

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická

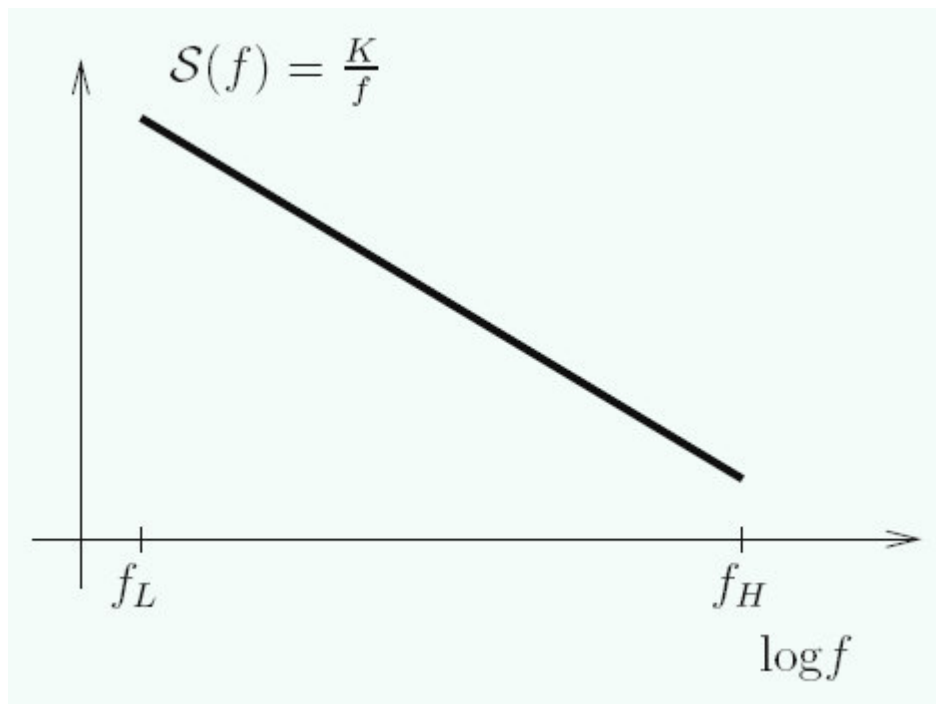
PROJEKT Č. 3

**Generování WSS procesu s
požadovanou korelační funkcí a
normálním rozdělením**

Vypracoval: Jan HLÍDEK
V rámci předmětu: Signály a soustavy (X37SGS)
Zpracováno do: 20. 12. 2007
Cvičení: čtvrtek sudý od 14:30 do 16:00

1. ZADÁNÍ

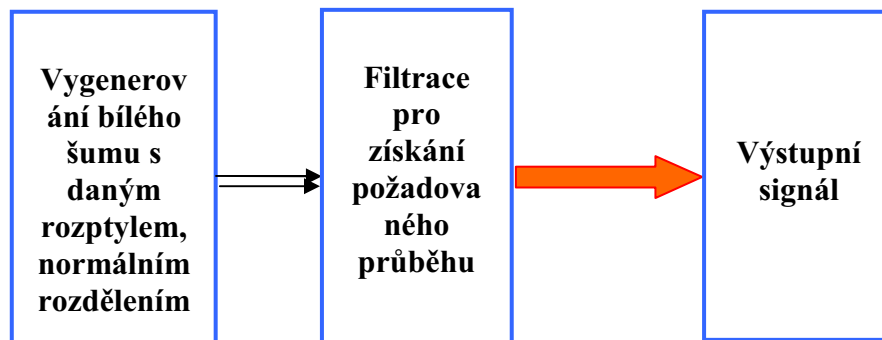
- Generujte WSS proces s normálním rozdělením, jehož spektrální výkonová hustota bude modelovat fázový šum oscilátoru typu f^{-1} (flicker phase).
- Parametry K a f_L volte podle vlastního uvážení, parametr f_H ztotožněte s hodnotou $f_{sa} / 2$.
- Zkoumejte vliv řádu filtru na přesnost aproximace.



- Do jednoho obrázku vykreslete PSD modelovaného procesu i ideální požadovaný tvar PSD.

2. POSTUP ŘEŠENÍ

Základním cílem úlohy je seznámení se s vygenerováním nějakého, svým způsobem charakteristického, signálu z bílého šumu. Stěžejní částí je návrh filtru.



Obr. 1 Rozpracování úlohy s naznačením jednotlivých fází

Pro snadné zorientování v tomto zpracování dané úlohy několik poznámek. Níže jsou uvedeny grafy důležitých průběhů a to přímo ve formě, kde jsou sledované průběhy dosti přiblíženy. Celkový náhled na jednotlivé grafy je k dispozici na konci tohoto zpracování – u zdrojového kódu. To umožňuje snadnou orientaci v kódu a přitom se obrázky dvakrát neopakují, což vede k přehlednosti.

Vysvětleme si nyní užívané značky. Dále je případně uvedeno, jaká hodnota byla zvolena pro tuto úlohu:

- f_{sa} - vzorkovací frekvence, zvoleno 2000 Hz
- f_L - volíme 120 Hz
- f_H - volíme jako $f_{sa} / 2 = 2000 / 2 = 1000$ Hz
- $f_{prech.}$ - přechodové pásmo filtru, zvoleno $f_{prech.} = 0,1 \cdot f_L = 12$ Hz
- řád f - zvoleny řády filtru 4 a 54
- t - čas [s]
- x - hodnoty vstupní veličiny v čase
- y - hodnoty výstupní veličiny v čase
- S - spektrální výkonová hustota (PSD), většinou ve [W/Hz]
- S_x - vstupní PSD
- S_y - výstupní PSD
- $|H|$ - amplitudová charakteristika filtru
- $|H|^2$ - zisk filtru
- var - rozptyl náhodné veličiny
- K - konstanta určující velikost výkonu, který na výstupu požadujeme, zvoleno $K = 60$

Nejdříve upozornění pro správnou interpretaci úlohy. Ze zadání je vidět, že frekvenční osa je uváděna jako logaritmická a proto je zmiňovaný průběh realizován přímkou. Pro řešení tohoto projektu jsem však raději zvolil lineární měřítko. Z toho tedy vyplývá, že potom požadovaný průběh výstupní PSD (K/f) je hyperbola. V základním měřítku jsou také vykreslovány oboustranné spektrální hustoty (viz Dodatek), v části Grafy pak většinou zvětšena jen strana pro kladné kmitočty.

Podívejme se tedy na způsob řešení. Snažím se každý odstavec věnovat jedné kapitole, tak jak je pak vygenerován okomentovaný zdrojový kód v dodatku ke konci této práce.

Na začátek jsou nadefinovány parametry. Úlohu budeme řešit tak, že si v podstatě navzorkujeme signál a k němu pak stanovujeme např. PSD. Protože to tedy bereme takto, budou výsledky na ose y u PSD vycházet po korekci $1/N_{sa}/f_{sa}$. Uvažujeme je také v lineárním měřítku, a to s jednotkami W/Hz . Tato jednotka se v souvislosti s PSD běžně používá, lze ověřit např. v úvodu zde: http://www.utko.feec.vutbr.cz/~silhavy/MVDP/index_soubory/7_MVDP.pdf. Zajímavostí je definování přechodového pásma filtru. Pracujeme totiž s jistou reálnou aproximací, a tak nemůžeme požadovat ideální průběh filtrace. Změnami velikosti tohoto pásma lze sledovat vliv na charakteristiku (samozřejmě u filtrů vyšších řádů). Např. při zúžení tohoto pásma na minimum, které dovoluje algoritmus Matlab, stoupá charakteristika velice strmě, zato však dosti překmitne směrem nahoru – Spektrální složky těsně za hranicí f_L jsou pak zdůrazněny více, než je požadováno.

Na vstup soustavy pouštíme bílý šum. Pomocí odmocniny z rozptylu mohou ovlivňovat velikost odchylky. Odmocnina je nutná, abychom se pohybovali ve stejných „jednotkách“ pro veličinu x . Pro kontrolu si vynášíme autokorelační funkci signálu na vstupu. Dostáváme v podstatě Diracův impuls v $\tau = 0$, z čehož plyne nekorelovanost. Do soustavy tak pouštíme opravdu nekorelovaný bílý šum.

Nyní vytvoříme ideální požadovanou spektrální výkonovou hustotu. Provedeme to jednoduše v cyklu a to tak, že jako nulové nastavíme hodnoty ležící mimo interval $\langle f_L, f_H \rangle$. Hodnoty uvnitř intervalu váhujeme dle jejich „vzdálenosti“ od f_L . Pro vykreslení je pak nutné pamatovat na to, že je nutno ještě použít funkci „fftshift“ pro správné zobrazení spektra.

Protože platí

$$S_y = |H(\Omega)|^2 \cdot S_x(\Omega),$$

tak pro zjištění amplitudové charakteristiky filtru vycházíme z následujícího vztahu:

$$|H(\Omega)| = \sqrt{\frac{S_y(\Omega)}{\sigma_x^2}}.$$

Je samozřejmě nutno uvažovat vztah pro naše znormování. Protože jsme však hodnotu S_y při vytváření nenormovali na Hz (což činíme až do grafů), lze tento vzorec využít. Hodnotu S_y tedy podělíme rozptylem (var_x) a odmocníme. Konkrétní realizace v Matlabu byla provedena jistými úpravami. Je nutno přenormovat osy a pro vhodné spočtení filtru také upravit počet hodnot. Musíme to udělat, protože doporučená metoda pro nalezení přenosové funkce filtru ze zadané spektrální hustoty, yulewalker, vyžaduje frekvence od 0 do 1 a to s rostoucí tendencí. Princip spočívá v rozdělení na propustné a nepropustné pásmo a jejich následné spojení do jednoho vektoru. Přenosová charakteristika je v nepropustném pásmu vynulována, pro propustné je cyklem vytvořena charakteristika dle zadání $S_y = \frac{K}{f}$ a výše zmíněného vztahu.

Dle konkrétního požadovaného stupně filtru vygenerujeme pomocí funkce „yulewalk“ koeficienty pro vytvoření filtru. Doděláváme zde ještě normování, tak aby skutečně frekvence vstupující do yulewalk byly mezi 0 a 1. Z ideální požadované přenosové funkce tak budu moci dále získat jistou reálně proveditelnou. Získáme ji díky funkci freqz(...). V grafech je pak zobrazeno porovnání mezi filtry jednotlivých řádů a ideálním průběhem.

Signál vyfiltrujeme na základě vytvořeného filtru – funkcí „filter“. Získáme tak časové průběhy signálu za filtrem, tedy na výstupu. Pro zajímavost porovnáme korelační funkce, kde je vidět, že se již nejedná o absolutně bílý šum s konstantní PSD, ale v korelační funkci se objevuje zvlnění a to tím více, čím vyšší je řád filtru.

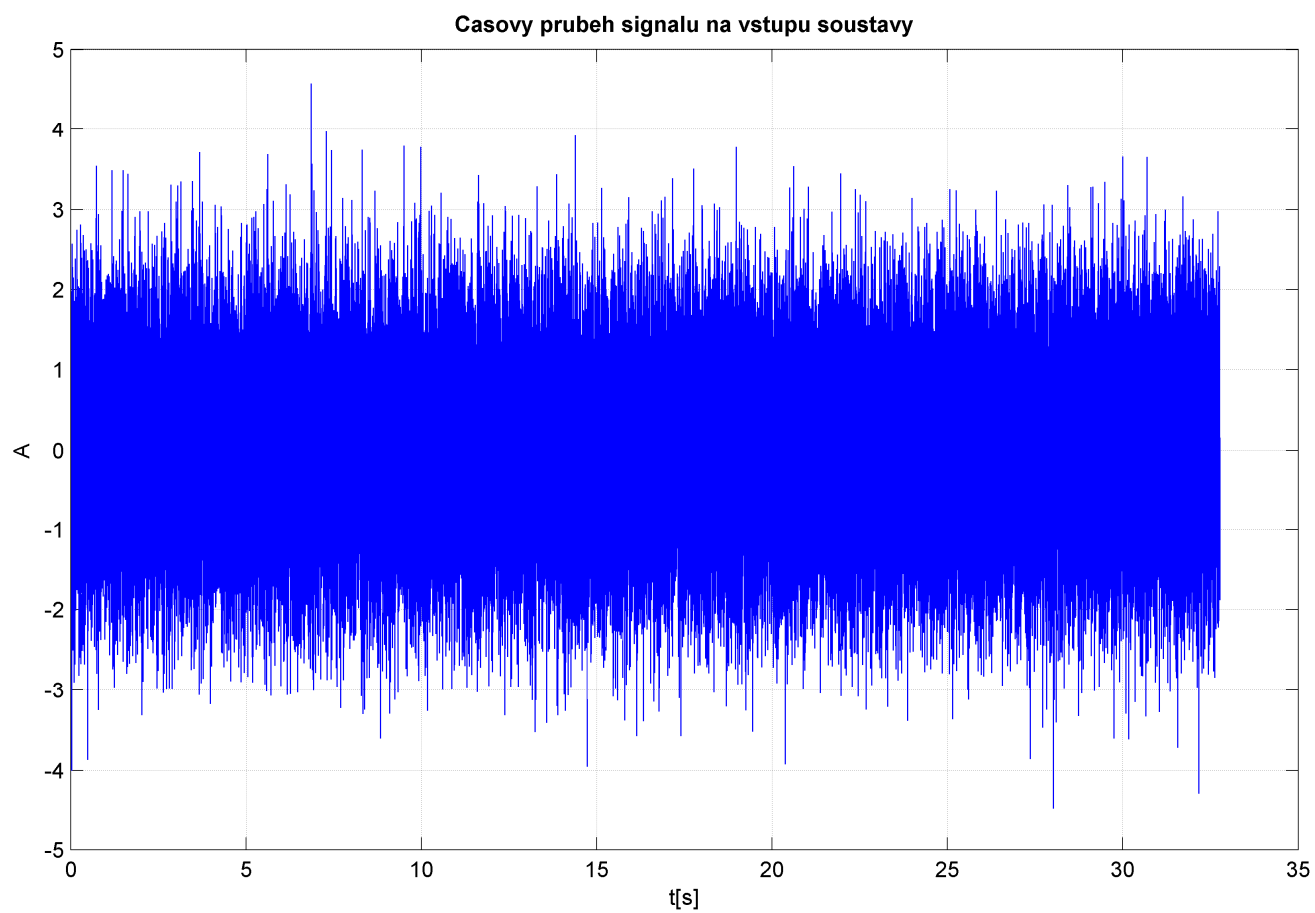
Spektrální výkonovou hustotu vypočteme standardně – schematicky takto:

