I9: Analýza číslicových signálů v časové oblasti, číslicové filtry (příklady, návrh a použití jednoduchých filtrů a filtrů vyšších řádů), kvantování a jeho důsledky. (ZZS)

Analýza v časové oblasti je už dost popsaná v I7 (SAS), kterou jsem dělala. Je tam obecně co je signál, dělení signálů a speciální signály, vzorkování, kvantování, energie a výkon signálu, autokorelační funkce a vzájemná korelace. Doplním sem ty praktičtější věci, které tam chybí, protože se v SASu nebraly (aliasing, periodizace a prosakování). Prezentace ze ZZS tady: https://moodle.fel.cvut.cz/course/view.php?id=4288&lang=cs (prezentace 2,4,5,7,8).

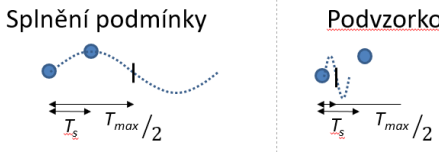
Vzorkování

Vzorkovací teorém

Rekonstrukce signálu

- mezi naměřenými hodnotami očekáváme spojité proložení

- pro rekonstrukci signálu se musí mezi dva sousední vzorky vejít více než půlperioda nejrychlejší harmonické složky



$$T_s < \frac{T_0}{2}$$

$$\frac{1}{f_s} < \frac{1}{2fmax}$$

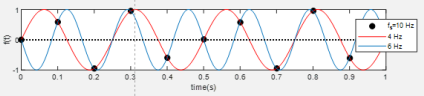
$$f_s > 2fmax$$

$$f_{max} < {}^{f_s}/_2$$

Při nedodržení vzorkovacího teorému dochází k aliasingu. To je způsobeno tím, že od sebe nelze rozeznat signál s frekvencí f a frekvencí fs-f -> spektrum se "přezrcadlí" a kolem fs/2 se překrývá -> sečtou se hodnoty od různých frekvenčních složek a hodnoty spektra jsou špatně.

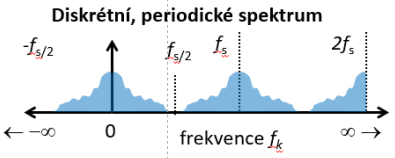
Příklad nejednoznačného vyjádření frekvence→ 6Hz složka se přičte ke 4Hz

f_s=10 Hz
$f_s/2$=5 Hz
f_0=6 Hz
$f_N(f_0)$=4 Hz



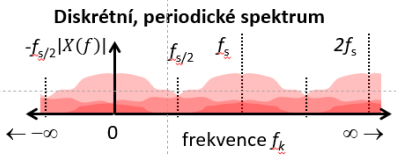
Vzorkovací teorém

$$f_{max} < {}^{f_s}/_2$$



Nedodržení teorému

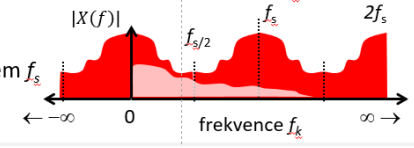
$$f_{max} > {}^{f_s}/_2$$



$$f > {}^{f_s}/_2$$

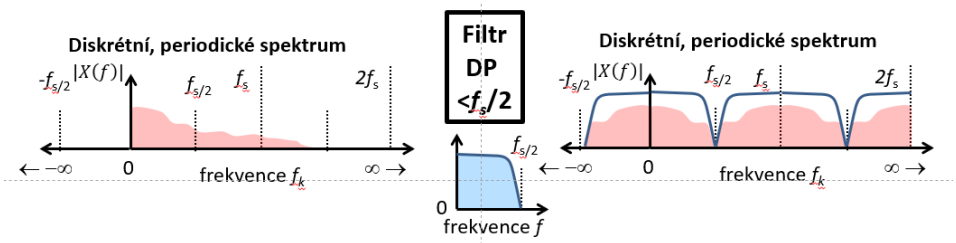
$$f_p(f) \triangleq |f + pfs|$$

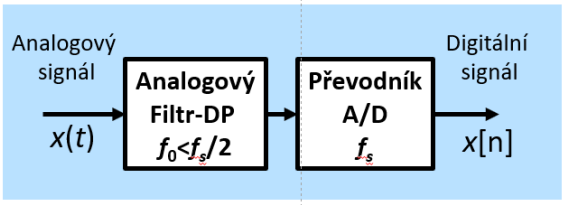
... frekvence se zrcadlí kolem f_s v jiné periodě spektra p



Aliasing

Aliasingu se předchází tím, že se signál předem filtruje dolní propustí s vhodnou mezní frekvencí. Vyšší frekvence, které nás už nezajímají se tím odfiltrují a můžeme vzorkovat s Fs, kterou potřebujeme (fs/2 > mezní frekvence DP). DP je tam v praxi skoro vždycky, většinou je v signálu nějaké vysokofrekvenční rušení, které by dělalo bordel, takže je třeba ho odfiltrovat. před tím, než signál AD převodníkem digitalizujeme.



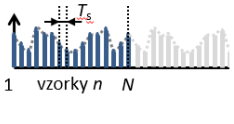


Periodizace

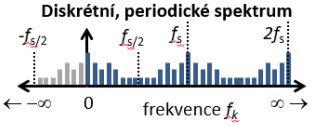
V praxi máme v naprosté většině případů z měření k dispozici neperiodický finitní signál. Když provedeme FFT, signál se periodizuje - celý náš signál pomyslně tvoří jednu periodu a naskládá se nekonečně krát za sebe.

Periodizace:
Budeme předstírat, že finitní signál $n \in (1, N)$ je jedna perioda

$$\Omega = 2\pi \frac{f}{f_s}$$
$$\Omega = 2\pi \frac{k}{N}$$



Diskrétní FT (DFT)



$$y[n] = \sum_{k=0}^{N-1} Y[k] e^{i\Omega n}$$

DFT

Inverzní

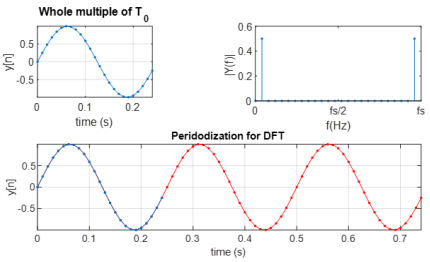
$$Y[k] = \frac{1}{N} \sum_{n=1}^{N} y[n] e^{-i\Omega n}$$

Problém je, že ty pomyslné periody na sebe většinou nenavazují a vznikají mezi nimi strmé “skoky”. Ty pak ve spektru vytvářejí vysoké frekvence, které v původním signálu ve skutečnosti nebyly. Tomu se říká prosakování.

