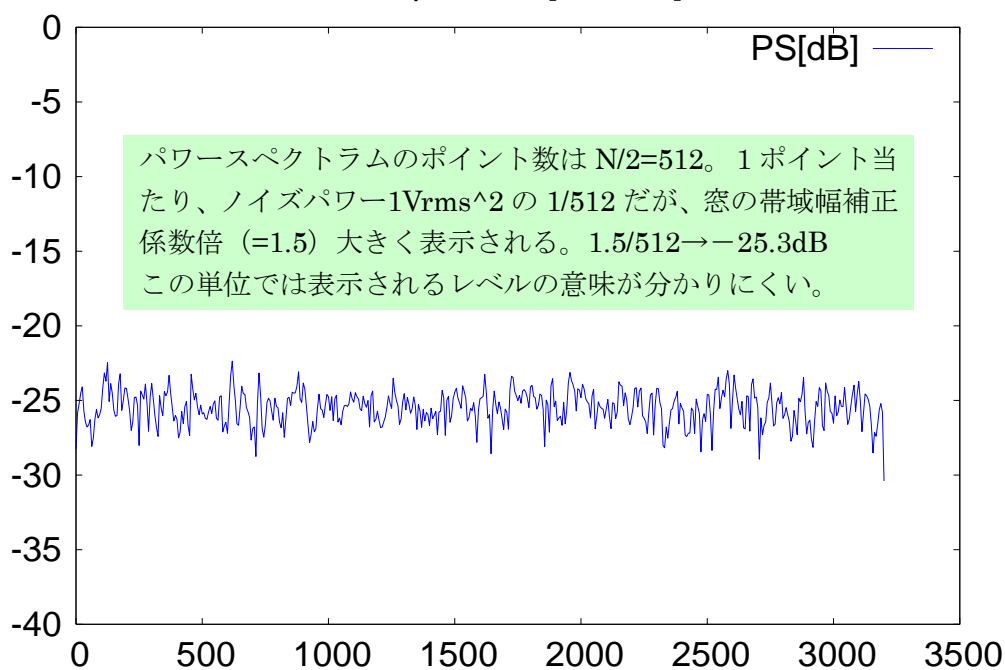
```
AvPxx=(abs(fft(w*xn(1:N)))^2+
abs(fft(w*xn(N+1:2*N)))^2+
abs(fft(w*xn(2*N+1:3*N)))^2+
abs(fft(w*xn(3*N+1:4*N)))^2+
```

```

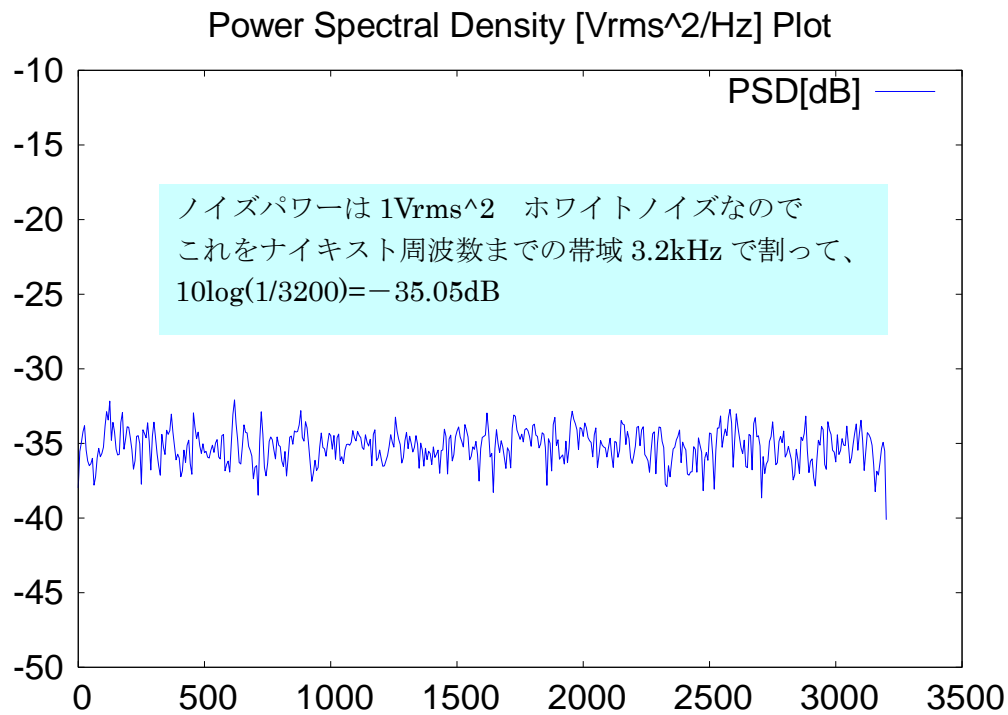
abs(fft(w*xn(4*N+1:5*N)))^2+
abs(fft(w*xn(5*N+1:6*N)))^2+
abs(fft(w*xn(6*N+1:7*N)))^2+
abs(fft(w*xn(7*N+1:8*N)))^2+
abs(fft(w*xn(8*N+1:9*N)))^2+
abs(fft(w*xn(9*N+1:10*N)))^2+
abs(fft(w*xn(10*N+1:11*N)))^2+
abs(fft(w*xn(11*N+1:12*N)))^2+
abs(fft(w*xn(12*N+1:13*N)))^2+
abs(fft(w*xn(13*N+1:14*N)))^2+
abs(fft(w*xn(14*N+1:15*N)))^2+
abs(fft(w*xn(15*N+1:16*N)))^2)/(N^2*16)/Amp_cf^2;
f=Freq_step*[0:N-1];
PS=2*AvPxx(1:N/2+1);
PS(1)=AvPxx(1);
PS(N/2+1)=AvPxx(N/2+1);
/*PS は FFT アナライザの表示に対応した片側パワースペクトル
単位は Vrms^2
ノイズパワー1Vrms^2 が N/2 個の周波数サンプルに分散、ただし
等価帯域幅補正係数分 (=1.5) 大きく表示される
1.5/(N/2)=1.5/512→-25.3dB
FFT のサンプル数によってレベルが変化する。*/
mgplot_reset(1);
mgplot_yrange(1, -40.0, 0.0);
mgplot_title(1,"Power Spectrum [Vrms^2] Plot");
mgplot(1,f(1:N/2+1),10*log10(PS),{"PS[dB]"});

```

Power Spectrum [Vrms²] Plot



```
/*PS を等価帯域幅=周波数刻み*ENBW_cf で割れば、 $V_{rms}^2/Hz$  単位になる。  
ノイズの場合は線スペクトルではないので  $V_{rms}^2/Hz$  で扱う方が合理的 */  
PSD=PS/(Freq_step*ENBW_cf);  
mgplot_reset(2);  
mgplot_yrange(2, -50.0, -10.0);  
mgplot_title(2,"Power Spectral Density [ $V_{rms}^2/Hz$ ] Plot");  
mgplot(2,f(1:N/2+1),10*log10(PSD),{"PSD[dB]"})
```



```
AvPps=sum(AvPxx)/ENBW_cf
Pps=sum(PS)/ENBW_cf
Ppsd=sum(PSD)*Freq_step
```

AvPps、Pps、Ppsd は下記のようにいずれもノイズパワーになっています。

$$\text{AvPps} = 0.998029$$

$$\text{Pps} = 0.998029$$

$$\text{Ppsd} = 0.998029$$

先に求めた P_{gaus} は窓関数を掛ける前の時間軸上のノイズデータから求めたパワー値ですが、P_{gaus} = 0.999327 と、誤差はわずかです。