$$S_{y1} = \frac{1}{konst.} |DFT[signal]|^2$$

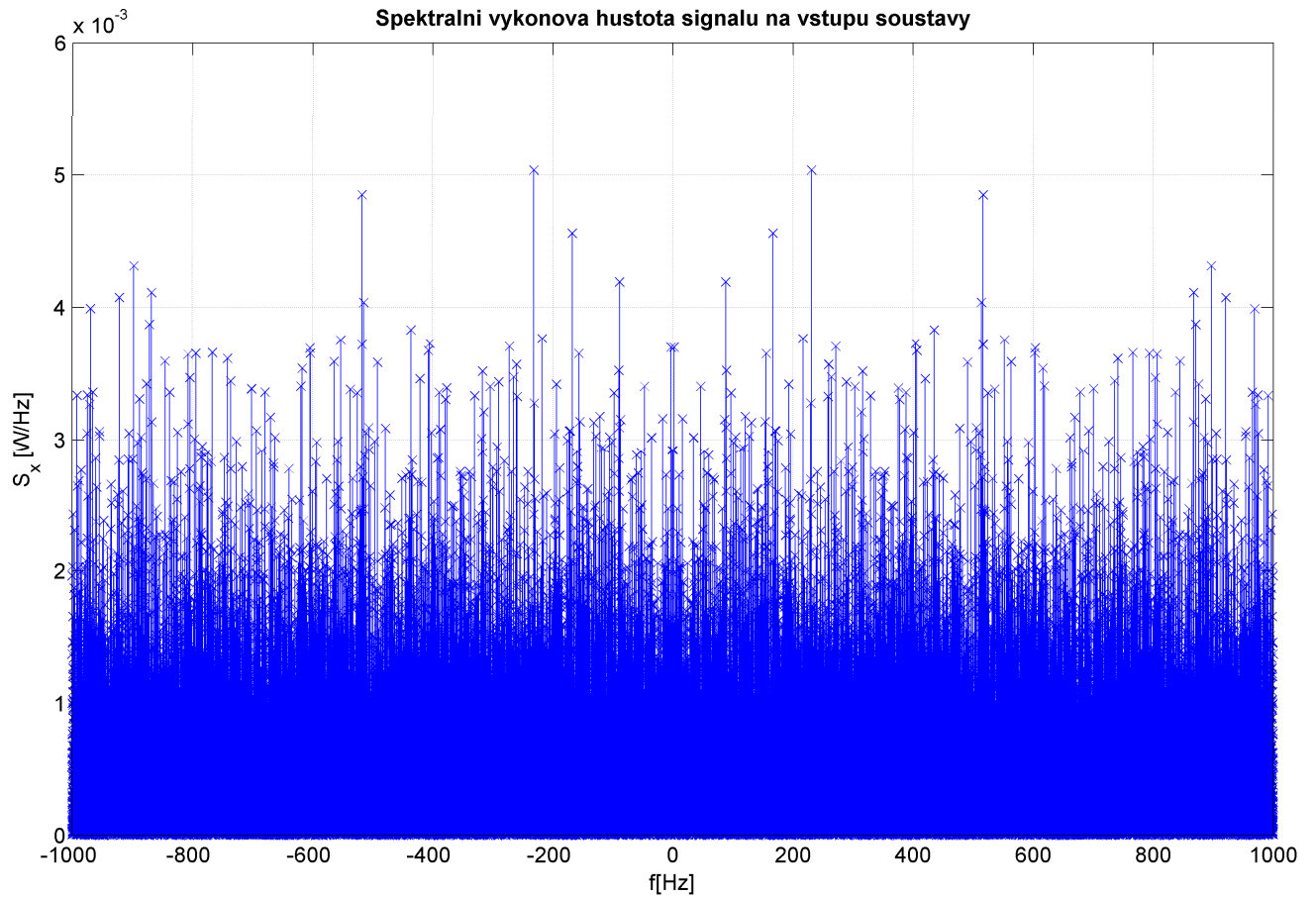
Pro ukázkou lze také např. vykreslit póly daného filtru atd.

Informace, které zde nezmiňuji, jsou ve většině případů vyjasněny v dodatku, kde je umístěn zdrojový kód programu i s komentáři, ilustračními obrázky a výpisy některých hodnot proměnných. Z technických důvodů mohou být poněkud odlišné od zobrazení v části grafy, protože je třeba pro vygenerování této podoby zdrojového kódu udělat novou realizaci šumu... (znovu spustit generování). Některé grafy jsou v tomto výpisu také uvedeny jen pro orientaci a pro zajímavost. Další informace a porovnání lze nalézt také v popiscích ke grafům a v samotných grafech.

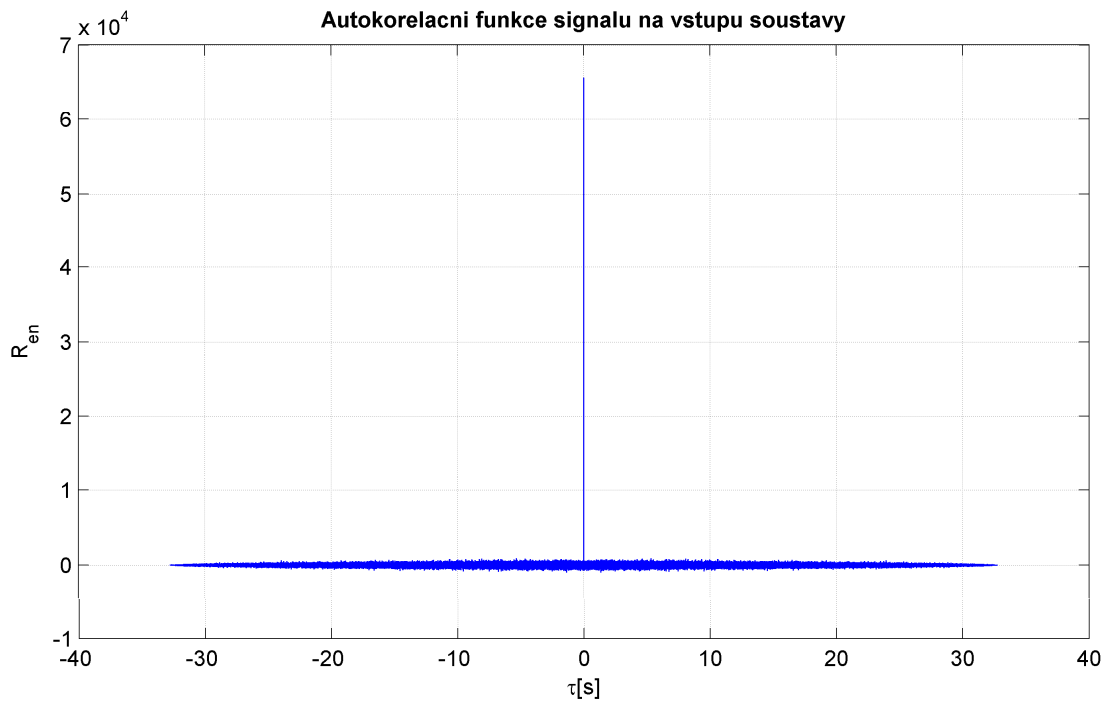
3. GRAFY



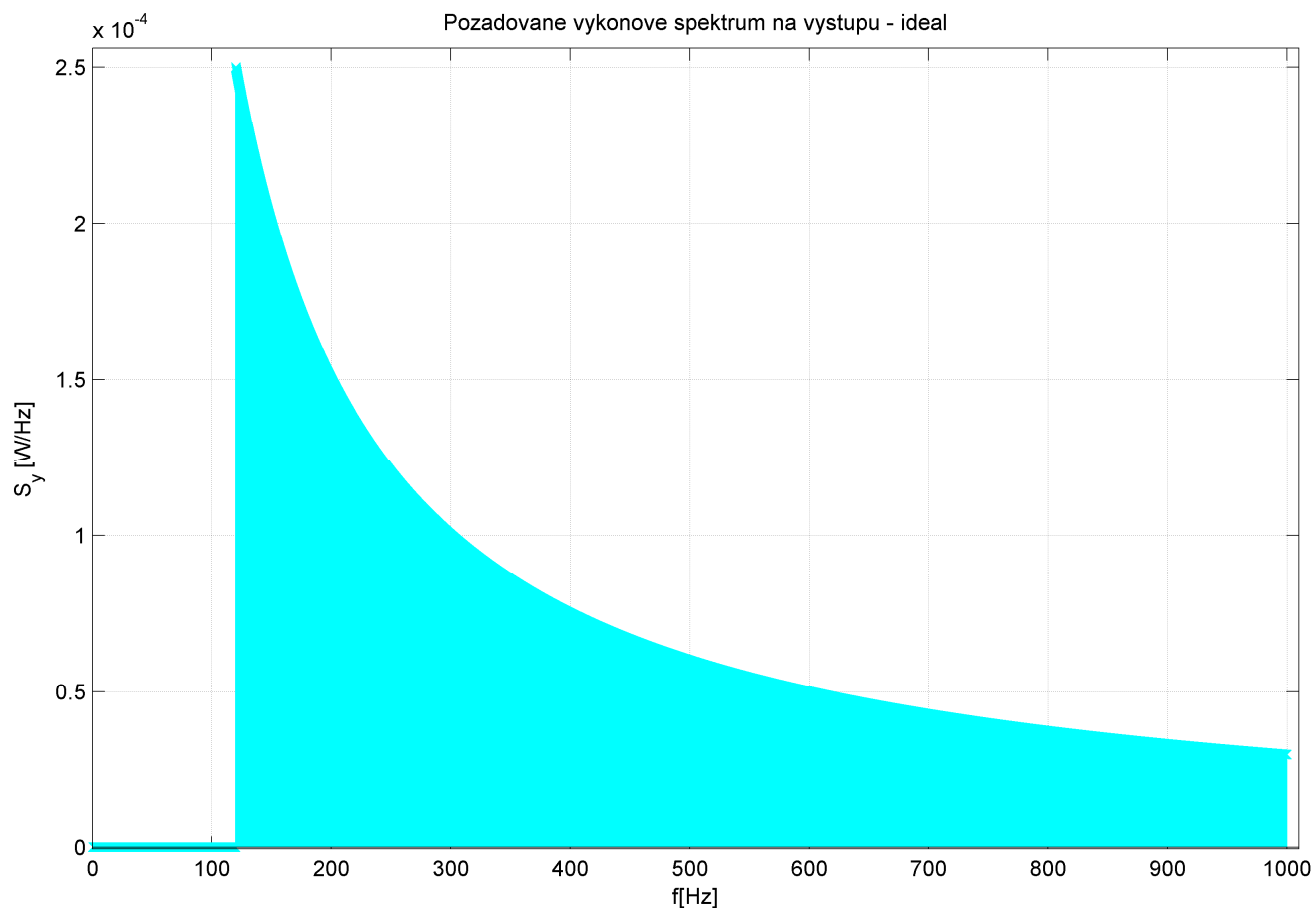
Obr. 2 Časový průběh signálu na vstupu soustavy (bílý šum s normálním rozdělením).



