

# Zpracování signálů a obrazů

## 3. Korelační metody



*Zimní semestr 2002/2003*

# 3. Korelační metody

*Korelační metody, zvyšování poměru signál/šum*

## ZADÁNÍ

Účelem cvičení je demonstrovat možnosti potlačení šumů (zvýšení poměru signál/šum) pomocí korelačních metod (korelační filtrace) a pomocí frekvenční filtrace.

Dále je úkolem cvičení seznámit se prakticky s principem zjišťování výkonové spektrální hustoty Welchovou metodou (metodou průměrování modifikovaných periodogramů).

## ÚKOL MEŘENÍ

1. Zadejte parametry pro 1. signál: (sinusový signál  $U_1=1.\sin(2.\pi.f.t)$ ,

amplituda=1V,  
frekvence=1Hz,  
fáze=0°.

K 1. signálu přidejte aditivní šum s rovnoměrným rozložením.

Dále zadejte parametry pro 2. signál: (sinusový signál  $U_2=2.\sin(2.\pi.f.t+90)$ ,

amplituda=1V,  
frekvence=1Hz,  
fáze=90°.

K 2. signálu přidejte aditivní šum s normálním rozložením. Počet bodů pro oba signály volte  $N=500$ .

Vypočtete autokorelační funkci pro 1.signál (AKF1), autokorelační funkci pro 2.signál (AKF2), vzájemnou korelační funkci mezi 1. a 2. signálem (KKF1-2) a vzájemnou korelační funkci mezi 2. a 1. signálem (KKF2-1). Vyzkoušejte si zobrazení vychýleného (koeficient  $1/N$ ) i nevychýleného odhadu (koeficient  $1/N-|m|$ ) autokorelačních funkcí výkonových signálů.

Naměřené signály si přibližně obkreslete do sešitu. Zdůvodněte, proč je v korelačních funkcích potlačen šum a porovnejte oba šумы z hlediska tohoto potlačení. Zjistěte, jak velká je efektivní hodnota zašuměného signálu a určete efektivní hodnotu samotného šumu v obou případech. (Ke zjištění velikosti

šumu použijte hodnoty autokorelační funkce pro nulový časový posuv, pro kterou platí

$$R_{xx}(0) = P_{\text{sig}} + P_{\text{noise}} .$$

Protože jde o prakticky bílý šum, je odpovídající špička v počátku autokorelační funkce velmi úzká.)

2. Filtrujte harmonický signál s parametry:

amplituda=1V,  
frekvence=10Hz,  
N=1024 bodů,  
fvzork=1000Hz,

ke kterému jste přidali šum s normálním rozložením,

- a) pomocí programu pro číslicové filtry typu FIR z minulého cvičení.  
dolní propust, řád=100, fzlom=20Hz.
- b) pomocí programu pro číslicové filtry typu IIR z minulého cvičení.  
dolní propust, typ Butterworth, řád=3, fzlom=20Hz.

Průběhy před i po filtraci obkreslete přibližně do sešitu. Zdůvodněte, proč je ve vyfiltrovaném signálu potlačen šum a proč kolísá amplituda vyfiltrovaného signálu.

3. Prohlédněte si zdrojové soubory ACF.M a CCF.M pro výpočet korelačních funkcí a vyzkoušejte program pro demonstraci šumů SUMY.M. Obkreslete do sešitu histogramy obou šumů a porovnejte je s příslušným ideálním rozložením hustoty pravděpodobnosti pro daný typ šumu. Dále posuďte zda se jedná o bílý šum.
4. Dále uvedeným postupem se seznámte s výpočtem výkonové spektrální hustoty signálů Welchovou metodou, která je připravena v programu MATLAB.

## POKYNY

Program pro výpočet korelací spustíte příkazem COR\_CZ (česká verze) (případně COR\_AN (anglická verze) ) v příkazovém okně Matlab Command Window. Programy pro filtraci spustíte příkazy FIR\_CZ (resp. FIR\_AN) a IIR\_CZ (resp. IIR\_AN).

