

Multimédia

Gregor Rozinaj, Renata Rybárová, Ivan Minárik, Juraj Kačur

Autoři: Gregor Rozinaj, Renata Rybárová, Ivan Minárik,
Juraj Kačur
Název díla: Multimédia
Přeložil: Jaroslav Svoboda
Vydalo: České vysoké učení technické v Praze
Zpracoval(a): Fakulta elektrotechnická
Kontaktní adresa: Technická 2, Praha 6
Tel.: +420 2 2435 2084
Tisk: (pouze elektronicky)
Počet stran: 131
Vydání: 1.

ISBN 978-80-01-05310-2

Recenzent: Jiří Egrt

Innovative Methodology for Promising VET Areas
<http://improvet.cvut.cz>



**Program
celoživotního
učení**

Tento projekt byl realizován za finanční podpory Evropské unie.
Za obsah publikací odpovídá výlučně autor. Publikace (sdělení) nereprezentují názory Evropské komise a Evropská komise neodpovídá za použití informací, jež jsou jejich obsahem.

VYSVĚTLIVKY



Definice



Zajímavost



Poznámka



Příklad



Shrnutí



Výhody



Nevýhody

ANOTACE

Tento výukový modul obsahuje základní informace o multimédiích. V první části jsou uvedeny základní znalosti o analogových a digitálních signálech, digitalizaci signálů, časové a frekvenční reprezentaci signálů. Další část obsahuje informace o zpracování signálů, analogových a digitálních technologiích, filtrech a popis telekomunikačního kanálu. Na závěr se modul věnuje kompresním technikám audio a video signálů a samotné aplikaci teorie zpracování signálů v multimédiích: syntéze řeči, rozpoznávání řeči a obrazu.

CÍLE

Hlavním cílem tohoto výukového kurzu je seznámení studentů se základy zpracování signálů, zejména s těmi technikami, které se využívají při zpracování multimédií. Studentům jsou představeny základní principy Fourierovy transformace, digitálních filtrů, lineárních systémů, kompresních technik a praktické použití všech vědomostí v moderní vědě.

LITERATURA

- [1] DÚHA, J., GALAJDA, P., KOTULIAK, I., LEVICKÝ, D., MARCHEVSKÝ, S., MIKÓCZY, E., PODHRADSKÝ, P. a kol. *Multimedia ICT technologies, network platforms and multimedia services*, Vydavateľstvo STU Bratislava, 2005, ISBN 80-227-2310-X.
- [2] GAMEC, J. *Spracovanie multimediálnych signálov*, publikované v rámci projektu ESF NGN, kód projektu v ITMS: 13120110126, Bratislava, 2007.
- [3] TALAFOVÁ, R. – ROZINAJ, G. – CEPKO, J. – VRABEC, J. *Multimedia SMS Reading in Mobile Phone* In: INTERNATIONAL JOURNAL of MATHEMATICS AND COMPUTERS IN SIMULATION, Issue 1, Volume 1, ISSN 1998-0159, 2007.
- [4] KOTULIAKOVÁ, J. – ROZINAJ, G. *Číslicové spracovanie signálov I*, FABER Bratislava, 1996.
- [5] KOTULIAKOVÁ, J. – ROZINAJ, G. – POLEC, J. – PODHRADSKÝ, P. a kolektív. *Číslicové spracovanie signálov II*, FABER Bratislava, 1997
- [6] HARDESTY, L. *Explained: The Discrete Fourier Transform*, 2009, <http://web.mit.edu/newsoffice/2009/explained-fourier.html> [online]
- [7] MINÁRIK, I. *Coding of audio signals at low speed*, Diploma Thesis, 2011
- [8] MARMOL, F.G., et al. *ANALYSIS: State of The Art on Identity, Security and Trust*, Deliverable D3.1, HBB-Next FP7-ICT-2011-7, 2012