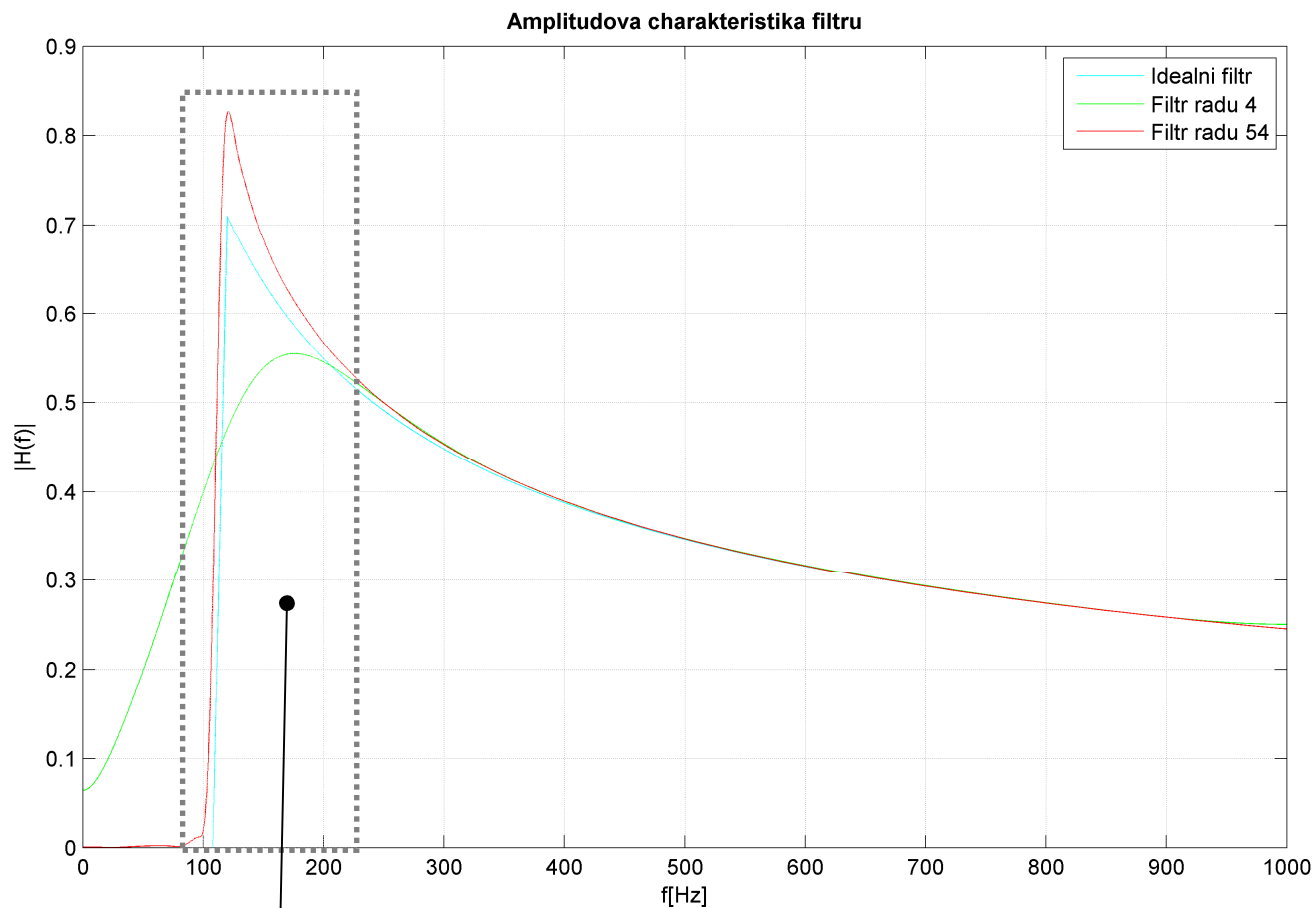
Obr. 3 PSD na vstupu soustavy – lze uvažovat jako přibližně konstantní spektrální hustotu.



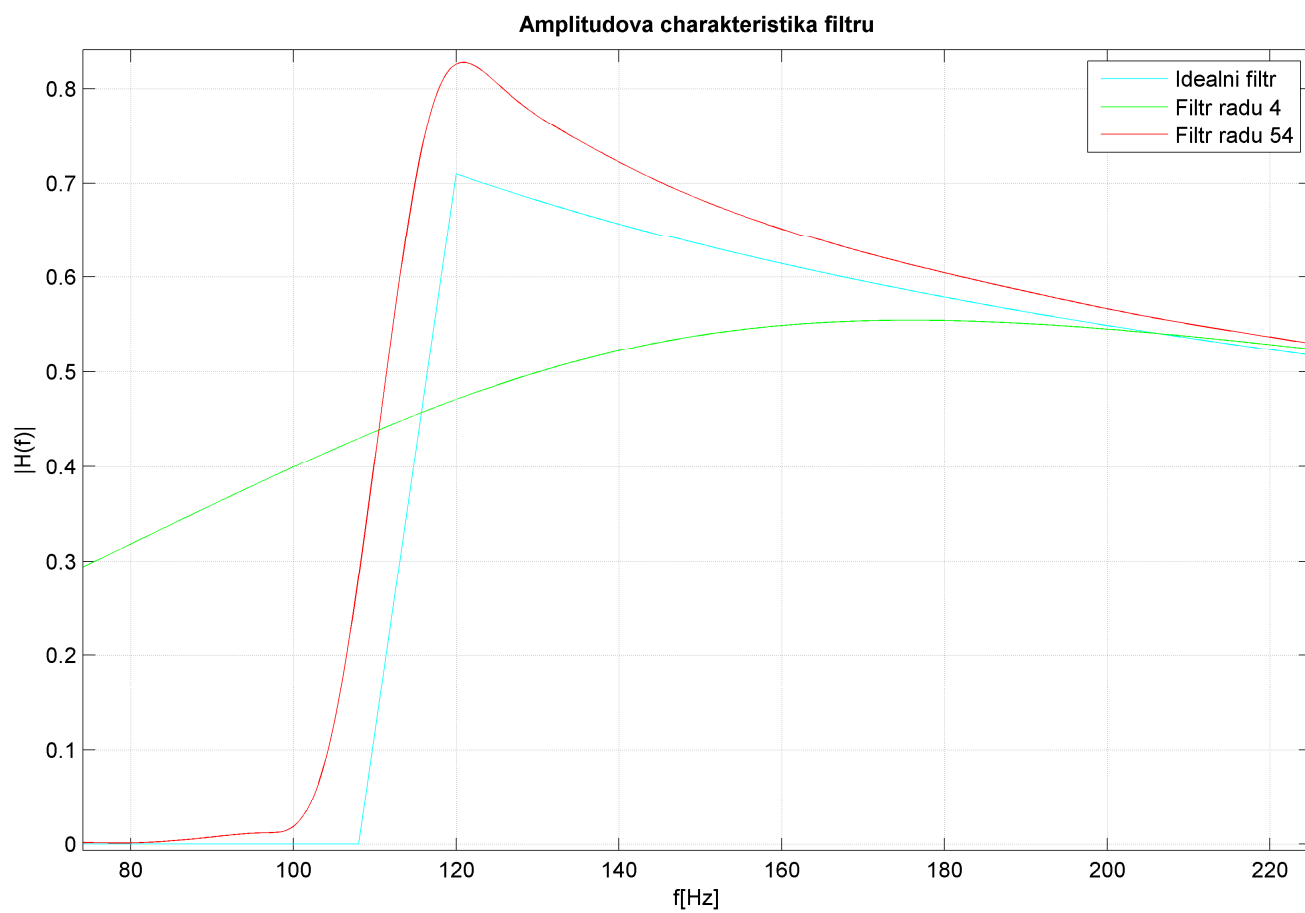
Obr. 4 Autokorelační funkce – kontrola vstupního signálu. Potvrzuje, že se skutečně jedná o bílý šum, doopravdy nekorelovaný a bez stejnosměrné složky. Normována jako energetická autokorelační funkce.



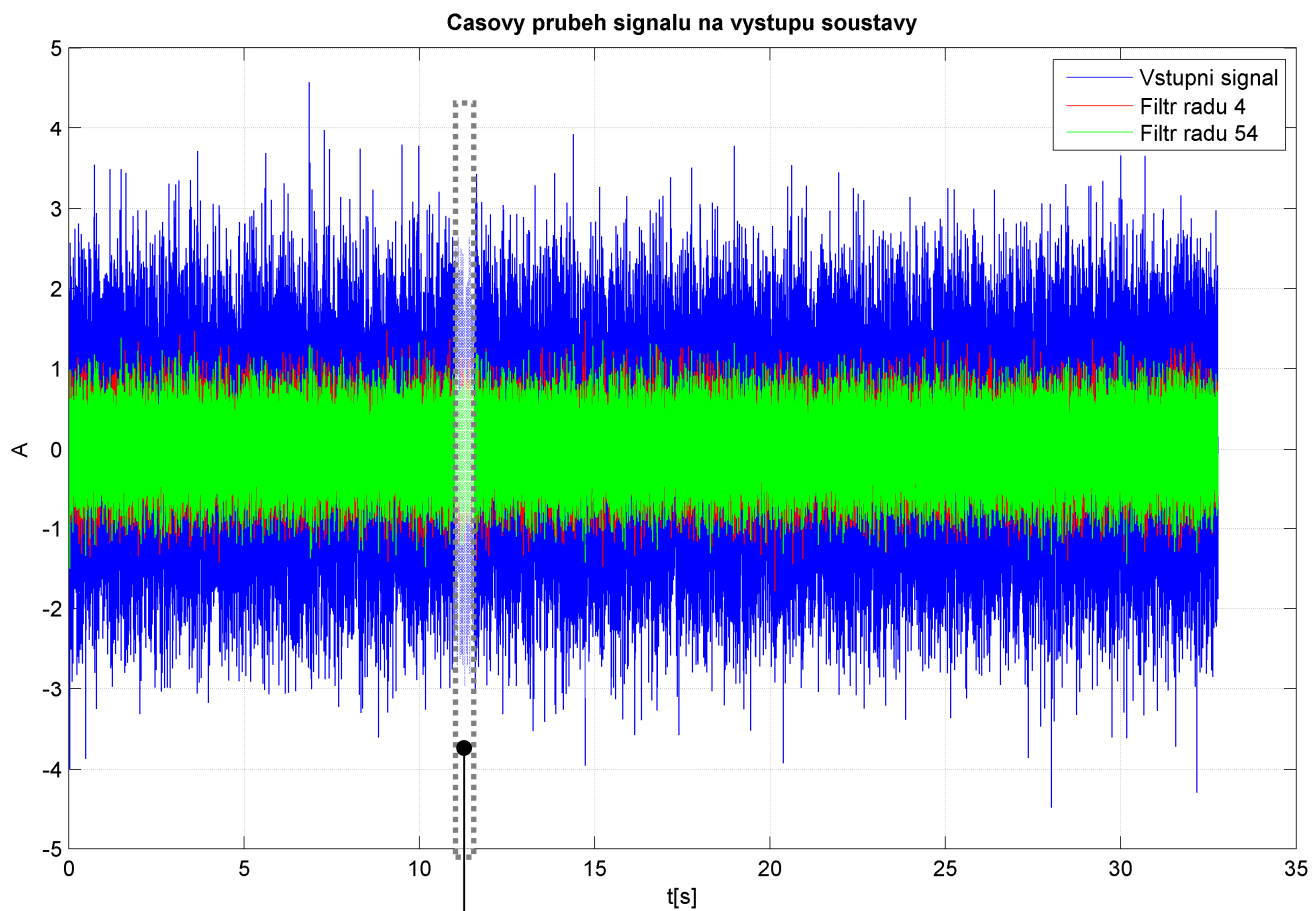
Obr. 5 Dle zadání požadujeme na výstupu tento průběh (uvažujeme-li lineární měřítko na ose f)