Prosakování

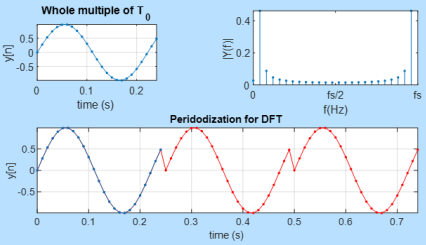
DFT

- Finitní neperiodický signál
- N signálu $= T_0$ harmonické
- Periodizujeme
- Hladký harmonický signál
- Ostrá spektrální čára



Prosakování

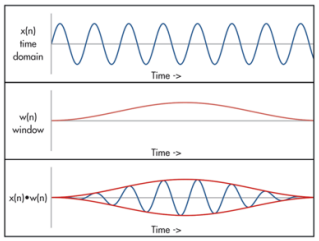
- Finitní neperiodický signál
- N signálu $\neq T_0$ harmonické
- Periodizujeme
- Neharmonický signál
- Spektrálně bohatý signál



Omezení prosakování

Váhování oknem

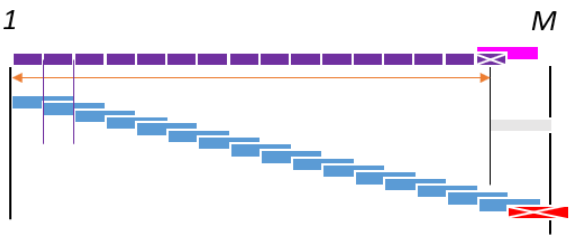
- Maska tlumící okraje signálu
- Tlumí nespojitosti u periodizace
- Omezí prosakování
- Rozmaže spektrum
- Snižuje energii
- Potlačuje krátké jevy mimo polovinu okna
- Množství typů: (Hamming, Hanning, ...)
- Hlavní lalok spektra okna
  - Široký – složen z více pomalých harmonických (rozmaže, ale neprosakuje)
  - Úzký – složen z méně pomalých, ale i rychlých harmonických



Omezit se dá buď tak, že budeme volit správně dlouhé úseky, aby periody navazovali (to v praxi dost těžko zařídíme), nebo se dá signál vážit oknem, které potlačí okraje signálu a omezí tím “skoky” na hranicích period (tam se ale zase nevyhneme rozmazání spektra).

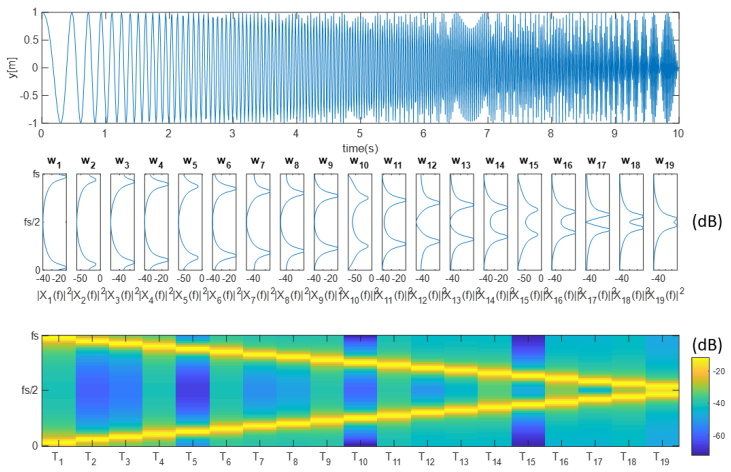
Časová segmentace a spektrogram

Využívá se, pokud chceme signál analyzovat po časových úsecích (například spektrogram, zpracování stochastických signálů). Celý finitní signál si rozdělíme na konečný počet úseků (typicky stejně dlouhých). Délku úseku volíme podle potřeby - kratší úsek znamená vyšší časové rozlišení, ale naopak menší rozlišení (větší rozmazání) ve spektru ve spektrogramu. Typicky se segmenty dělají s určitým překryvem, signál v segmentu se taky váží oknem (Hamming atd.), aby nedocházelo k prosakování při periodizaci signálu během FFT. Okna potlačují signál na krajích úseků, a je proto dobré, aby se úseky překrývaly.



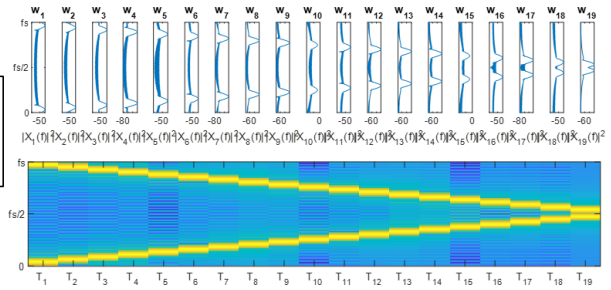
Pokud se pro každý úsek spočte FFT, dostaneme spektra pro každý úsek. Vypočítá se kvadrát spektra (spektrální hustota) a vyjádří se v dB (postihnou se tak rozdíly ve všech řádech stejnoměrně). Kvadráty spektra se dají sestavit “za sebe” a tím dostaneme spektrogram. Spektrogram tedy poskytuje informaci o tom, jak se v čase mění zastoupení frekvencí v signálu.

Spektrogram



Je dobré každý úsek váhovat oknem a využít doplnění signálu nulami, aby se zvýšilo frekvenční rozlišení (na intervalu <0,fs> bude více spektrálních čar s kratšími rozestupy -> jemnější rozlišení).

váhování
doplnění „0“
FFT



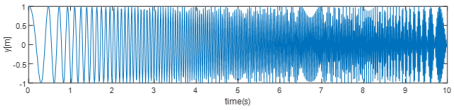
Podle toho, jestli chceme spíše vysoké časové, nebo frekvenční rozlišení volíme délku okna.