- [9] DEVENTER, O. et al. *ANALYSIS: Multi-User, Multimodal & Context Aware Value Added Services*, Deliverable D5.1, HBB-Next FP7-ICT-2011-7,2012
- [10] LEVICKÝ, D., RIDZOŇ, R. *Multimédiá a multimedialne technológie*, publikované v rámci projektu ESF NGN, kód projektu v ITMS: 13120110126, Vydalo Vydavateľstvo STU v Bratislave v spolupráci s AGROGENOFOND Nitra, 2007, ISBN 978-80-227-2604-7
- [11] PODHRADSKÝ, P. *Fourierov rad a Fourierova transformácia*, publikované ako študijný materiál pre predmet inžinierskeho štúdia "Analogové a digitálne signály a sústavy I", študijný program Telekomunikácie, FEI STU Bratislava, 2003

Obsah

| | | |
|----------|--|------------|
| 1 | Signály | 8 |
| 1.1 | Co je signál? | 8 |
| 1.2 | Důležité signály | 12 |
| 1.3 | Digitalizace analogového signálu | 20 |
| 1.4 | Modulace | 24 |
| 2 | Časová a frekvenční reprezentace | 30 |
| 2.1 | Fourierova transformace | 30 |
| 2.2 | Diskrétní Fourierova transformace | 34 |
| 2.3 | Spektrum | 36 |
| 2.4 | Ortogonální transformace | 43 |
| 3 | Analogové a digitální technologie | 48 |
| 3.1 | Multiplexování | 48 |
| 3.2 | Lineární diskrétní časově – invariantní systém | 52 |
| 3.3 | Filtry | 59 |
| 4 | Telekomunikační spoj | 69 |
| 4.1 | Telekomunikační spoj | 69 |
| 5 | Kompresní metody | 72 |
| 5.1 | Kompresce audio signálů | 75 |
| 5.2 | Kompresce řeči | 81 |
| 5.3 | Kompresce statických obrázků | 87 |
| 5.4 | Kompresce videa | 94 |
| 6 | Zpracování multimédií | 103 |
| 6.1 | Syntéza řeči | 103 |
| 6.2 | Rozpoznávání obrazu | 106 |
| 6.3 | Animace obličejů | 109 |
| 6.4 | Rozpoznávání řeči | 112 |
| 6.5 | Multimodální rozhraní | 124 |
| 7 | Příklady v Matlabu | 126 |
| 7.1 | Příklady v Matlabu | 126 |

Pod termínem *Multimédia* rozumíme integraci textu, obrazu a zvuku za účelem zprostředkování informace. Každá z uvedených třech modalit má více reprezentací. Pod textem rozumíme, kromě běžného textu, také hypertext, tabulky, webové stránky apod. Obrazová modalita má pestrou škálu skupin jako statický obraz, dynamický obraz, grafika, animace, video atd.

Multimédia vnímáme našimi smysly. Kromě zraku a sluchu, kterými vnímáme zatím zmíněné modality, vnímáme svět také dalšími třemi smysly – čich, chuť a hmat, a proto v obecnosti zahrnujeme do termínu multimédia signály, které působí na všech pět smyslů. Multimediální komunikaci často omezujeme na komunikaci, kde potřebujeme alespoň dva smysly, i když se v multimediálních aplikacích často zabýváme též pouze jednosmyslové komunikaci.

Multimediální aplikace je zpravidla podmíněna možností interakce, tj. člověk je schopen řídit chod multimediální aplikace. Z tohoto pohledu např. klasickou televizi nepovažujeme za multimediální signál, neboť program neumíme řídit.

Médium je prostředek, kterým se informace vyjadřuje, vnímá ukládá nebo přenáší.

Multimediu je médium založené na více modalitách.