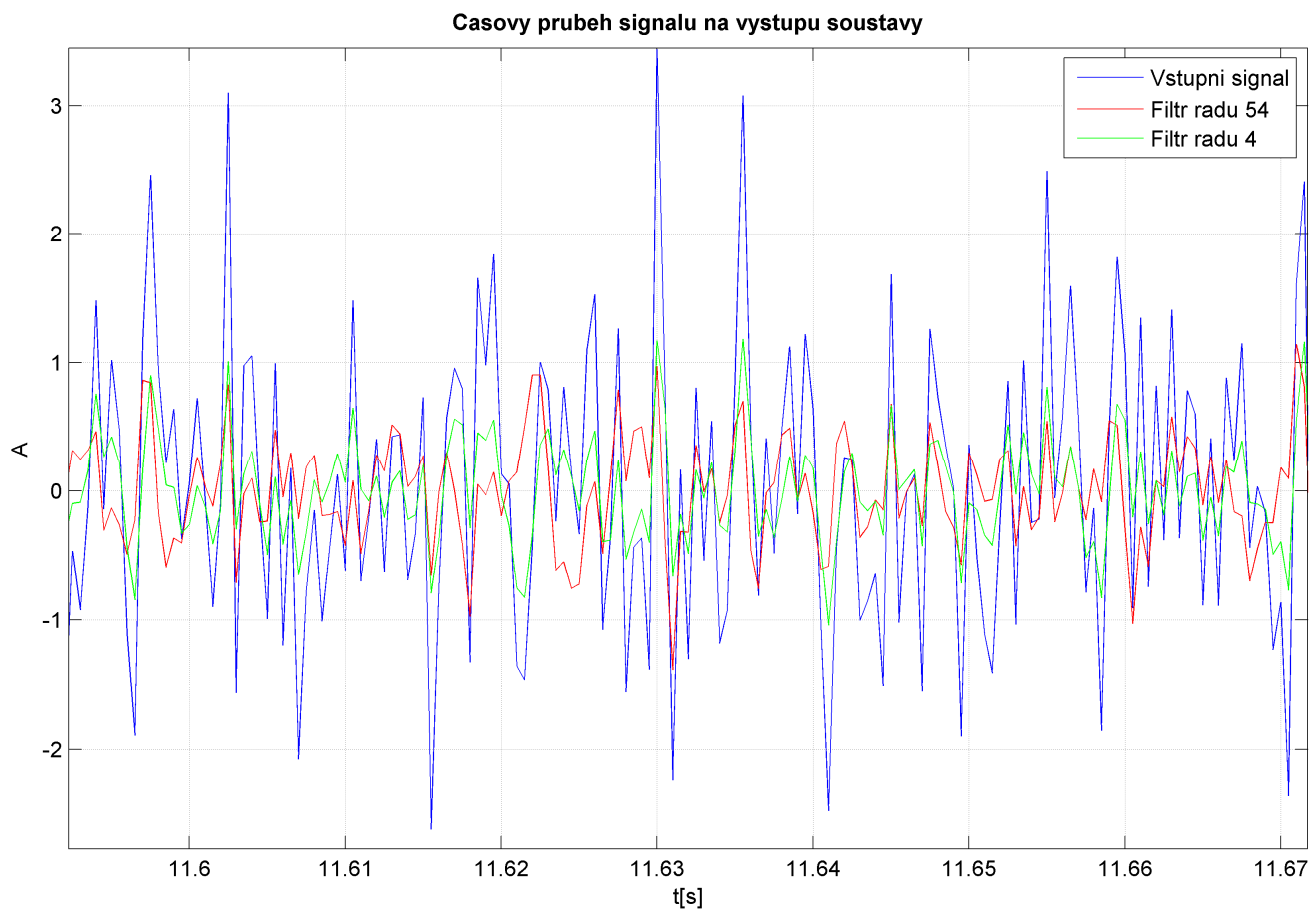
Obr. 6 Amplitudové charakteristiky filtru pro různé řády filtru. Nejlepší přiblížení poskytuje filtr vyššího řádu.



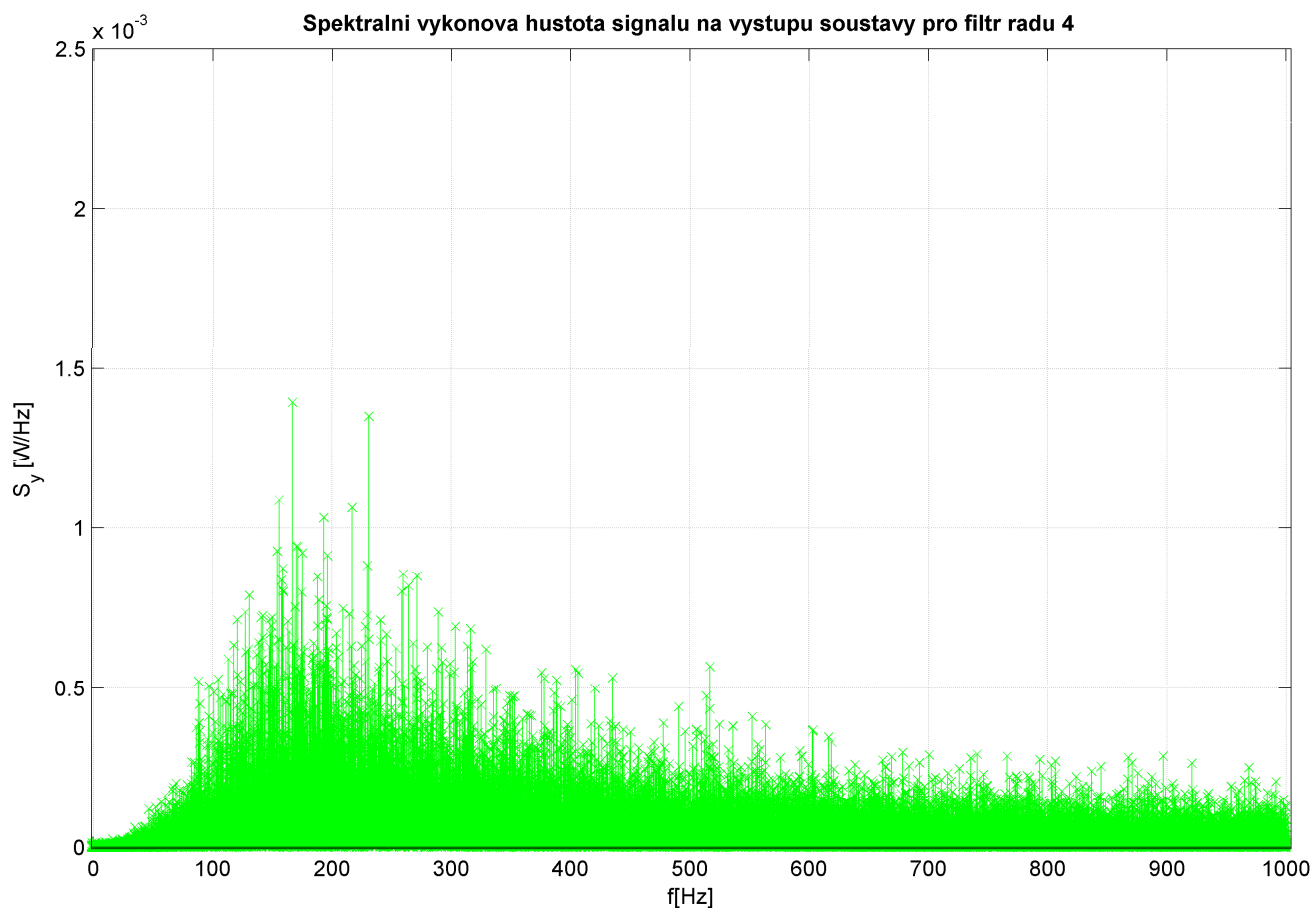
Obr. 7 Detail amplitudové charakteristiky filtru v okolí kmitočtu f_L .



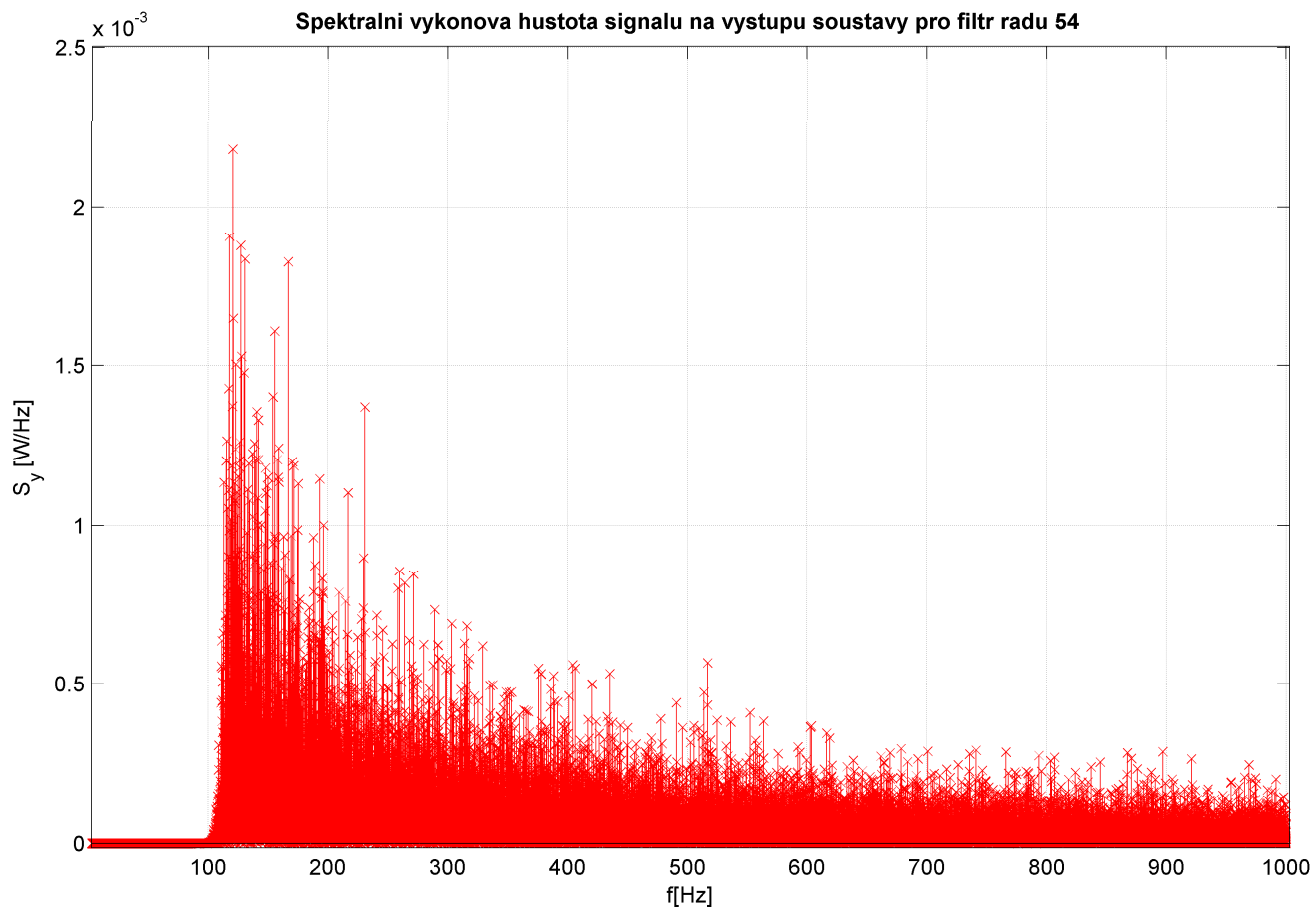
Obr. 8 Průběh výstupního signálu v závislosti na čase. Je patrná potlačená amplituda, což je způsobeno např. vlivem filtrace.



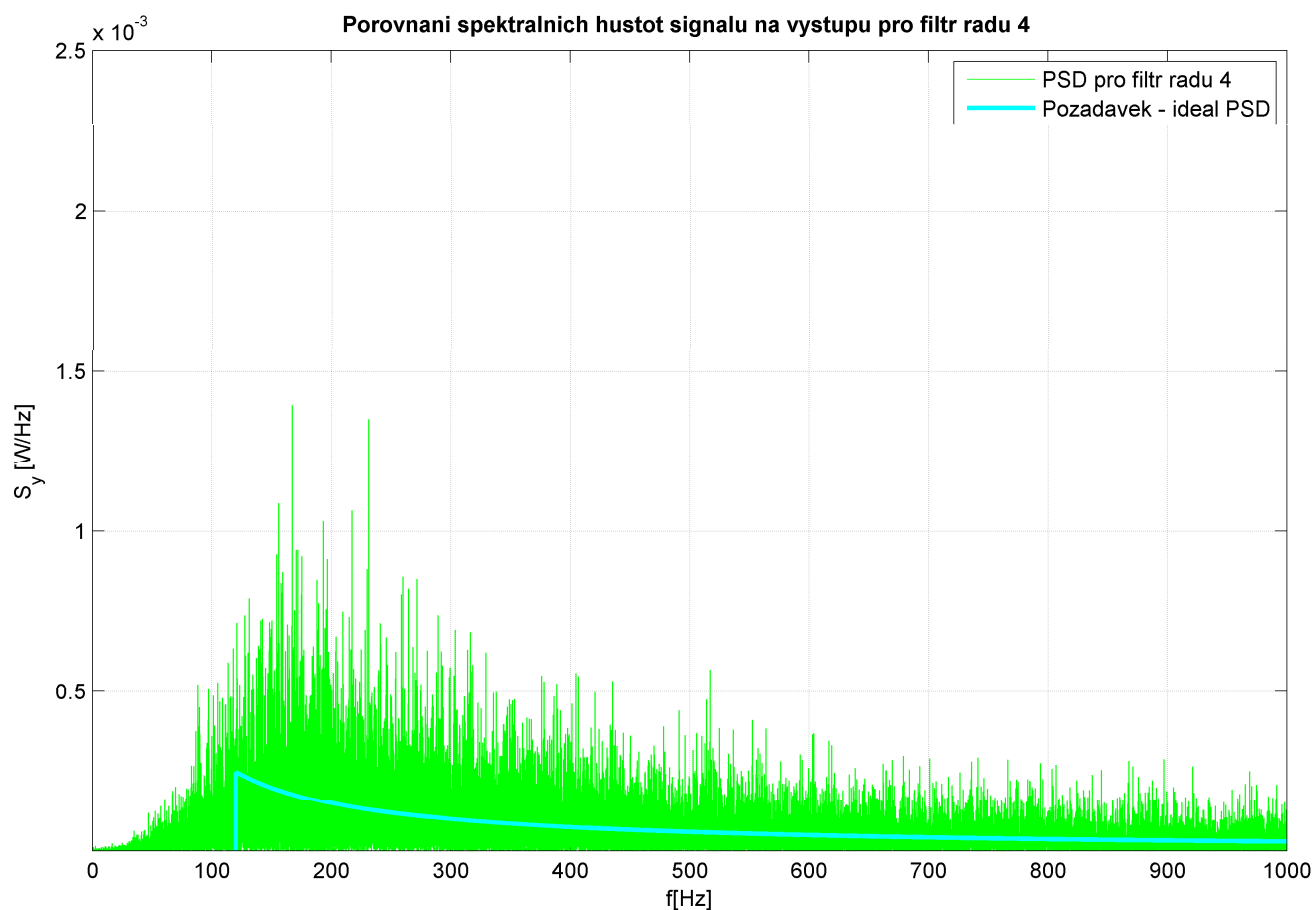
Obr. 9 Detail výstupního signálu ze soustavy v závislosti na čase. Je vidět, že někdy dochází někdy trochu ke zpoždění, zvláště u filtru vyššího řádu. Přesně se strejit do původního signálu ale ani není účelem této úlohy. Hlavní je, že signál splňuje kmitočtové požadavky na něj kladené.