Vliv segmentace na spektrogram

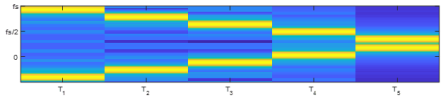
Délka okna

- *N vzorků* = *K* spektrálních čar $f \in \langle 0, fs)$
- $\uparrow N \rightarrow \downarrow \Delta f$ (lepší frekvenční rozlišení) $\uparrow T$ (horší časové rozlišení) ↑CPU
- $\downarrow N \rightarrow \uparrow \Delta f$ (horší frekvenční rozlišení) $\downarrow T$ (lepší časové rozlišení) ↓CPU

f_s=100 Hz
$T=N.f_s$
$\Delta f=f_s/N$

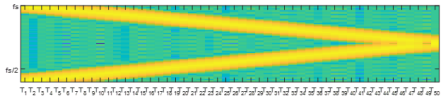


N=200 (2s)
ΔT=2 s
Δf=0.5 Hz



Skoková dynamika

N=20 (0.2 s)
ΔT=0.2 s
Δf=5 Hz

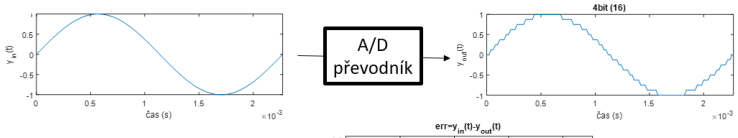


Rozmazaná frekvence

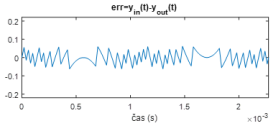
Kvantování a jeho důsledky

- amplituda je diskrétní, nespojitá
- počet hodnot, které signál může nabývat je dán A/D převodníkem (2,4,8,16,...n bitů, počet úrovní 2^n)
- v důsledku toho vzniká kvantizační chyba. Ta bude nejvýše ½ kvantizačního kroku (rozestupu mezi hladinami) a to v případě, že se skutečná hodnota signálu trefí přesně do poloviny mezi dvě kvantizační hladiny.
- SQNR říká, jak byla pro daný signál kvantizace kvalitní.
- Správná volba kvantizačního kroku a AD převodník závisí na aplikaci (tabulka dole), někdy nevadí, že kvantizace je “hrubá” a SQNR vyšší, protože je pak prozec kvantizace rychlejší / signál pak zabírá míň místa v paměti.

Kvantizační šum

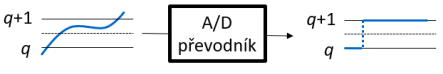


Kvantizační šum
Rozdíl mezi vstupem a výstupem



Kvantizační šum

Kvantizační chyba
< ½ kvantizačního kroku



$$SQNR \approx 20 \log_{10}(2^B)\ [\text{dB}]$$

B... počet bitů převodníku

Odstup signál-šum SQNR (Signal to Quantization Noise Ratio)

- Energie signálu
- Energie kvantizačního šumu

$$SQNR = \frac{E\{y_{in}\}}{E\{y_{in} - y_{out}\}} = \frac{E\{y_{in}\}}{E\{err\}}\ [-]$$

$$SQNR_{dB} = 10 \log_{10}(SQNR) \qquad [\text{dB}]$$

$$E = \int_{t_1}^{t_2} |y(t)|^2 dt$$

$$E = \sum_{n=1}^{N} |y[n]|^2$$

Rozlišení sluchu

Bitů	Hodnot	*SQNR* [dB]	Pozn.
24	16,8×10⁶	144,49	Studiové audio, DVD
16	65.536	96,33	Běžné audio, CD
12	4096	72,24	Komprese
8	256	48,16	Telefon

Výhody digitální reprezentace signálu:

- Libovolná fyzikální veličina -> datová řada
- Datová řada -> matematická analýza
- Strojové zpracování (kontrolery, PC, smartphone)
- Snadný převod mezi analogem a digitálem (A/D, D/A)
- Zpracování/analýza je numerické, není závislá na dalších veličinách (historicky např. analogové počítače)
- Ukládání a komprese
- **Odolnost proti rušení** (stejně se ale může projevit, skutečný analogový signál je snímán senzorem a to není bezchybné + i AD převodníky způsobují chybu)

LTI soustavy

Filtry jsou obecně LTI soustavou, kterou lze reprezentovat impulzní odezvou. Impulzní odezva je odezva systému na diracův impuls na vstupu (v případě diskrétních signálů jednotkový impuls). Protože spektrum diracova impulsu je rovnoměrné, přenos systému (a teda i filtru) je vyjádřen spektrem impulzní odezvy. Přenos je komplexní a má tedy modul (reálnou část) a fázi (imaginární část). Modul popisuje útlum/zesílení harmonických složek, fáze reprezentuje časové zpoždění harmonických složek. Průchod signálu LTI kauzální s pamětí (teda i filtrem) vede ke zpoždění signálu.

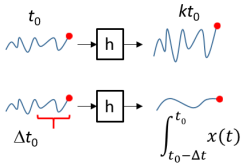
Rozdělení systémů

Kauzální

- bez paměti (ideální zesilovač)
  - pouze přítomnost
- s pamětí (filtr)
  - přítomnost + minulost

Nekauzální

- přítomnost + minulost + budoucnost



- **Invariantní** - systém se nemění v čase
- **Kauzální** - signál na výstupu (časový) závisí na vstupním signálu
- **Lineární** – platí princip superpozice

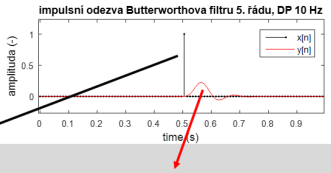
- **LTI lze popsat impulzní odezvou**

Identifikace systému

- Vstupní signál $x[n]$ je průchodem systému h změněn na $y[n]$

$x[n]$ → **h** → $y[n]$

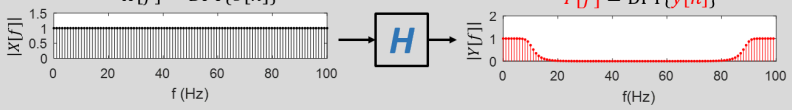
- Do vstupu jednotkový impulz $x[n] = \delta[n]$
- Výstup je impulzní odezva $y[n]$



V časové oblasti je výsledek průchodu signálu LTI soustavou vyjádřen konvolucí signálu s impulzní odezvou LTI systému. Ve spektrální oblasti je spektrum signálu po průchodu LTI systémem vyjádřeno váhováním (násobením) spektra signálu přenosovou funkcí LTI systému.