Pro usnadnění práce lze spustit současně s Matlabem libovolný textový editor (např. MS Word). Nemusíte pak pracně obkreslovat obrázky změřených závislostí, ale pomocí příkazu Edit-Copy (Alt+PrintScreen) uložit aktivní obrázek do schránky a vyvolat jej pomocí příkazu Paste (Vlož - Shift+Insert) ve zvoleném textovém editoru. Ty je možno pak upravit a vytisknout. Přepínání mezi aplikacemi (Matlab, Word, atd.) se provádí stiskem kláves Alt+Tab. Zájemci mohou uložit hodnoty výpočtu do specifikovaného souboru do adresáře F:\.

Ovládání vlastního programu je jednoduché. Kliknutím myši na zvoleném tlačítku nastavíte daný parametr nebo aktivujete danou operaci. Spočtení korelace nebo filtrace dle nastavených parametrů provedete kliknutím na tlačítko Start (Run). Zvětšení (zoom) provedete označením bloku v grafu. Zvětšení zrušíte dvojitém kliknutím kdekoli v grafu. Další vysvětlení funkce tlačítek lze nalézt pod tlačítkem Info.

**Poznámka:** Základní příkazy MATLABu z oblasti **statistického zpracování signálů** zahrnují výpočet korelačních a kovariančních funkcí a výpočet výkonové spektrální hustoty. K tomu se používají následující příkazy:

<b>xcorr(x,y)</b>	vypočítá odhad <i>energetické</i> vzájemné <i>korelační</i> funkce signálů $x(n)$ a $y(n)$ ; při výpočtu se využívá FFT, $x(n)$ a $y(n)$ jsou vektory délky $M$ .
<b>xcov(x,y)</b>	vypočítá odhad <i>energetické</i> vzájemné <i>kovarianční</i> funkce signálů $x(n)$ a $y(n)$ ; při výpočtu se využívá FFT, $x(n)$ a $y(n)$ jsou vektory délky $M$ .
<b>xcorr(x,y,'unbiased')</b>	vypočítá nevychýlený odhad <i>výkonové</i> vzájemné <i>korelační</i> funkce signálů $x(n)$ a $y(n)$ ; (suma ve výpočtu korelační funkce je dělena $M- m $ ); při výpočtu se využívá FFT, $x(n)$ a $y(n)$ jsou vektory délky
<b>xcorr(x,y,'biased')</b>	vypočítá vychýlený odhad <i>výkonové</i> vzájemné <i>korelační</i> funkce signálů $x(n)$ a $y(n)$ (suma ve výpočtu korelační funkce je dělena $M$ ); při výpočtu se využívá FFT, $x(n)$ a $y(n)$ jsou vektory délky;
<b>xcorr(x,y,'coeff')</b>	počítá normovanou korelační funkci (pro autokorelační funkci je první hodnota rovna 1)
<b>psd(x)</b>	vypočítá výkonovou spektrální hustotu signálu $x(n)$ ; program umožňuje několik variant tohoto příkazu;
<b>csd(x,y)</b>	vypočítá vzájemnou výkonovou spektrální hustotu signálů $x(n)$ a $y(n)$ ; program umožňuje několik variant tohoto příkazu;
<b>tfe(x,y)</b>	vypočítává odhad přenosové funkce soustavy, na jejímž vstupu je signál $x(n)$ a na jejímž výstupu je signál $y(n)$ . Výpočet je proveden jako podíl vzájemného a vlastního výkonového spektra. Program umožňuje několik variant tohoto příkazu;
<b>cohere(x,y)</b>	počítá kvadrát absolutní hodnoty koherenční funkce signálů $x(n)$ a $y(n)$ .