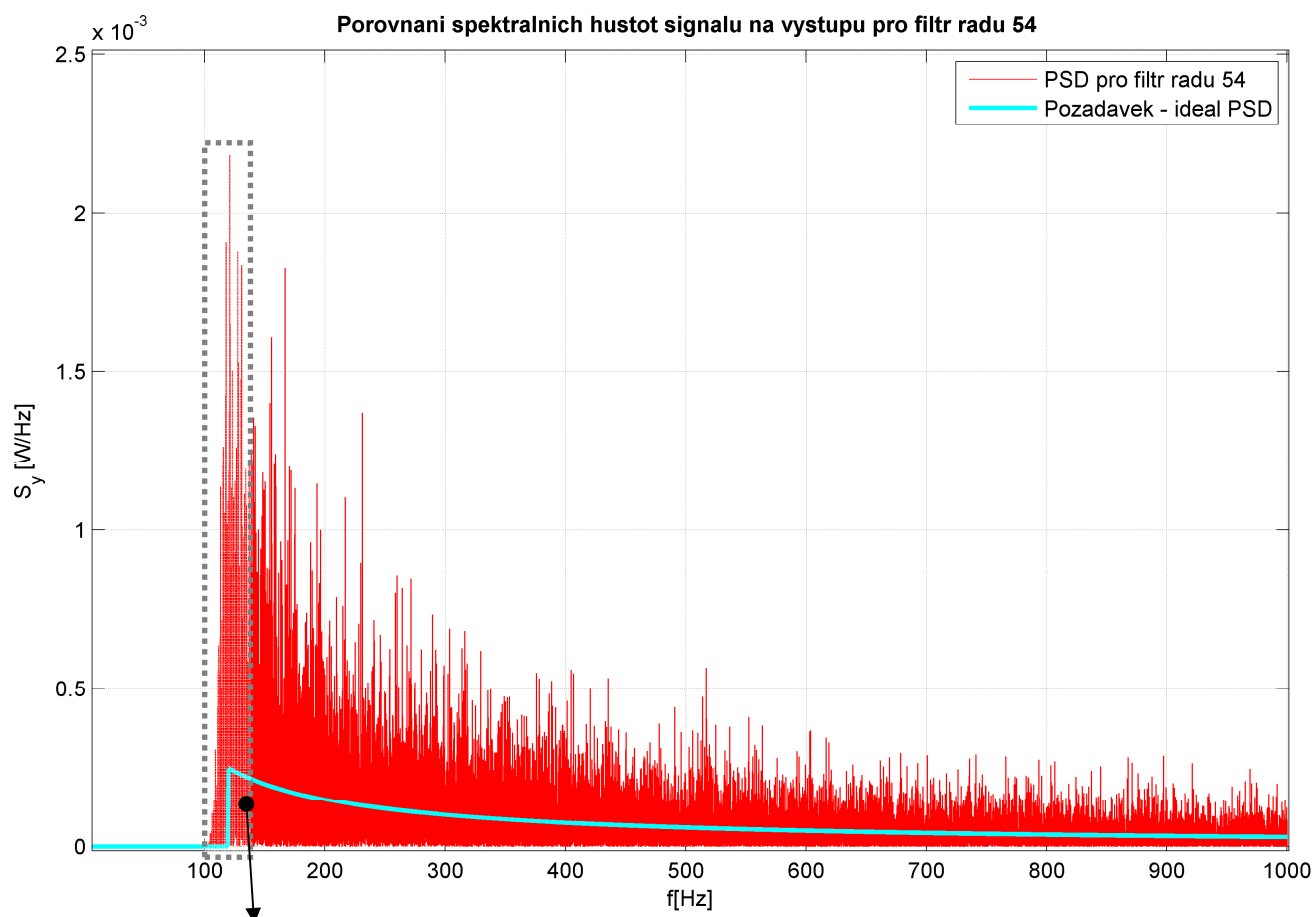
Obr. 10 Funkcí „stem“ vykreslená PSD na výstupu pro filtr nižšího řádu.



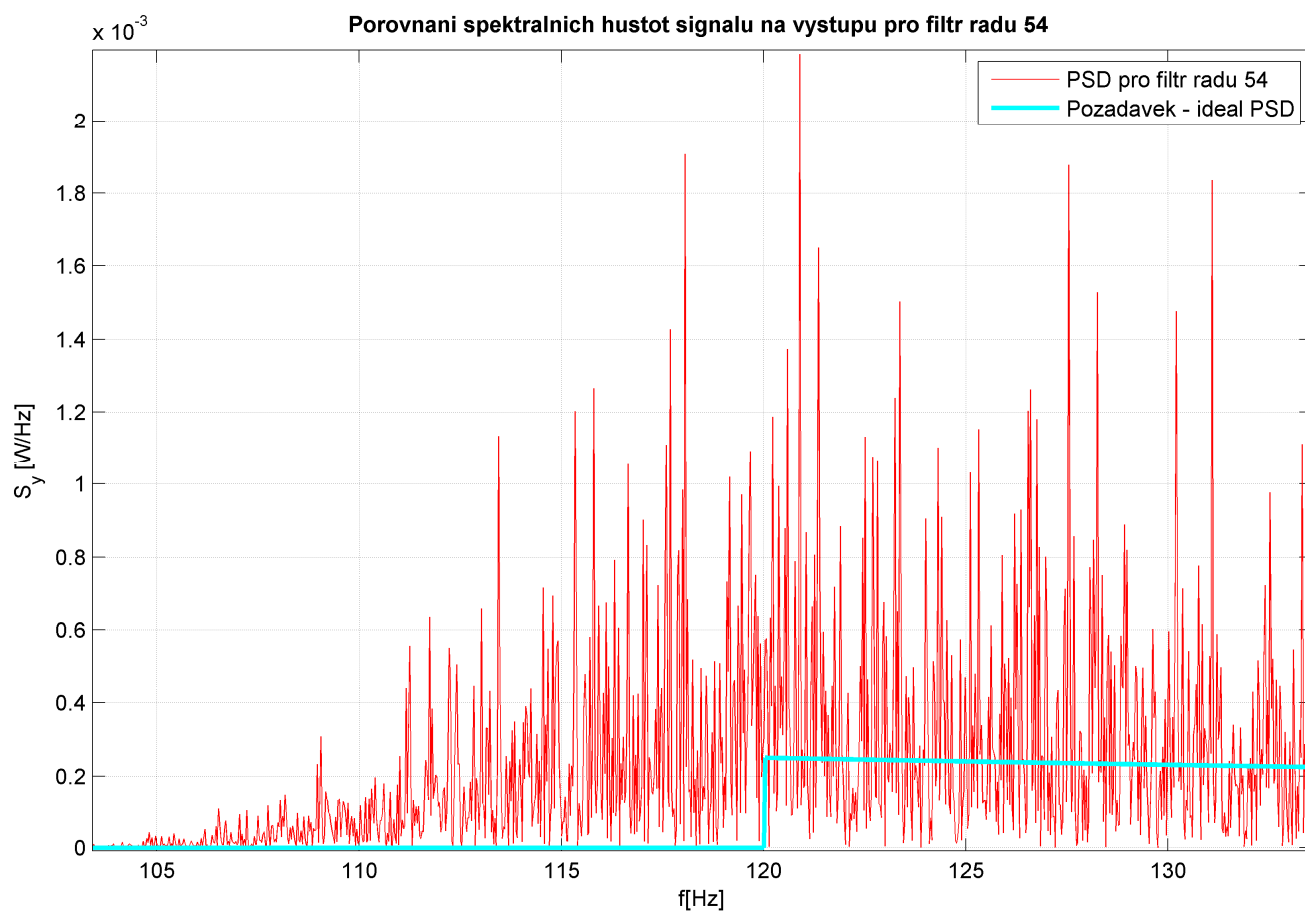
Obr. 11 Funkcí „stem“ vykreslená PSD na výstupu pro filtr vyššího řádu.



Obr. 12 Výstup PSD za filtrem nižšího řádu – porovnání s ideální PSD, kterou požadujeme. Je vidět, že např. díky šumu je skutečná charakteristika dosti nerovnoměrná. Pro tento řád filtru je aproximace dosti nedokonalá.



Obr. 13 Výstup PSD za filtrem vyššího řádu – porovnání s ideální PSD, kterou požadujeme. Je vidět, že poměrně snadno jsou ztlumeny frekvence pod f_L - viz. detail. Zde je znát, že překmit, ke kterému dochází je poměrně veliký oproti tomu, co bychom požadovali. V podstatě ale tak veliký není, protože to rozhodně není překmit o řád. Také jen některé ze spektrálních čar dosahují větších hodnot.



Obr. 14 Detail PSD v okolí f_L – je znát rychlý pád přenosu k nule.

4. ZÁVĚR

Úloha byla zajímavá a umožňovala nám dostatečně si ujasnit otázku filtrace apod. Bylo možno vyzkoušet mnoho variant – velikosti přechodového pásma, řádu filtru atd., a tak získat představu o chování filtru pro různé podmínky. Pomocí filtru se podařilo namodelovat fázový šum oscilátoru typu „flicker phase“.

Porovnáme-li jednotlivé filtry dle jejich amplitudových charakteristik, tak je vidět, že pro filtr nižšího řádu (např. 4) se nedaří dobře aproximovat uvažovanou ideální charakteristiku. Charakteristika filtru nízkého řádu je málo strmá – než tedy stihne „vystoupat“ do požadované výšky, už musí zase klesat. O filtrech s vyšším řádem (54) jsem se již zmiňoval. Ideální charakteristiku aproximuje poměrně dobře. Řešení však opět není úplně ideální, protože většinou „překmitne“ nad definovaný vrchol, což způsobuje zdůraznění frekvenčních složek blízkých f_L více než je požadováno.

Jak je vidět z grafů, tak některé složky spektra převyšují definovanou charakteristiku. Je to způsobeno právě tím, že filtr není ideální a je nutno vždy volit určitý kompromis dle toho, jakou vlastnost aktuálně potřebujeme zajistit. Další komplikací je, že se někdy vygenerované složky šumu s největší amplitudou „strefí“ právě do onoho citlivého pásma okolo f_L , což činí následné úpravy obtížnějšími a hůře vypadajícími.