Ve spektru:

$X[f] = \mathrm{DFT}\{\delta[n]\}$ → **H** → $Y[f] = \mathrm{DFT}\{y[n]\}$



Jaký je přenos systému?

$$H[f] = \frac{Y_{out}[f]}{X_{in}[f]} \Rightarrow \quad Y[f] = X[f] \cdot H[f]$$

Dosadíme spektrum jednotkového impulzu

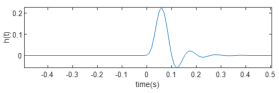
$$Y[f] = \Delta[f] \cdot H[f]$$
$$Y[f] = 1 \cdot H[f]$$
$$Y[f] = H[f]$$

Spektrum impulzní odezvy je frekvenční přenos systému !!!

$$x[n] * h[n] \approx X[f] \cdot H[f]$$

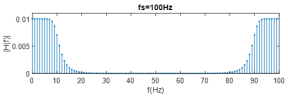
$x[n]$ → $*h[k]$ → $y[n]$

$y[n]$: konvoluce signálu a impulzní odezvy



$X[f]$ → $\cdot H[f]$ → $Y[f]$

$Y[f]$: spektrum signálu váhováno spektrem systému



V případě diskrétních finitních signálů (většina prakticky používaných) je třeba nějak vyřešit "přečuhující" okraje při výpočtu krajních hodnot konvoluce -> buď se doplňuje nulami, nebo se signál periodicky zopakuje

Co dát na okraje signálu a masky?

- standardně 0
- v DSP využijeme „periodizace signálu"

0 5 1 2 3 4 6 9 0 0 0 0 0 0 0 5 1 2 3 4 6 9 0 0 0 0 0 0 5 1 2 3 4 6 9 0 0 0 0 0 0

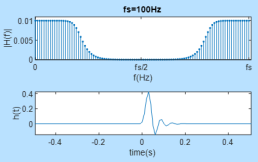
$\overline{h[k]}$ -1 2 1

M

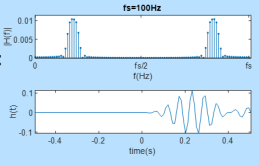
Filtry

= systém bránící průchodu vybraným harmonickým složkám
- přenos a impulsní odezva
- signál prochází systémem a je konvolován s impulsní odezvou
- ve spektrální oblasti je systém frekvenčně selektivní, spektrum signálu je váženo přenosem systému

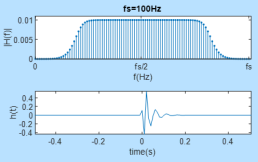
Dolní propust
(Horní zádrž)
Low pass
(High stop)



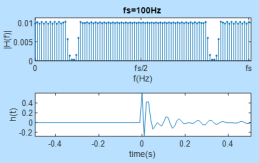
Pásmová propust
Band pass



Horní propust
(Dolní zádrž)
High pass
(Low stop)



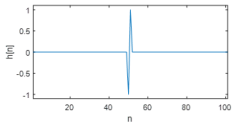
Pásmová zádrž
Band stop

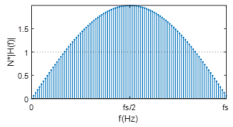


Příklady filtrů

Příklady jednoduchých systémů

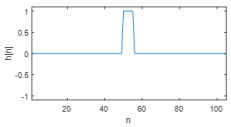
Diferenciátor
- $y[n] = x[n] - x[n-1]$
- Jak vypadá konvoluční maska *h*? $y[n] = x[n] * h[n]$
- $h[n] = [-1, +1];$

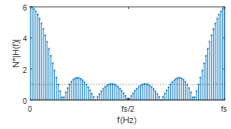




Horní propust
Zesiluje vysoké kmitočty
H(*f*)>1

Sumátor konečné délky
- $y[n] = x[n] + x[n-1] + \cdots + x[n-5]$
- Jak vypadá konvoluční maska *h*? $y[n] = x[n] * h[n]$
- $h[n] = [1\ 1\ 1\ 1\ 1\ 1];$

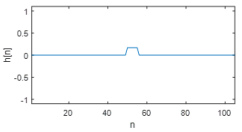


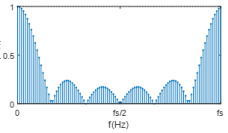


Dolní propust
Zesiluje nízké kmitočty
H(*f*)>1

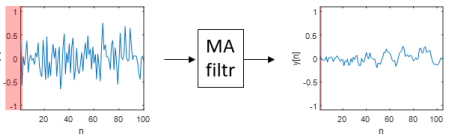
Klouzavý průměr (Moving Average MA-filter)
- Obecný průměr: $\bar{x} = \frac{1}{N}\sum_N x[n]$

Př. *N*=6 (průměr šesti sousedních vzorků)
- $y[n] = \frac{1}{N}(x[n] + x[n-1] + \cdots + x[n-5])$
- Jak vypadá konvoluční maska *h*? $y[n] = x[n] * h[n]$
- $h[n] = [1/6, 1/6, 1/6, 1/6, 1/6, 1/6];$





Dolní propust
Zesiluje nízké kmitočty
H(*f*)≤1



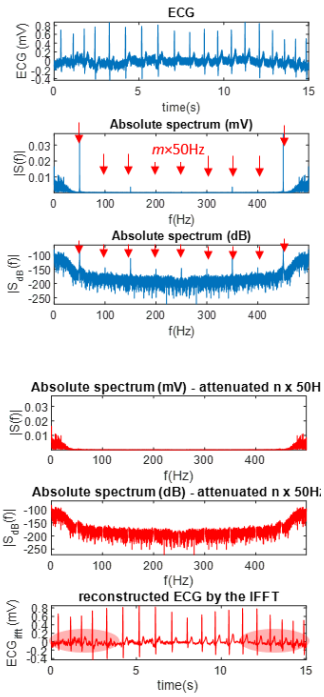
$$y[n] = \frac{1}{6}\sum_{k=0}^{6} x[n+k]$$

Sumátor je vhodný pro potlačení vysokofrekvenčního šumu - šum se vyprůmětuje a zůstanou jen pomalejší složky signálu (nižší frekvence).