Postup k demonstraci *Welchovy metody pro výpočet výkonové spektrální hustoty signálů*:

Vytvořte 1001-prvkový signál  $xn$ , tvořený dvěma sinusovkami a šumem:

**Fs = 1000;**                      **% vzorkovací frekvence**

```
t = 0:1/Fs:1; % jedna sekunda vzorkovaného signálu  
xn = sin(2*pi*50*t) + 2*sin(2*pi*120*t) + randn(size(t));
```

Hrubý odhad PSD (Power Spectral Density = výkonové spektrální hustota) signálu xn je:

```
Pxx = abs(fft(xn,1024)).^2/1001)
```

Tento odhad se nazývá **periodogram**. Jde o nekonzistentní odhad PSD - je asymptoticky nevychýlený, ale s s růstem počtu bodů se nezmenšuje jeho disperse - nevyhlazuje se

To si můžete demonstrovat následujícími příkazy v MATLABu:

```
Pxx_short = abs(fft(xn,256)).^2/256;  
plot((0:255)/256*Fs, 10*log10(Pxx_short))  
plot((0:1023)/1024*Fs, 10*log10(Pxx))
```

(Pozn.: Při psaní řádků v MATLABu podobných řádku předchozímu lze s výhodou získat opakovaný předchozí řádek pomocí klávesnicové šipky vzhůru. Po požadované úpravě tohoto řádku ho uložte příkazem ENTER. Potřebujete - li psát další podobné řádky - příklad viz níže - opakujte tento postup.)

Disperzi odhadu PSD je možno zmenšit rozdělením signálu do nepřekrývajících se sekcí a zprůměrováním odpovídajících periodogramů těchto sekcí:

```
Pxx = ( abs(fft(xn( 1:256))).^2 + ...  
abs(fft(xn( 257:512))).^2 + ...  
abs(fft(xn( 513:768))).^2 ) / (256*3);  
plot((0:255)/256*Fs,10*log10(Pxx))
```

Tento odhad má dispersi 1/3 disperse periodogramu o délce 256 ukázaného výše. Zvětšením počtu sekcí se disperse zmenší. Většího počtu sekcí lze dosáhnout překrytím jednotlivých sekcí:

```
Pxx = ( abs(fft(xn( 1:256))).^2 + ...  
abs(fft(xn(129:384))).^2 + ...  
abs(fft(xn(257:512))).^2 + ...  
abs(fft(xn(385:640))).^2 + ...  
abs(fft(xn(513:768))).^2 + ...  
abs(fft(xn(641:896))).^2 ) / (256*6);  
plot((0:255)/256*Fs,10*log10(Pxx))
```

V tomto případě jsou sekce statisticky závislé, což vede k vyšší disperse; je tedy třeba najít kompromis mezi počtem sekcí a jejich překrytím.

Jiná možnost je použít *modifikovaný periodogram*, kdy se na jednotlivé sekce signálu aplikuje vhodné okno. To redukuje statistickou závislost sekcí působenou překrytím, protože okna klesají spojitě k nule na obou koncích. Jiné než obdélníkové okno také zmenšuje vliv postranních laloků (spektrální leakage), ale zvětšuje šířku špiček ve spektru. Při užití vhodných oken (Hamming, von Hann, Kaiser) se disperse podstatně snižuje i při překrytí 50%.

Použití okna von Hann („hanning“) se provede následovně:

```
w = hanning(256)';  
Pxx = ( abs(fft(w.*xn( 1:256))).^2 + ...  
        abs(fft(w.*xn(129:384))).^2 + ...  
        abs(fft(w.*xn(257:512))).^2 + ...  
        abs(fft(w.*xn(385:640))).^2 + ...  
        abs(fft(w.*xn(513:768))).^2 + ...  
        abs(fft(w.*xn(641:896))).^2 ) / (norm(w)^2*6);  
plot((0:255)/256*Fs,10*log10(Pxx))
```

Všimněte si, že se spektrální špičky rozšířily a šumový práh se vyhladil.

Uvedená metoda průměrování modifikovaných periodogramů se podle svého autora nazývá metoda Welchova. V MATLABU je implementována ve funkcích **psd** a **csd**, které dovolují volbu délky FFT, okna a počtu bodů překrytí.

Základní příkaz:

```
Pxx = psd(x)
```

předpokládá délku FFT 256, okno von Hann (hanning), nulové překrytí sekcí a vyjmutí lineárního trendu. Je-li signál  $x(n)$  ve V, je Pxx ve  $V^2/Hz$ .