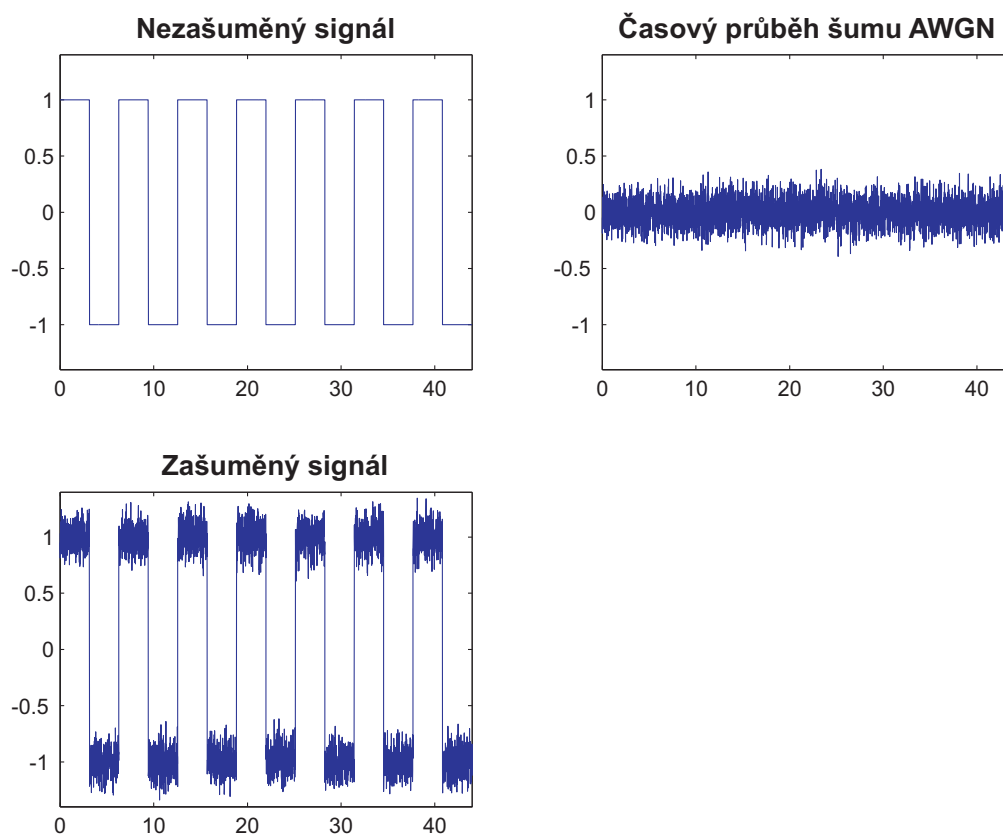
1 Signály

1.1 Co je signál?



Význam slova signál v tomto kurzu označuje funkci, která komplexně a úplně popisuje chování určitého jevu.

Signál ve fyzickém světě je v zásadě jakákoliv veličina, která vykazuje změnu v čase (například hlas) nebo v prostoru (obraz), která může poskytnout informaci o stavu fyzického systému, nebo přenášet zprávu mezi pozorovateli. Skutečný signál je vždy smíšený také s šumem.



Skutečný signál – signál+šum

V oblasti elektrotechniky a zpracování signálu rozlišujeme dva druhy signálů – analogový a digitální signál.

Analogový signál



Analogový signál je jakýkoliv spojitý signál, přičemž může nabývat teoreticky libovolné hodnoty.



Analogový signál může nabývat libovolné hodnoty v čase a amplitudě. Reprezentuje chování charakteristické vlastnosti nebo jevu v čase.

Typickým analogovým signálem je elektrický signál, nebo teplota měnící se v čase. Analogový signál může být deterministický (umíme ho přesně definovat, např. matematickým vzorcem), nebo stochastický (jeho průběh je náhodný a neumíme ho předpovídat).

Digitální signál



Digitální signál je reprezentován sekvencí diskrétních (obvykle předdefinovaných) hodnot.



Digitální signál může mít jen konečné množství vzorků v daném časovém úseku. Jednoduchý příklad jak získat diskrétní signál, je vzorkování spojitého (analogového) signálu. Příkladem digitálního signálu je například teplota vzduchu měřená každých 5 minut, nebo nuly a jedničky používané v počítačích.