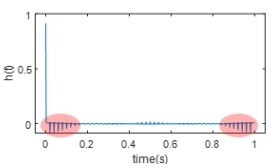
Filtrace ve spektru je snadná (prostě vynulujeme ty frekvenční složky, které nám vadí). Má ale nevýhody - upravené spektrum má pak strmé přechody, které se po IFFT v časové oblasti projeví ošklivou "zubatou" impulzní odezvou. Každá rychlá strmá změna v původním signálu v časové oblasti je podobná jako jednotkový impuls (např. ty R-špičky v EKG), a vybudí tedy impulzní odezvu. Ta se pak projeví při průchodu signálu filtrem - za každou špičkou se k signálu přičtou ty "zuby". Není to proto moc použitelné -> kompromis při návrhu filtru.

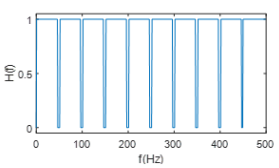
Příklady jednoduchých systémů

Filtrace ve spekltru (nulování spektrálních čar)
- z minulého cvičení
- EKG + síťový brum
- Nulování násobků spektrálních čar 50 Hz
- IDFT



Jak vypadá impulzní odezva filtru?



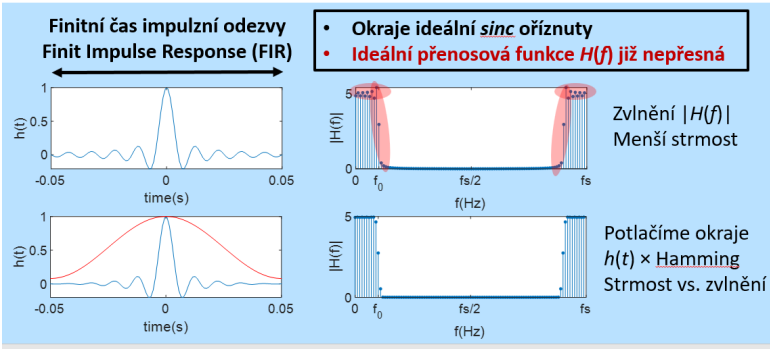


Každá nespojitost ~ odezva na jednotkový skok
Každá R-špička ~ jednotkový impulz

Filtrace ve spektru → ošklivá impulzní odezva
Návrh filtru → kompromis mezi strmostí a odezvou

FIR (Finite impulse response)

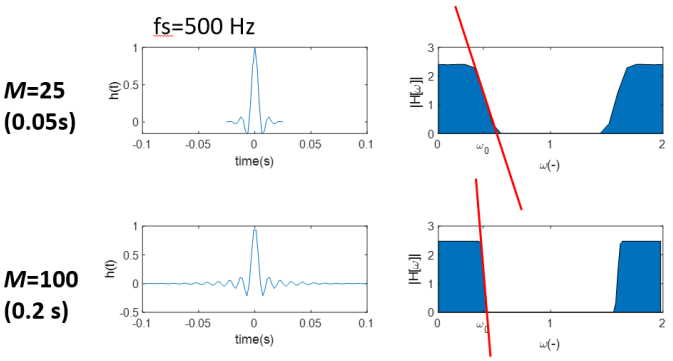
Impulzní odezva ideální DP je sampling function (Sa), která má hodnoty na (-Inf, Inf). Reálně můžeme digitálně reprezentovat jen finitní úsek, takže ze Sa se vezme jen M+1 vzorků. V důsledku toho není DP ideálně strmá a je v propustném pásmu zvlněná. Zvlnění jde zábrání tím, že se Sa převází tlumícím oknem (Hamming atd.). Podobně to platí i pro ostatní typy filtrů (HP, pásmové propustě a zádrže). Řád filtru M odpovídá počtu vzorků Sa, které se využijí. Čím vyšší řád, tím strmější filtr (takže lepší). Nevýhoda je, že aby byl filtr dost strmý a použitelný, musí se využít hodně vzorků (stovky, tisíce) a filtrace je pak výpočetně náročná.



Strmost FIR filtru

Řád filtru (filter order) *M*

- Odpovídá délce impulzní odezvy o délce *M*+1
- Nekonečná strmost (ideální) pro $h(t) \in (-\infty, \infty)$

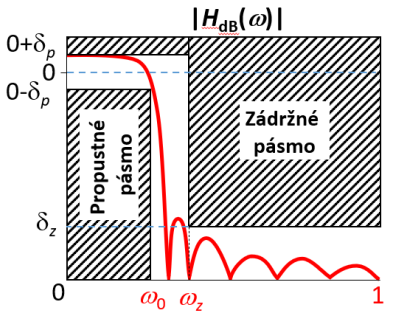


Při návrhu filtru nás zajímají následující parametry:

Ideální vs. reálný filtr

Toleranční pásmo filtru

- Zvlnění v propustném pásmu δ_p
- Minimální útlum v zádržném pásmu δ_z
- Mezní (zlomový) kmitočet ω_0 (-3dB)
- Frekvence, kdy je dosažen požadovaný útlum ω_z (strmost filtru)



Hledáme odpovídající impulzní odezvu (konvoluční masku)

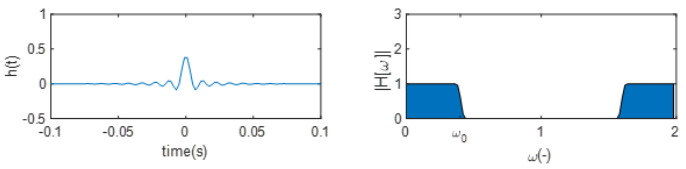
$\omega \in (0,2)$ – normovaný kmitočet [-]

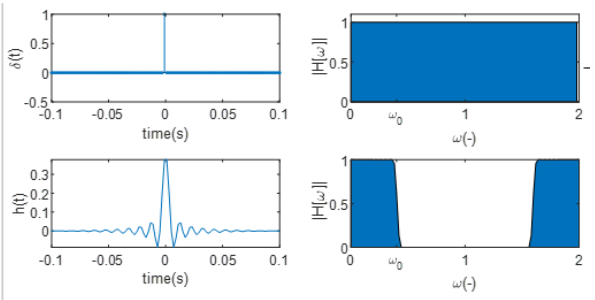
Jiné typy filtrů (HP, pásmová propusť a zádrž) se dají odvodit z DP.