Poslední výše uvedený příklad lze zapsat jednodušeji (při specifikování překrytí sekcí ve 128 bodech a vyřazení odstranění trendu) takto:

```
nfft = 256;           %délka fft  
Fs = 1000;          % vzorkovací frekvence  
window = hanning(256); %volba okna  
noverlap = 128;      %počet bodů překrytí v sekcích  
dflag = 'none';      %neodstraněn trend  
Pxx = psd(xn,nfft,Fs>window,noverlap,dflag);
```

Pořadí argumentů funkce psd je důležité (s výjimkou řetězce 'dflag', který může být kdekoliv). Vzorkovací frekvence neovlivní odhad PSD, pouze definuje frekvenční osu pro kreslení. psd generuje obrázek:

```
psd(xn,nfft,Fs>window,noverlap,dflag)
```

Chcete-li kreslit PSD sami, získáte vektor frekvencí pomocí druhého argumentu:

```
[Pxx,f] = psd(xn,nfft,Fs,256,noverlap,dflag);  
plot(f,10*log10(pxx)), grid
```

Protože signál xn je reálný, vrací psd pouze frekvence od 0 do  $F_s/2$  (Nyquistova frekvence). Na rozdíl od toho výpočet PSD pomocí FFT výše dával grafy od 0 do  $F_s$ .

Chceme-li dostat správně i amplitudové hodnoty PSD, musíme hodnoty vypočtené pomocí funkce psd ještě upravit multiplikativní konstantou  $\text{norm}(w)^2 / \sum(w)^2$ , kde w je vektor použitého okna. Tento činitel nezávisí na délce okna a ve výše uvedeném tvaru ani na typu okna. Lze tedy např. napsat:

```

w1=hanning(256); w2=hanning(500);
[Pxx1,f1]=psd(xn,256,Fs,w1,128,'none');
[Pxx2,f2]=psd(xn,1024,Fs,w2,250,'none');
plot(f1,10*log10(Pxx1*norm(w1)^2/sum(w1)^2))
plot(f2,10*log10(Pxx2*norm(w2)^2/sum(w2)^2))

```

V obou takto získaných obrázcích je odstup spektrálních špiček 6 dB, vyšší špička je 0 dB.

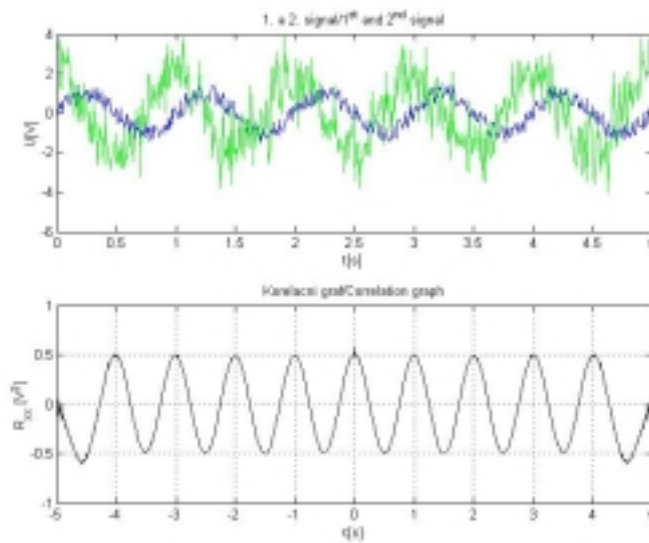
Signál na frekvenci 50 Hz má amplitudu 1 V. Jeho výkon ( na odporu  $1\Omega$ ) je tedy  $(1/\sqrt{2})^2 = 1/2$  W. Matlab počítá dvoustrannou PSD  $S_{xx}$ , takže hodnota výkonu má být poloviční, tedy  $1/4$ . Tomu odpovídá v dB hodnota výkonu  $10*\log_{10}(0.25) = -6\text{dB}$ . To je také v obrázku.

Signál na frekvenci 120 Hz má amplitudu 2 V. Odpovídající výkon je tedy  $(2/\sqrt{2})^2=2\text{W}$ . Špička v průběhu dvoustranné PSD bude poloviční, tedy 1W. Tomu odpovídá hodnota 0 dB, která je také v grafu.

Všimněte si také o něco nižšího šumového prahu v druhém obrázku. To odpovídá delšímu použitému oknu.