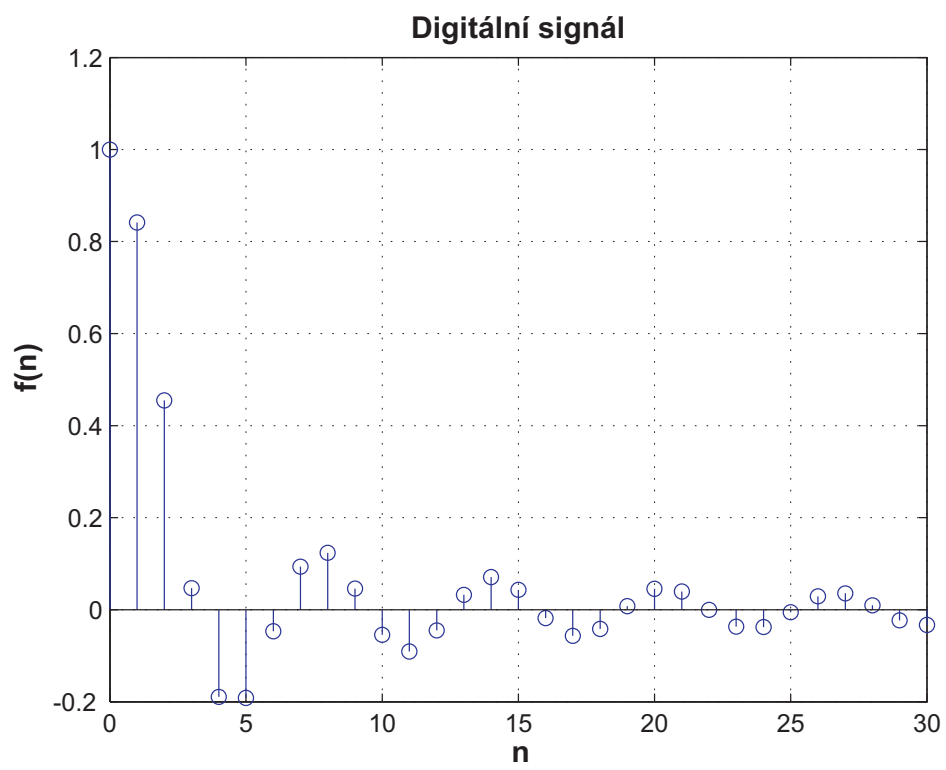
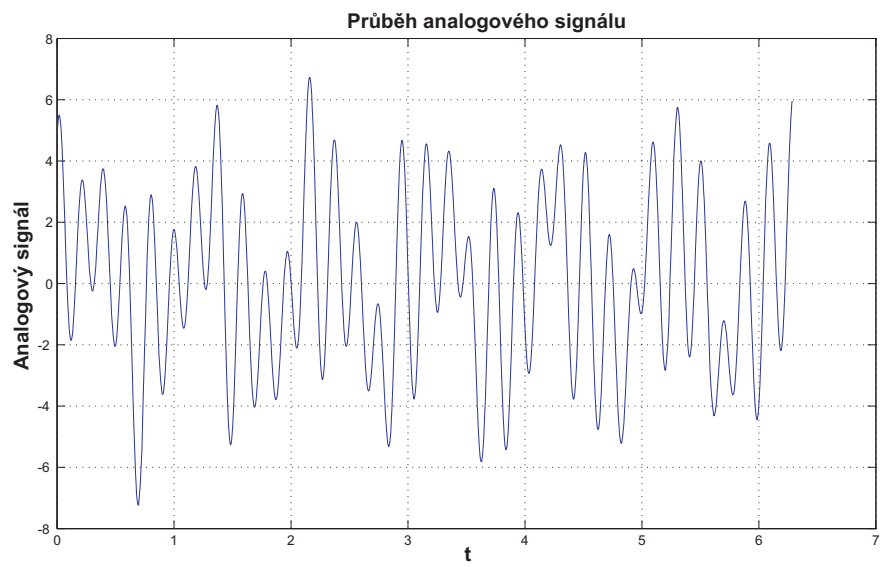
Všechny procesy v přírodě jsou analogové (představte si například graf zobrazující teplotu měnící se v čase, nebo rychlost auta v čase).