Ačkoli byla úloha poměrně hravá, je velké množství možností, co tímto směrem ještě vyzkoušet. Lze se zabývat např. modely AR, MA, ARMA atd. Možné je rovněž podrobnější rozpracování této úlohy, co se týká např. většího počtu porovnávaných stupňů filtru

5. DODATEK

Contents

- [Projekt 3 - Generovani WSS procesu](#)
- [Nadefinovani parametru](#)
- [Vstup soustavy - bily sum](#)
- [Vystup soustavy - spektralni vykonova hustota](#)
- [Prenormovani pro metodu Yule - Walk](#)
- [Yule - Walk, ziskani koeficientu](#)
- [Vystupni signal po filtraci](#)
- [Porovnani PSD funkci](#)

Projekt 3 - Generovani WSS procesu

X37SGS - Signaly a soustavy Jan Hlidek

```
close all
clear all
clf
```



Nadefinovani parametru

```
format long
f_sa = 2000      % Vzorkovaci frekvence
N_sa = 2^16     % Pocet vzorku
var_x = 1       % Rozptyl
% Konstanty pro pozadavky na spektralni vykonovou hustotu
f_L = 120
f_H = f_sa / 2
K = 60
% Pro filtr
f_prechod = 0.1*f_L   % Prechodove pasmo filtru
% Rad filtru:
rad1 = 4
rad2 = 54

% Osy:
caso = 0 : 1/f_sa : (N_sa-1)/f_sa;
freqo = -(N_sa-1)/2*f_sa/N_sa : f_sa/N_sa : (N_sa-1)/2*f_sa/N_sa;

f_sa =

    2000
```

```

N_sa =
    65536

var_x =
    1

f_L =
    120

f_H =
    1000

K =
    60

f_prechod =
    12

rad1 =
    4

rad2 =
    54

```

Vstup soustavy - bily sum

```

% Proces s normalnim rozlozenim pro dany rozptyl:
x = sqrt(var_x)*randn(1, N_sa);
stred_hodn = mean(x)           % ...kontrola ss

% Spektralni vykonova hustota sig. na vstupu
S_x = 1/N_sa/f_sa*abs(fft(x)).^2;

% Graf signalu na vstupu v zavislosti na case
figure;
plot(caso, x);
grid on
title('Casovy prubeh signalu na vstupu soustavy');
xlabel('t[s]');
ylabel('A');
% Graf signalu na vstupu - ve spektru
% neuvadime v dBm/Hz, ale v W/Hz. Grafy pak vypadaly lepe.
% ze to lze uvadi uvodem napr.:
% http://www.utko.feec.vutbr.cz/~silhavy/MVDP/index_soubory/7_MVDP.pdf
figure;
stem(freco, fftshift(S_x), 'bx')
title('Spektralni vykonova hustota signalu na vstupu soustavy');
xlabel('f [Hz]');
ylabel('S_x [W/Hz]');
grid on

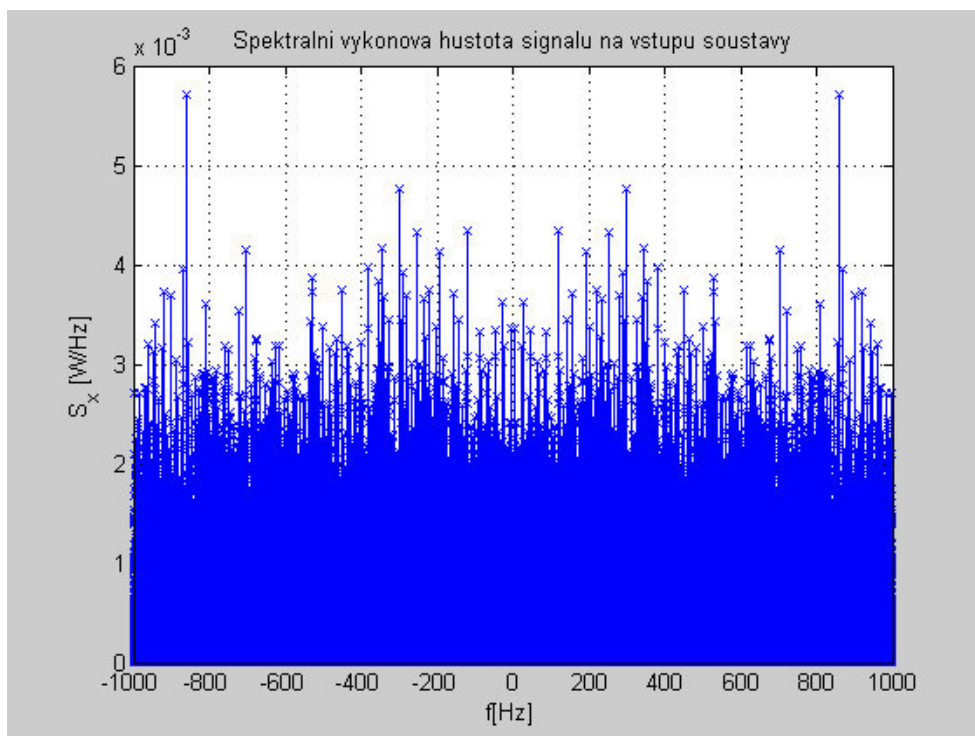
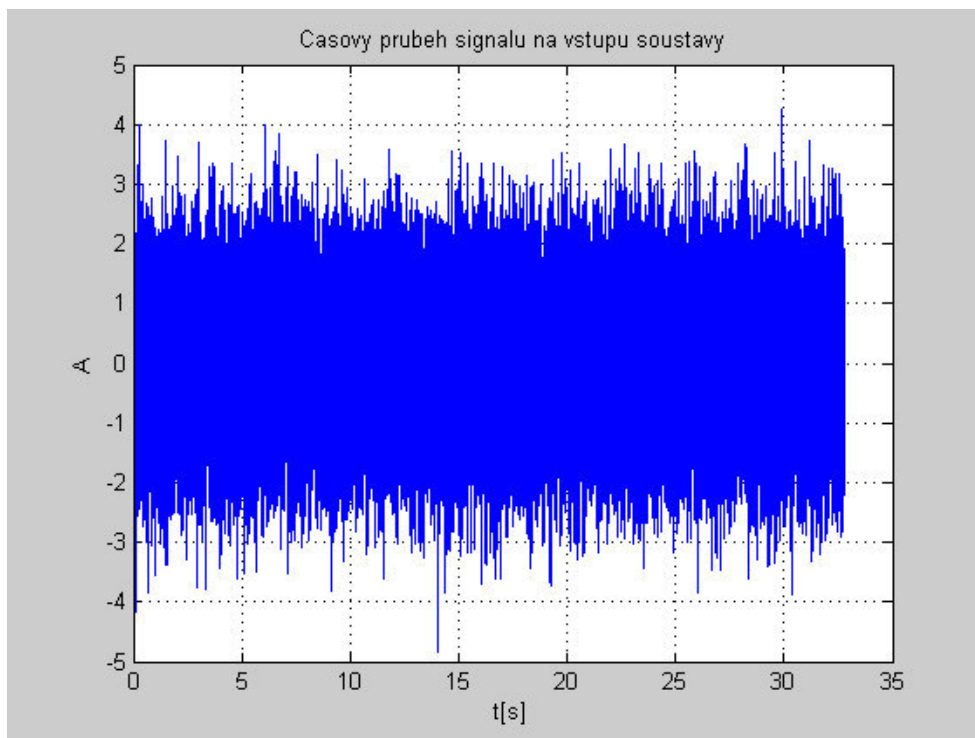
%Kontrola pro vstup - Korelacni funkce
% ... pouze pro kontrolu, ze se jedna o skutecne bily sum, nekorelovany.
% Normovano jako korelace energeticka, je to pouze pro kontrolu...
cor_noise = xcorr(x);
caso_cor = -(N_sa-1)/f_sa : 1/f_sa : (N_sa-1)/f_sa;
figure;
plot(caso_cor, cor_noise)
title('Autokorelacni funkce signalu na vstupu soustavy');
xlabel('{\tau}[s]');
ylabel('R_{en}');
grid on

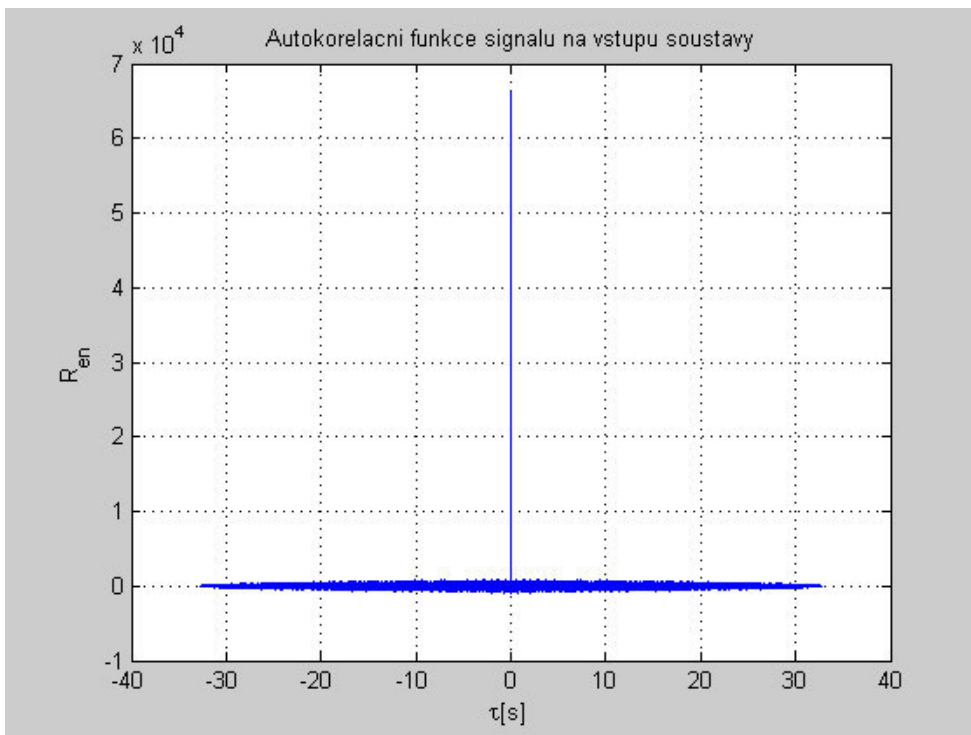
```

```
%*****
```

```
stred_hodn =
```

```
0.005510571355271
```



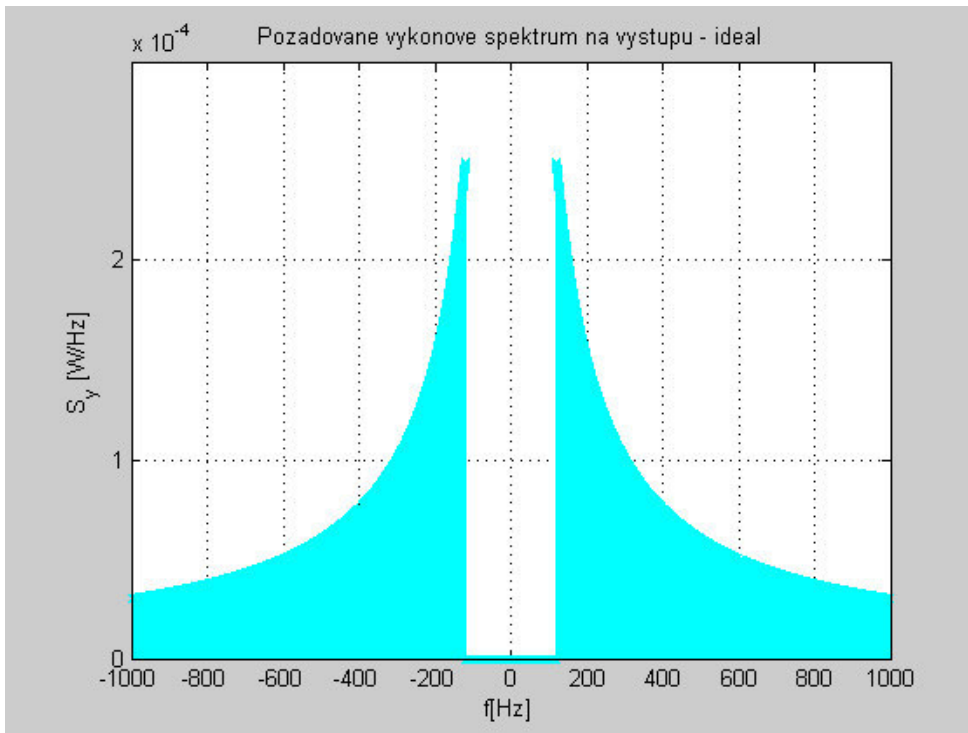


Vystup soustavy - spektralni vykonova hustota

Provedeme realizaci, abychom tak mohli modelovat fazovy sum oscilatoru dle zadani ($S(f) = K/f$). Napr. vynulovani mezi 0 az f_L atd. pomoci cyklu...

```
S_y(0+1) = 0; % S_y ... jen spektrum, do grafu nutno upravit!
i = 0;
for i = 1 : N_sa/2,
    if ( i*f_sa/N_sa < f_L ) | ( i*f_sa/N_sa > f_H ),
        S_y(i+1) = 0;
        S_y(N_sa-i+1) = 0;
    else
        S_y(i+1) = K/(i*f_sa/N_sa);
        S_y(N_sa-i+1) = K/(i*f_sa/N_sa);
    end
end

% Vykresleni vykonoveho spektra na vystupu - idealne co pozadujeme:
% abychom dostali stejne jednotky s kterymi budeme dale pocitat - abychom
% tedy mohli grafy PSD dale porovnavat, prejdeme na vyjadreni W/Hz - cili
% nasobime 1/f_sa
figure;
stem(freco, 1/f_sa*fftshift(S_y), 'cx')
title('Pozadovane vykonove spektrum na vystupu - ideal');
xlabel('f [Hz]');
ylabel('S_y [W/Hz]');
grid on
```