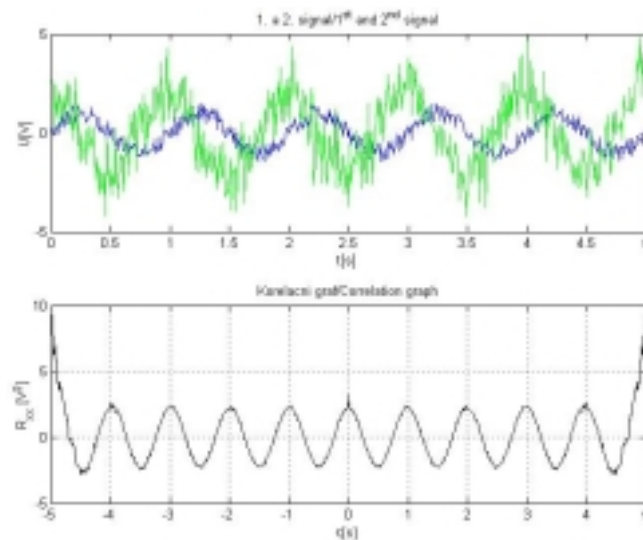
## 1. úkol měření

Autokorelační funkce pro 1.signál (AKF1):



SNR: 16dB

Autokorelační funkce pro 2.signál (AKF2):

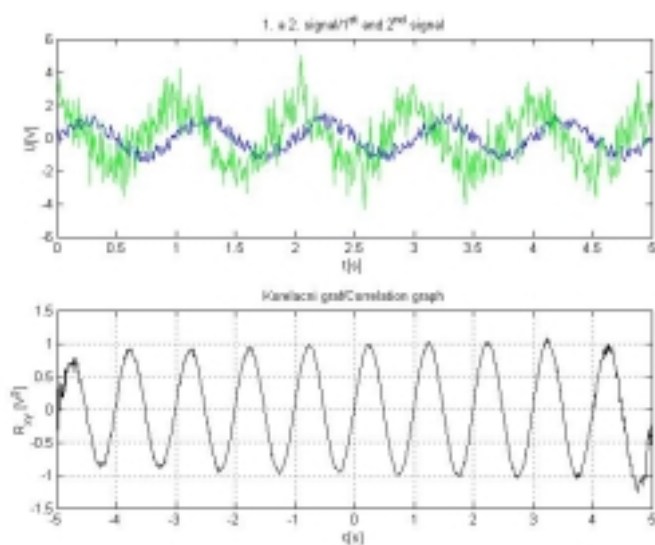


SNR: 6dB

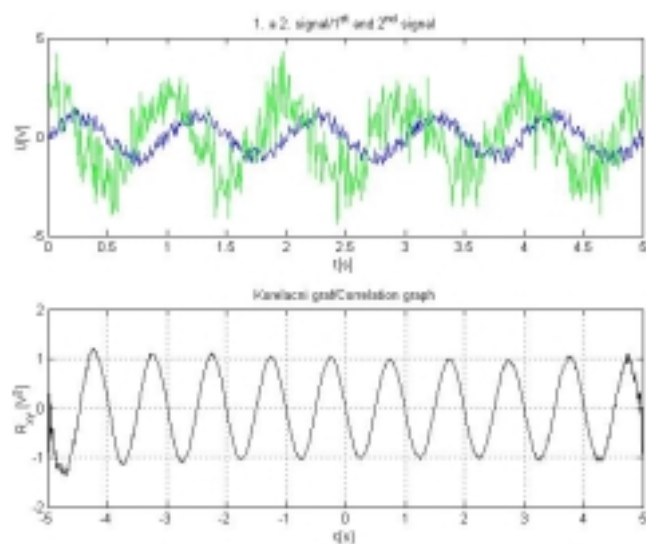
Při použití korelačních funkcí je potlačen šum, neboť korelační funkce porovnává jak moc se podobají (míru podobnosti) signály posunuté navzájem o  $\tau$  (časový úsek).