Dolní propust:

$$A = {}^{f_s}/_{2f_0}$$

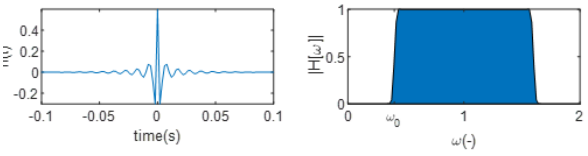
$$h(t) = A^{-1}\frac{\sin(\Omega_0 t)}{\Omega_0 t}$$

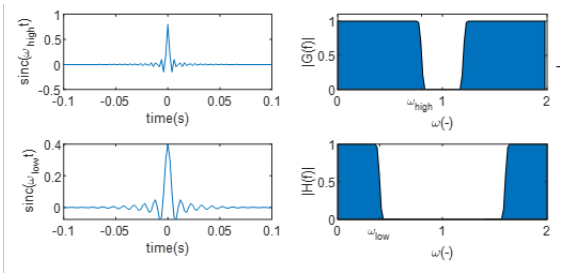




Horní propust:

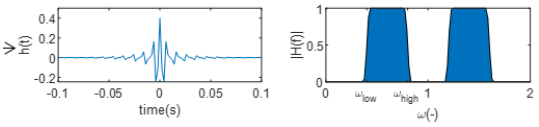
$$h(t) = \delta(t) - A^{-1}\frac{\sin(\Omega_0 t)}{\Omega_0 t}$$





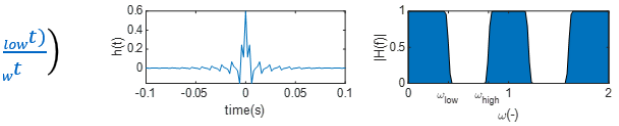
Pásmová propust:

$$h(t) = A^{-1}\frac{\sin(\Omega_{high}t)}{\Omega_{high}t} - B^{-1}\frac{\sin(\Omega_{low}t)}{\Omega_{low}t}$$



Pásmová zádrž:

$$h(t) = \delta(t) - A^{-1}\frac{\sin(\Omega_{high}t)}{\Omega_{high}t} + B^{-1}\frac{\sin(\Omega_{low}t)}{\Omega_{low}t}$$



Dá se navrhovat i jinými způsoby než přímo přes Sa:

Prototypová okna

- sinc(Ωt) + Hamming (funkce *fir1* v MATLAB)

Vzorkování spektra

- Spojitý předpis přenosové funkce
- Vzorkování spektra: $\omega \in (0, 2)$ počet čar = řád filtru *M*+1
- Odpovídající impulzní odezva (funkce *fir2* v MATLAB)
- (viz. cvičení identifikace systému, impulzní odezva, konvoluce)

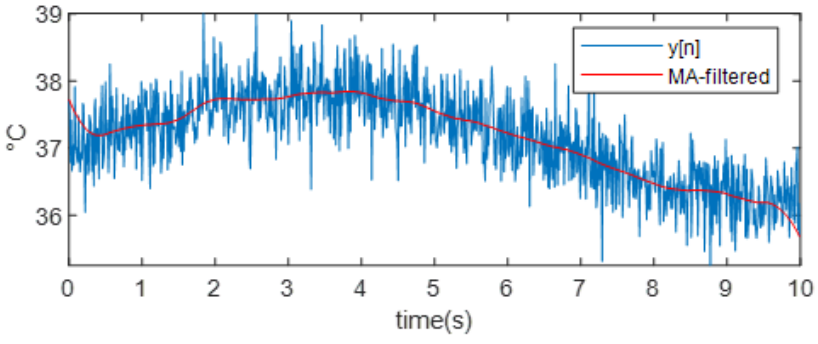
Optimalizační metody

- Optimalizační prokládání tolerančního pásma
- řád, typ okna, polynom aproximace apod.
- *firls*, *firpm* v MATLAB

filterDesigner app v MATLAB

Příklady jednoduchých FIR filtrů

Klouzavý průměr (moving average) - průměrování v okně (slouží k vyhlazení signálu)

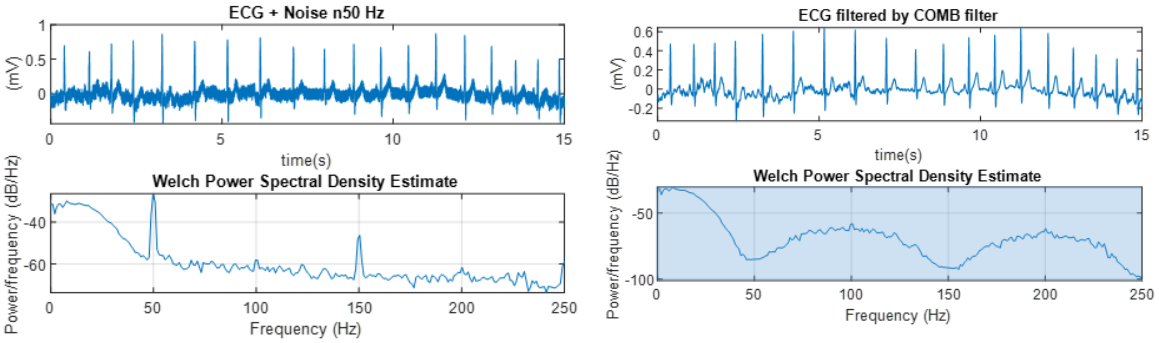


Použití:

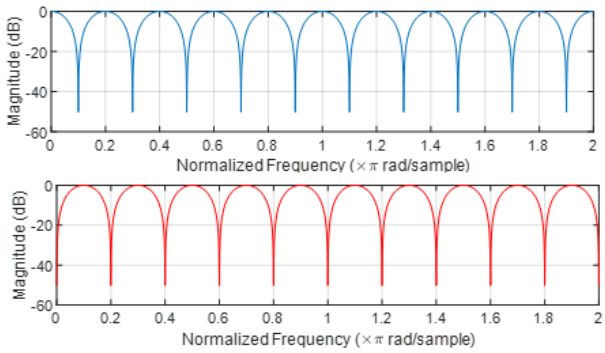
- Dolní propust
- Vyhlazení signálu
- Aritmetická průměr
- Apriorní pravděpodobnost

- impulzní odezva: $b[n] = [{}^{1}/_{N}, {}^{1}/_{N}, \ldots, {}^{1}/_{N}]$

Hřebenový filtr - pásmová zádrž (filtruje buď sudé nebo liché násobky zvolené frekvence)



- impulzní odezva $b[n]=[0.5\ 0\ \ldots\ 0\ \pm 0.5]$



$+b_D$ liché násobky f_0

$$\omega = \frac{f_0}{f_s} = \frac{1}{2D} \rightarrow \boldsymbol{D = \frac{f_s}{2f_0}}$$

$-b_D$ sudé násobky f_0

$$\omega = \frac{f_0}{f_s} = \frac{1}{D} \rightarrow \boldsymbol{D = \frac{f_s}{f_0}}$$

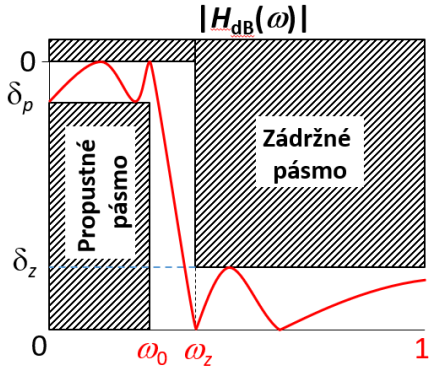
IIR filtry (Infinite impulse response)

Impluzní odezva je předepsána komplexním polynomem, hodnoty jsou definovány na (-Inf, Inf). IIR filtry mají i zpětnou vazbu, to znamená že hodnoty na výstupu se vrací "zpět" a přispívají k výpočtu následujících hodnot. Zpětná vazba je buď kladná (hodnoty se zpětně přičítají), nebo záporná (hodnoty se odčítají). Kvůli zpětné vazbě může dojít k nestabilitě filtru (to se u FIR nestane), takže se stabilita musí během návrhu kontrolovat. IIR filtry mají i analogový ekvivalent (známé filtry jako Butterworth, Chebyshevm Eliptické…). Tyhle známé filtry jsou např. v MATLAB už k dispozici naimplemetované, jen se musí vhodně zvolit parametry jako řád filtru, mezní kmitočet, přenos v propustném a zádržném pásmu atd. Řád filtru odpovídá stupni polynomu, který ho reprezentuje. Obecně jsou to složité polynomy vyšších řádů, už filtry 2.řádu jsou ale použitelné (není třeba tak vysokých řádů jako u FIR).

Návrh IIR filtru

Toleranční pásmo filtru

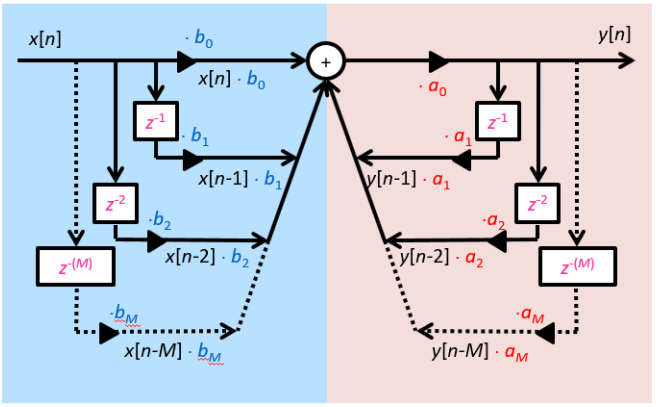
- Zvlnění v propustném pásmu δ_p
- Minimální útlum v zádržném pásmu δ_z
- Mezní (zlomový) kmitočet ω_0 (-3dB)
- Frekvence, kdy je dosažen požadovaný útlum ω_z (strmost filtru)



FIR:
Hledáme odpovídající finitní impulzní odezvu (konvoluční masku)

IIR:
Hledáme polynom předepisující infinitní impulzní odezvu

$\omega \in (0,2)$ – normovaný kmitočet [-]



ıce - IIR

$\frac{\cdot b_1 z^{-1}+\cdots+b_M z^{-M}}{\cdot a_1 z^{-1}+\cdots+a_N z^{-N}}$

olexní polynom

·koliv
ve jmenovateli
ı čitateli
ní fázové zpoždění

Zobrazení pólů a nul polynomu $H[e^{i\omega}]$

- **nuly (o):** hledáme *z*, pro které *H*[*z*]=0
- **póly (×):** hledáme *z*, pro které *H*[*z*]=∞
- MATLAB funkce *zplane*(b,1)

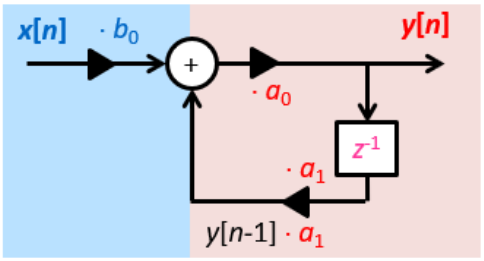
FIR:
- Násobné reálné póly
- Polynom pouze v čitateli
- Lineární fáze (symetrie *h*(*t*))

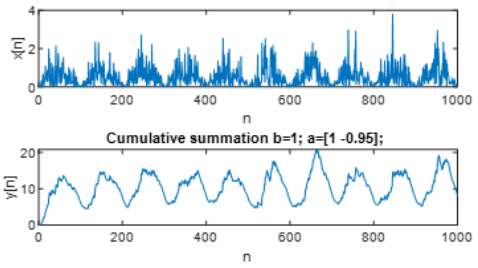
Dopředná vazba:
- FIR (konvoluční automat)
- koeficienty *b*[*k*]
- systém vždy stabilní