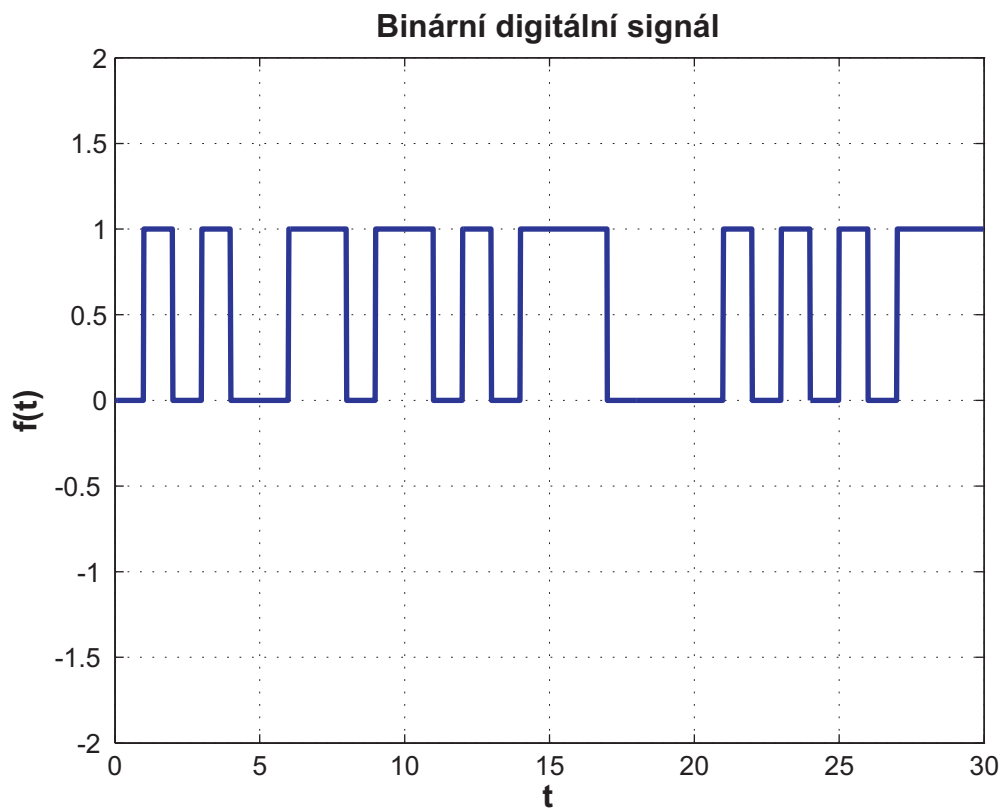
+

Hlavní výhodou zpracování analogového signálu je skutečnost, že se při tom neztrácí žádná informace. Avšak digitální signály jsou mnohem jednodušeji zpracovatelné (například hudební CD je digitální – a může být velmi jednoduše konvertovatelné na MP3). Digitální signál je též méně náchylný na degradaci šumem.

–

Analogový signál je složitější na zpracování a namáhavěji se s ním pracuje (představte si vinylovou desku, kvalita je sice lepší než CD, ale zkuste si písničku z desky dát do telefonu). Výhodou analogového signálu je však to, že se neztrácí informace jako u digitálního signálu procesem vzorkování a kvantizace (představte si tabulku hodnot s teplotami za jeden den – je to jen 24 číselných hodnot).





Příklad analogového a digitálního signálu

1.2 Důležité signály

V předcházející části byl vysvětlen význam slova signál. Nebyly však zmíněny případy signálů s více než jednou dimenzí. V této části budou uvedeny nejdůležitější signály v oblasti digitálního zpracování signálu a multimédií.

Jednorozměrné signály



Signál, který je funkcí jedné nezávislé proměnné, se nazývá jednorozměrný signál. Obvykle je proměnnou čas t (například $f(t)=5t$), v případě diskrétních signálů, je obvykle místo času diskrétní proměnná n , reprezentující číslo vzorku (například $f(n) = n+1$).

V následujících definicích a vzorcích x reprezentuje množinu reálných čísel $\{R\}$ a n reprezentuje přirozená čísla $\{N\}$.

Diracova delta funkce δ je všeobecná funkce reálných čísel, kde každá hodnota kromě nuly je rovná nule. Delta funkce se někdy znázorňuje jako nekonečně vysoká a nekonečně tenká čára vycházející z nuly. Kdybychom vypočítali plochu, kterou tento impuls zabírá, dostaneme číslo 1. V oblasti zpracování signálu je tato funkce též často známa pod názvem jednotkový impuls.