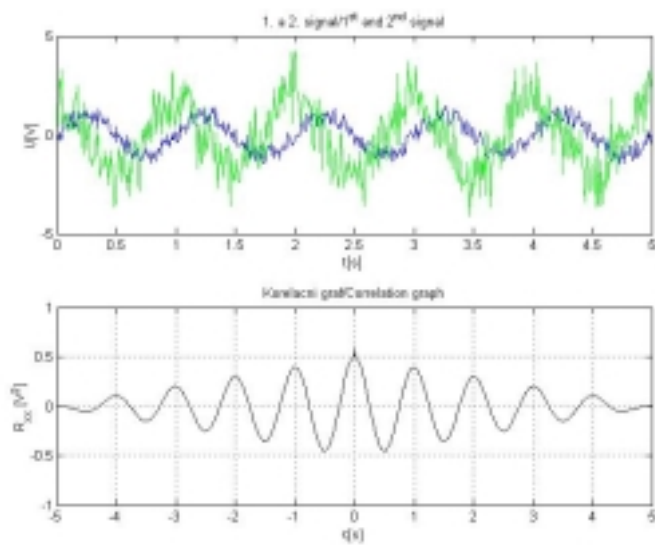
Vzájemná korelační funkci mezi 1. a 2. signálem (KKF1-2):



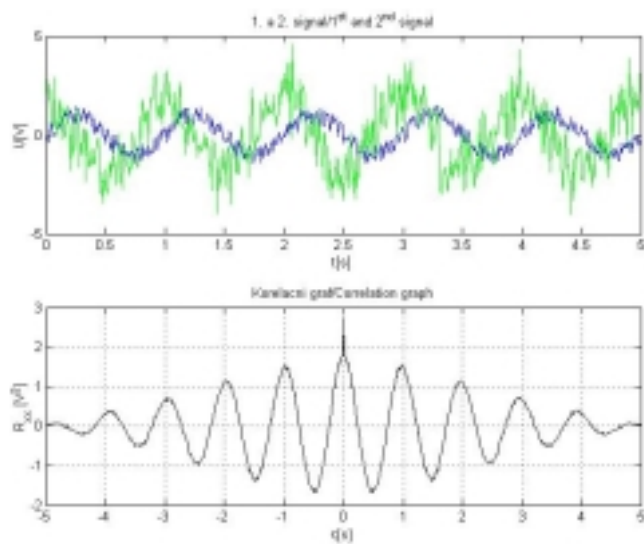
Vzájemná korelační funkci mezi 2. a 1. signálem (KKF2-1):



Autokorelační funkce pro 1.signál (AKF1) – vychýlený odhad:

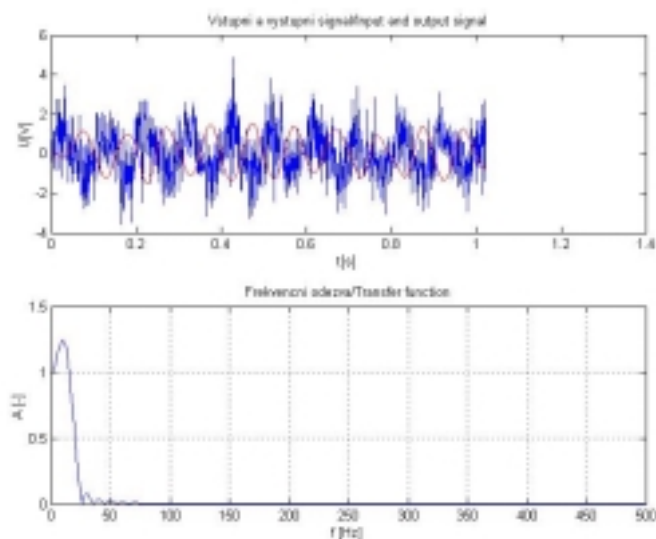


Autokorelační funkce pro 2.signál (AKF2) – vychýlený odhad:

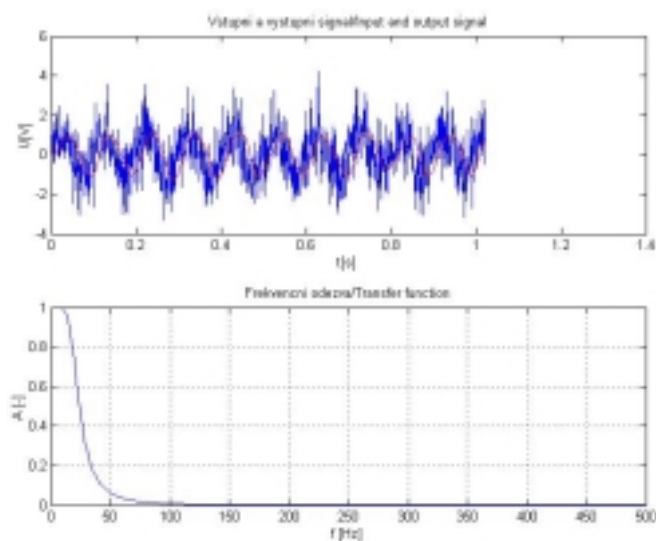


## 2. úkol měření

Filtrování harmonického signálu pomocí programu pro číslicové filtry typu FIR: dolní propust, řád=100, fzlom=20Hz:



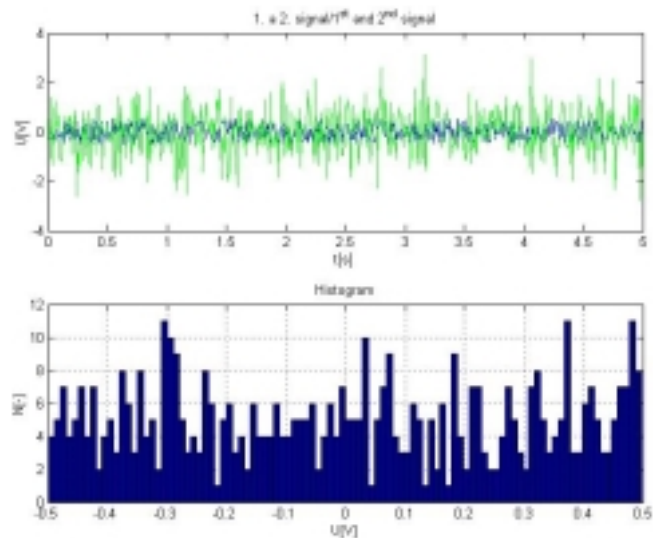
Filtrování harmonického signálu pomocí programu pro číslicové filtry typu IIR: dolní propust, typ Butterworth, řád=3, fzlom=20Hz:



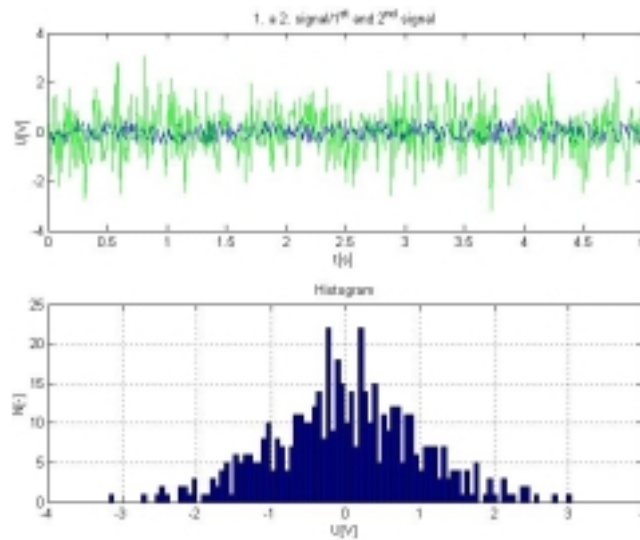
Protože je filtr dolní propust a šum je vysoká frekvence, nenachází se šum po odfiltrování – po vyhlazení má vliv jen na amplitudu.

### 3. úkol měření

Histogram rovnoměrného šumu (šum by měl mít zastoupeny všechny složky rovnoměrně v celém rozsahu):



Histogram normálního šumu (šum by měl být rozložen podle normálního rozložení v daném rozsahu):



### ZÁVĚR

Zjišťovali jsme možnosti potlačení šumů (zvýšení poměru signál/šum) pomocí korelačních metod (korelační filtrace) a pomocí frekvenční filtrace.