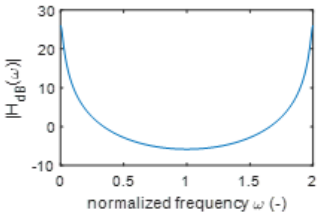
Zpětná vazba:
- IIR (výstup do nekonečna vracen do sčítačky)
- koeficienty *a*[*k*]
- záporná zpětná vazba (**tlumení=filtr**)
- kladná zpětná vazba (**nestabilita !!!**)

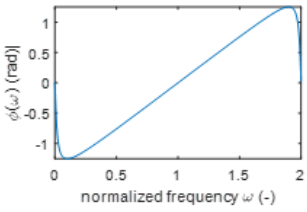
Příklady jednoduchých IIR filtrů

Stabilní sumátor - vlastně ekvivalent matlab funkce cumsum. Chová se jako dolní propusť, takže je vhodný na vyhlazení signálu a hledání obálky signálu.



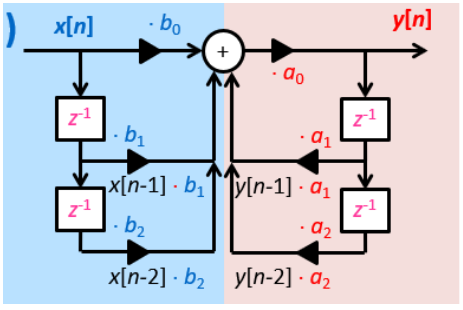




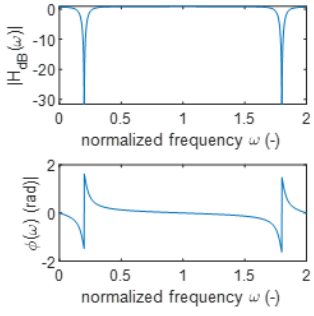


Kumulativní součet $k<1$
- $y[n] = x[n] + ky[n-1]$
- Diferenční rovnice:
- $a_0 y[n] + ky[n-1] = b_0 x[n]$

Bikvadratický filtr (notch) - hodně strmá pásmová zádrž -> vhodné pro filtraci násobků rušení 50 Hz a obecně filtraci určitých specifických frekvencí.



Selektivní filtr (inverzní rezonátor)
- $H[z] = \frac{Y[z]}{X[z]} = \frac{b_0+b_1 z^{-1}+b_2 z^{-2}}{a_0+a_1 z^{-1}+a_2 z^{-2}} \cdot \frac{z^2}{z^2}$
- $H[z] = \frac{Y[z]}{X[z]} = \frac{b_0 z^2+b_1 z+b_2}{a_0 z^2+a_1 z+a_2}$



Použití:
- Velmi ostrá pásmový zádrž:
- Filtrace 50 Hz
- Poloměr pólů nastavuje šířku zádržného pásma
- 2. řád – rychlý výpočet

Bikvadratický rezonátor - vlastně opak notch filtru, hodně strmá pásmová propusť, není moc používaný.

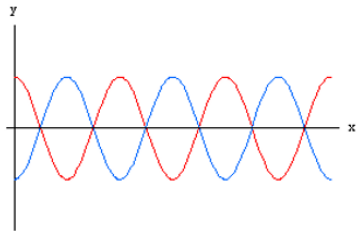
Stabilita filtrů

Pokud omezený vstup vyvolá na výstupu opět omezený výstup, je filtr stabilní. Pokud při omezeném vstupu rostou hodnoty na výstupu do nekonečna, je filtr nestabilní. Nestabilita souvisí s kladnou zpětnou vazbou, ta vlastně říká, že každou hodnotu na výstupu ještě přičtu k následující hodnotě a tak dále. Na výstupu tedy může být větší a větší hodnota a filtr vlastně pořád zesiluje výstup. Kladná zpětná vazba nemusí vždy znamenat nestabilitu, dá se "vyvážit" zápornou zpětnou vazbou.

Jestli je filtr stabilní nebo ne jde poznat podle pólů. To znamená, že se udělá z-transformace a najdou se nulové body jmenovatele. Ty se vykreslí do z- roviny. Pokud jsou póly uvnitř jednotkové kružnice v Z-rovině, je filtr stabilní, pokud jsou vně, je nestabilní.

Oproti FIR má IIR zpětnou vazbu (ZV)

- Tlumená impulzní odezva

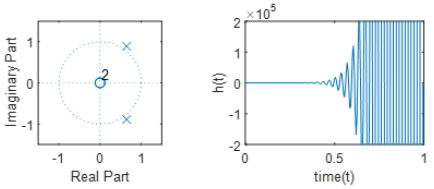


Rezonátor: (+/-) zpětná vazba, ale celkové zesílení <1

Póly uvnitř jednotkové kružnice

Nestabilita filtru

- Netlumená impulzní odezva
- Kladná zpětná vazba
- Výstupní signál je zpětnou vazbou přičítán k dalšímu vzorku (zesílení >1)
- $y[n-1]=y[n-1]+a \cdot y[n]$



Póly vně jednotkové kružnice

Porovnání FIR a IIR

Prakticky se používají spíš IIR než FIR, hlavně kvůli o dost vyšší strmosti.

FIR	IIR
Dopředné vazby (pouze)	**Zpětné** vazby
Konvolučně: $y[n]=x[n]*h[n]$	Rekurzivně: $y[n]=x[n]+a \cdot y[n-1]$
Vždy stabilní (všechny póly nulové)	**Kontrola stability** (póly uvnitř jednotkové kružnice)
Velký řád filtru M_b=stovky, tisíce **Velké zpoždění** **Velká výpočetní náročnost** **Snadná implementace**	**Malý řád** filtru N_a=jednotky **Malé zpoždění** **Malá výpočetní náročnost** **Složitější implementace**
Malá strmost	**Velká strmost**
Symetrická impulzní odezva – **lineární fáze**	Nesymetrická, tlumená – **nelineární fáze**
Snadný návrh (sinc)	**Složitý návrh** (polynomy)
Realizace pouze **DSP**	**DSP i HW** realizace (RLC)