Prenormovani pro metodu Yule - Walk

```

% !!! pocet hodnot v pasmech (zkusit pocitat z N_sa)
% Pozor, zde je nutno nekdy ZMENIT pri zmene vstupnich parametru delici
% pomer! (napr. aby bylo i v neprop. pasmu dost vzorku)
N_neprop = round((( N_sa / f_sa ) * f_L)/8)
N_prop = round((( N_sa / f_sa) * ((f_sa/2) - f_L))/10)
N_yule = N_neprop + N_prop % Celkove vzorku pro navrh filtru

% Frekvence pro yule (vyzaduje 0 az 1)
f_neprop = 0 : (f_L - f_prechod)/(N_neprop-1) : f_L - f_prechod;
f_prop = f_L : (f_H - f_L)/(N_prop-1) : f_H;
f_yule = [f_neprop f_prop]; % Jednotlive casti pasma dany dohromady

% Prenosova funkce H ... Uziti amplitudove charakteristiky filtru,
% vztahu abs(H) = sqrt( S_y / var_x )
H_neprop = zeros(1, N_neprop);
i = 0;
for i = 0 : N_prop-1,
    H_prop(i+1) = sqrt((K/(f_L + i*(f_H - f_L)/N_prop-1))/var_x);
    % ... H jen kladne, abs lze neuvazovat
end
% Jednotlive casti pasma dame dohromady
H_yule = [H_neprop H_prop];

N_neprop =
    492

N_prop =
    2884

N_yule =
    3376

```

Yule - Walk, ziskani koeficientu

```

% Impulsova odezva pro rady filtru pomoci yulewalk
% nanormovani frekvence kvuli vyžadovanému rozmezi funkci - viz nize
[b1,a1] = yulewalk(rad1, f_yule/(f_sa/2), H_yule);
[b2,a2] = yulewalk(rad2, f_yule/(f_sa/2), H_yule);
% yulewalk designs recursive IIR digital filters using a least-squares fit

```

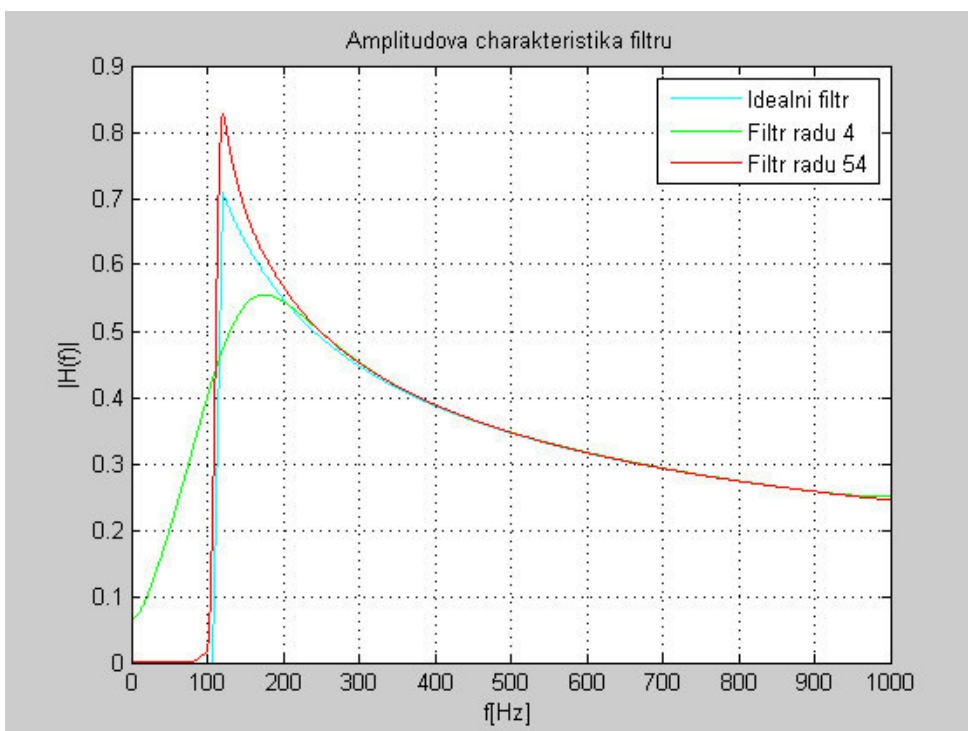
```

% to a specified frequency response.
% [b,a] = yulewalk(n,f,m) returns row vectors b and a containing the n+1
% coefficients of the order n IIR filter whose frequency-magnitude
% characteristics approximately match those given in vectors f and m:
% f is a vector of frequency points, specified in the range between 0
% and 1, where 1 corresponds to half the sample frequency (the Nyquist
% frequency). The first point of f must be 0 and the last point 1, with
% all intermediate points in increasing order. Duplicate frequency points
% are allowed, corresponding to steps in the frequency response.
% m is a vector containing the desired magnitude response at the points
% specified in f.
% f and m must be the same length.
% plot(f,m) displays the filter shape.

% Prenosova charakteristika tohoto "diskretniho" filtru, který potřebujeme
% pro její vykreslení. Signal filtrujeme az nize
[H_rad1,f_rad1] = freqz(b1, a1, N_yule, f_sa);
[H_rad2,f_rad2] = freqz(b2, a2, N_yule, f_sa);
% [h,f] = freqz(b, a, l, fs) returns the frequency response vector h and the
% corresponding frequency vector f for the digital filter whose transfer
% function is determined by the (real or complex) numerator and denominator
% polynomials represented in the vectors b and a, respectively. The vectors
% h and f are both of length l. For this syntax, the frequency response is
% calculated using the sampling frequency specified by the scalar fs
% (in hertz). The frequency vector f is calculated in units of hertz (Hz).
% The frequency vector f has values ranging from 0 to fs/2Hz.

% Graficke znazorneni prenosovych charakteristik filtru (charakteristiky
% zisku ...  $H^2$ )
figure;
plot(f_yule, abs(H_yule), 'c-')
hold on
plot (f_rad1, abs(H_rad1), 'g-')
hold on
plot (f_rad2, abs(H_rad2), 'r-')
title('Amplitudova charakteristika filtru');
xlabel('f [Hz]');
ylabel('|H(f)|')
legend('Ideální filtr', ['Filtr radu ',sprintf('%1.0f ',rad1)], ['Filtr radu ',sprintf('%1.0f ',rad2)]
grid on

```



Vystupni signal po filtraci

```

% Pokud pouzijeme vyse vypocteny filtr na vstupni signal (sum), ziskame zde
% odpovidajici vystupni signal:
y_rad1 = filter(b1, a1, x);
y_rad2 = filter(b2, a2, x);

% Casovy prubeh
figure;

```

```

plot(caso,x,'b-');
hold on
plot(caso,y_rad2,'r-');
hold on
plot(caso,y_rad1,'g-');
title('Casovy prubeh signalu na vystupu soustavy');
xlabel('t[s]');
ylabel('A');
grid on
legend('Vstupni signal', ['Filtr radu ',sprintf('%1.0f ',rad2)], ['Filtr radu ',sprintf('%1.0f ',rad1)

% Korel. fce na vystupu - pro zajimavost
cor_noise1 = xcorr(y_rad1);
cor_noise2 = xcorr(y_rad2);
figure;
plot(caso_cor, cor_noise,'b-')
hold on
plot(caso_cor, cor_noise1,'g-')
hold on
plot(caso_cor, cor_noise2,'r-')
title('Autokorelacni funkce signalu na vystupu soustavy porovnana se vstupem');
xlabel('{\tau}[s]');
ylabel('R_{en}');
grid on
legend('vstupni', ['rad filtru ',sprintf('%1.0f ',rad1)], ['rad filtru ',sprintf('%1.0f ',rad2)],1);
xlim([-0.03 0.03]);
ylim([-2300 70000]);

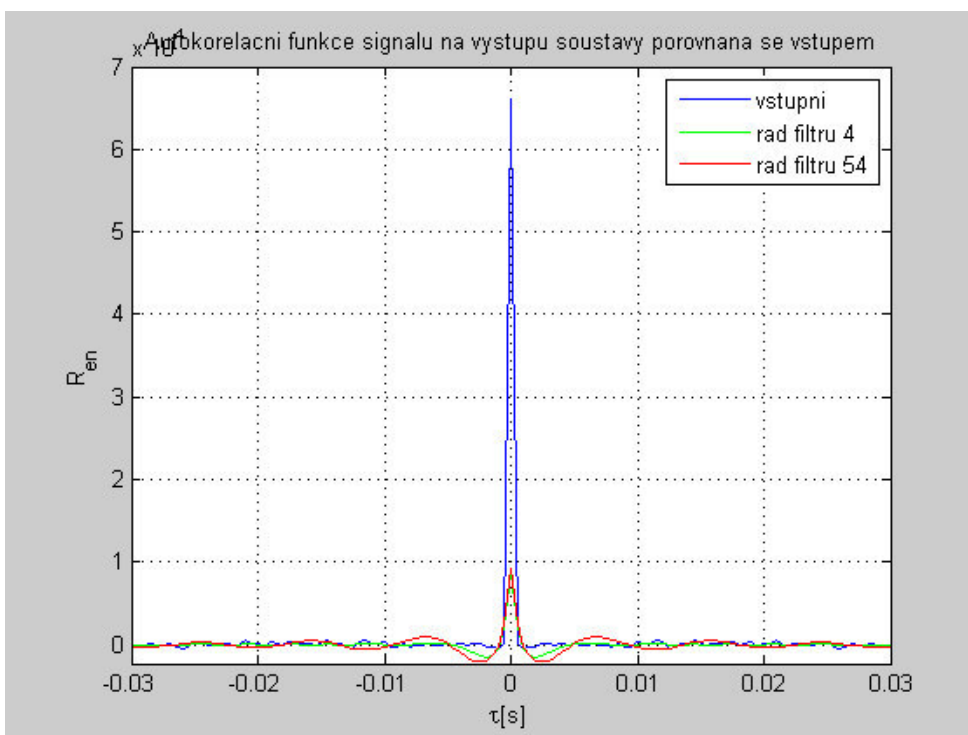
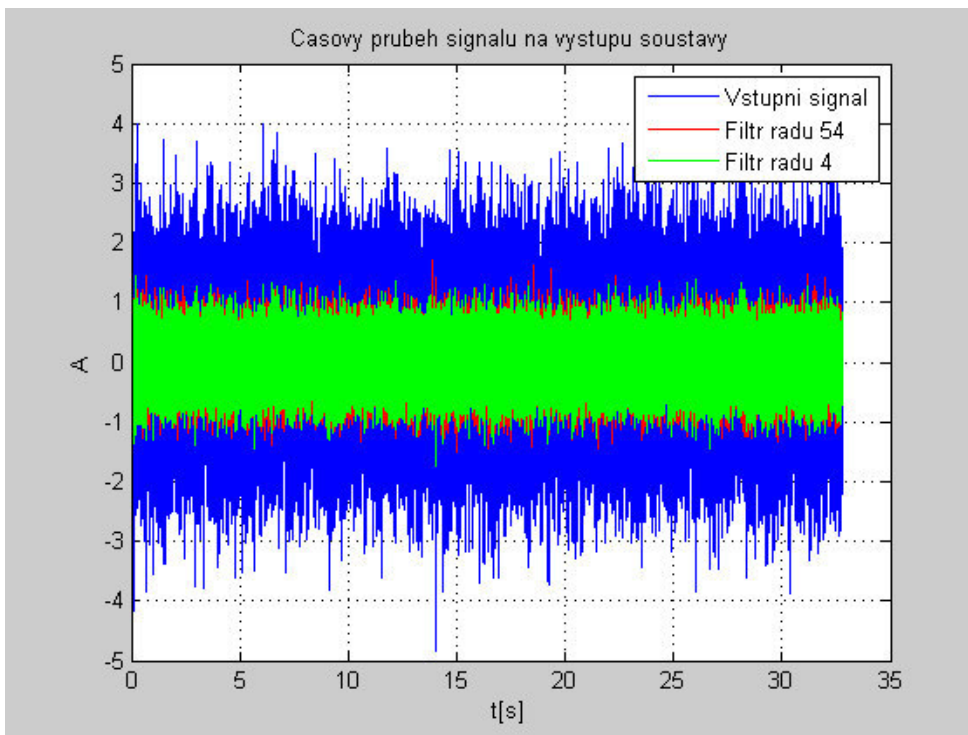
% Spektralni vykonova hustota na vystupu po filtraci
S_y_rad1_filter = 1/N_sa/f_sa*((abs(fft(y_rad1))).^2);
S_y_rad2_filter = 1/N_sa/f_sa*((abs(fft(y_rad2))).^2);

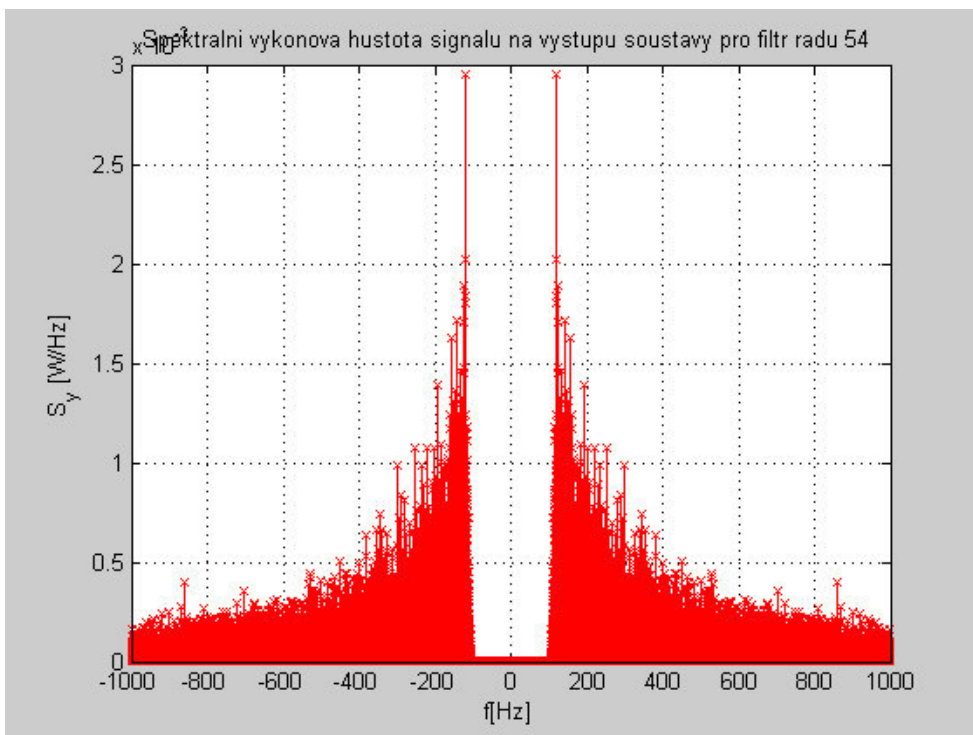
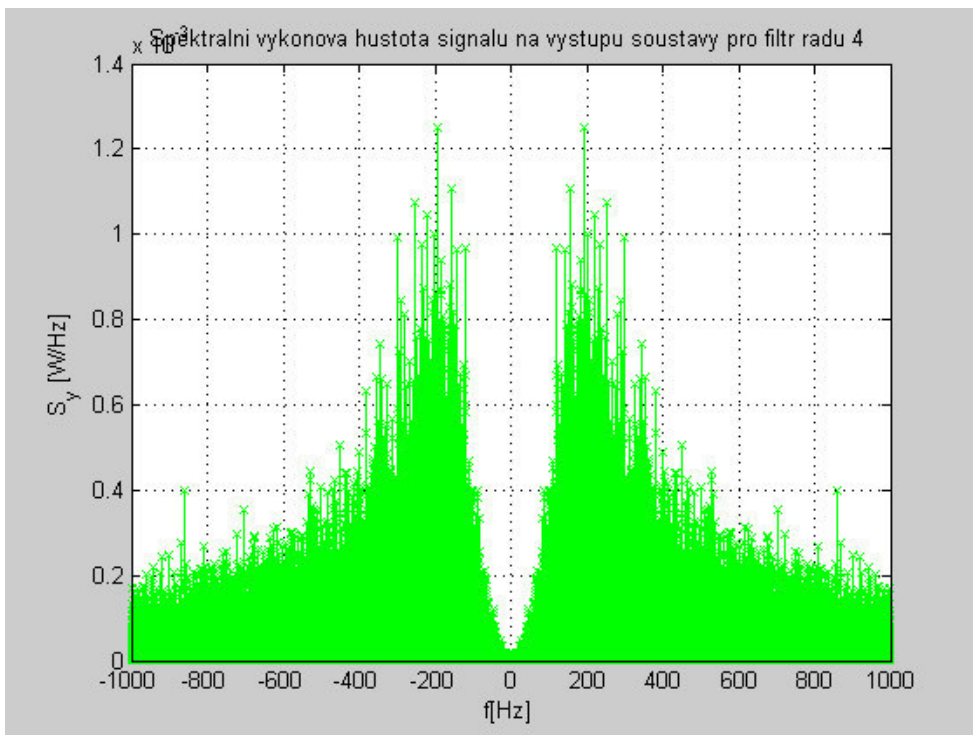
% Prubeh spektralnich hustot pro ruzne rady filtru
figure;
stem(frezo, fftshift(S_y_rad1_filter),'gx')
title(['Spektralni vykonova hustota signalu na vystupu soustavy pro filtr radu ',sprintf('%1.0f ',r
xlabel('f[Hz]');
ylabel('S_y [W/Hz]');
grid on

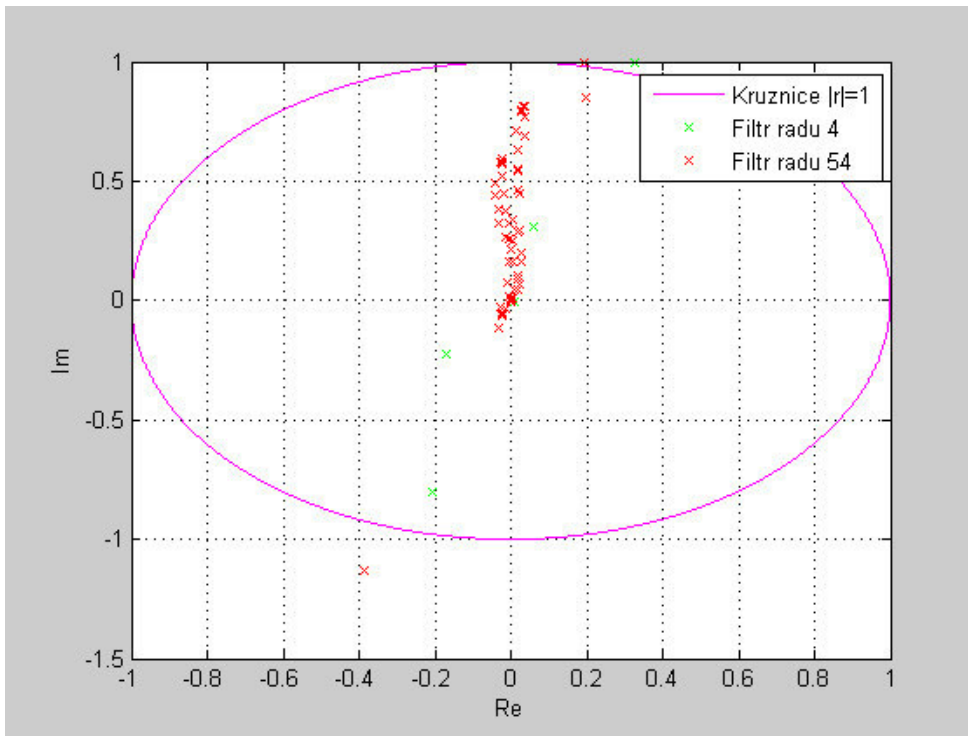
figure;
stem(frezo, fftshift(S_y_rad2_filter),'rx')
title(['Spektralni vykonova hustota signalu na vystupu soustavy pro filtr radu ',sprintf('%1.0f ',r
xlabel('f[Hz]');
ylabel('S_y [W/Hz]');
grid on

% Koeficienty filtru v komplexni rovine - pro ukazku
figure;
plot( exp( 2*pi*j*(0:1e-5:1)), 'm'); % vykresleni kruznice
hold on
plot( b1,a1, 'gx');
hold on
plot( b2,a2, 'rx');
xlabel('Re');
ylabel('Im');
grid on
legend('Kruznice |r|=1', ['Filtr radu ',sprintf('%1.0f ',rad1)], ['Filtr radu ',sprintf('%1.0f ',rad2