Matematická definice:

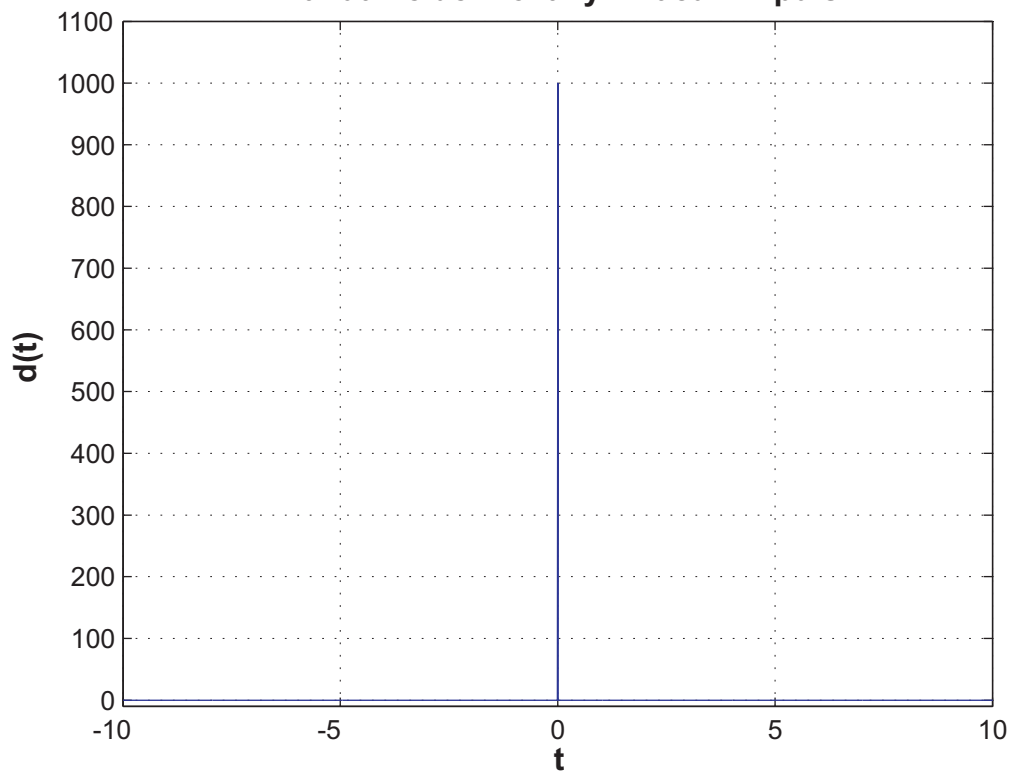
$$\delta(x) = \begin{cases} +\infty & x = 0 \\ 0 & x \neq 0 \end{cases}$$

někdy postačí jen redukováná definice

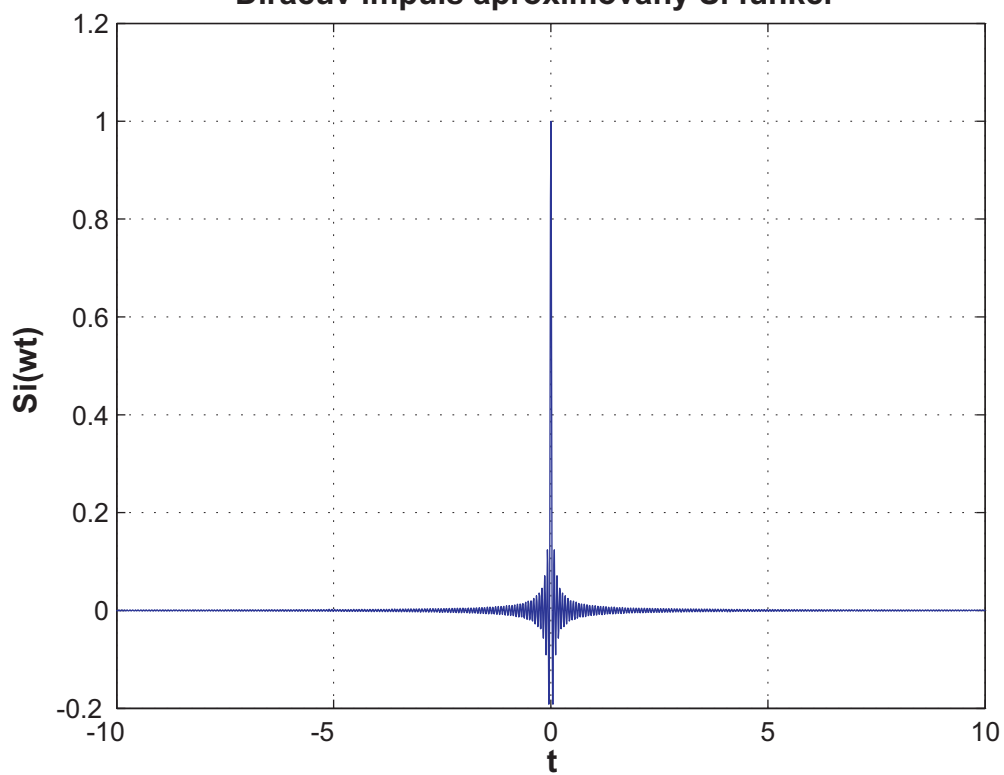
$$\int_{-\infty}^{\infty} \delta(x) dx = 1.$$