```







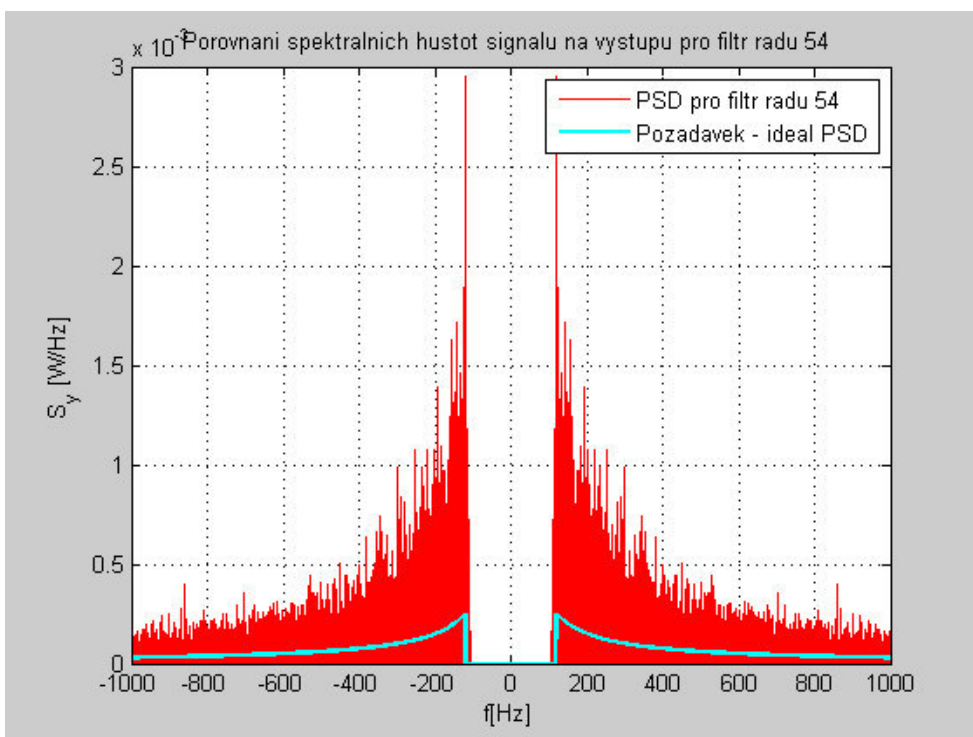
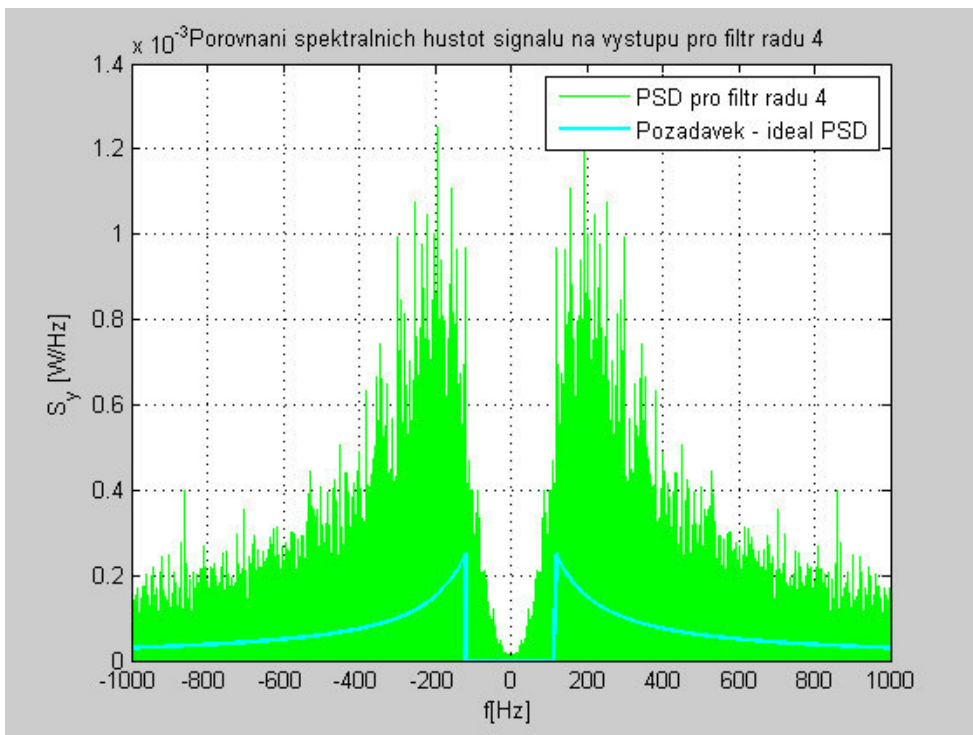
Porovnaní PSD funkcí

```
figure;
plot (freqo, fftshift(S_y_rad1_filter), 'g-')
hold on
plot(freqo, 1/f_sa*fftshift(S_y), 'c-', 'LineWidth', 2)
title(['Porovnaní spektrálních hustot signálu na výstupu pro filtr radu ', sprintf('%1.0f ', rad1)]);
xlabel('f [Hz]');
ylabel('S_y [W/Hz]')
grid on
legend(['PSD pro filtr radu ', sprintf('%1.0f ', rad1)], 'Pozadavek - ideal PSD', 1);

figure;
plot (freqo, fftshift(S_y_rad2_filter), 'r-')
hold on
plot(freqo, 1/f_sa*fftshift(S_y), 'c-', 'LineWidth', 2)
title(['Porovnaní spektrálních hustot signálu na výstupu pro filtr radu ', sprintf('%1.0f ', rad2)]);
xlabel('f [Hz]');
ylabel('S_y [W/Hz]')
grid on
legend(['PSD pro filtr radu ', sprintf('%1.0f ', rad2)], 'Pozadavek - ideal PSD', 1);

% Cim je vetsi prechodove pasmo, tim je prekmit mensi

% Pro studovani vlivu radu filtru je dobre jeste napr. nekolikrat menit v
% zahlavi hodnoty rad1 a rad2.
```



Published with MATLAB® 7.4