Na obrázku níže je zobrazena ideální a aproximovaná Diracova delta funkce. Aproximovaná Diracova funkce je zobrazena pro lepší vysvětlení, jak můžeme v reálném světě získat delta funkci.

Manuálně definovaný Diracův impuls



Diracův impuls aproximovaný Si funkcí

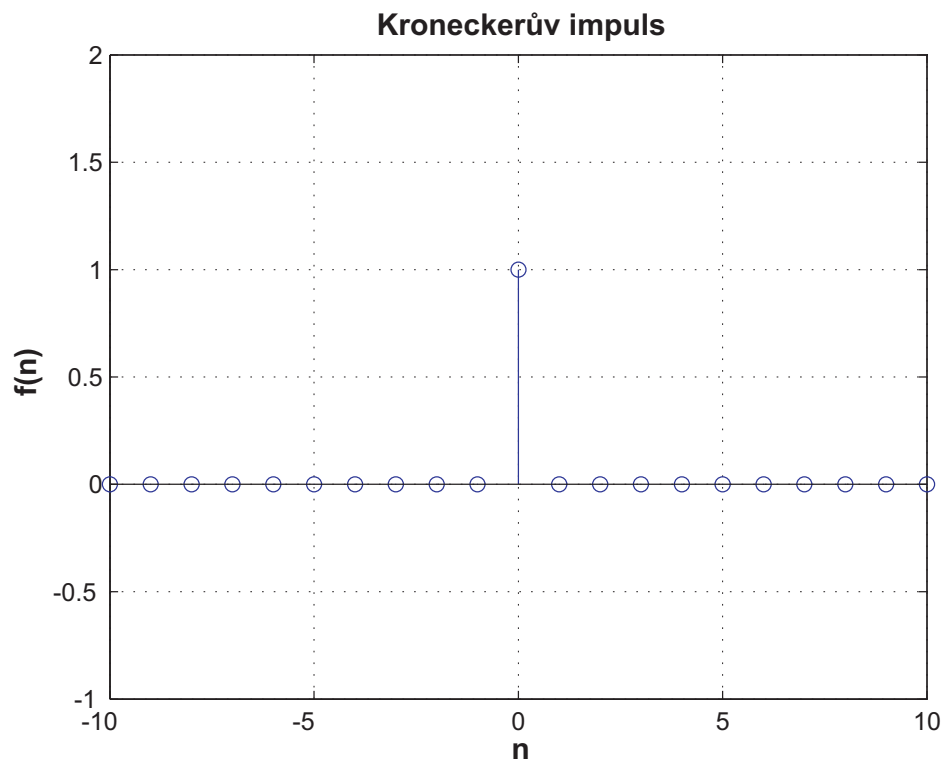


Diracova delta funkce – ideální a přibližná funkcí $sinc()$.

V diskretním světě je ekvivalentem Diracovy delta funkce tzv. Kroneckerova delta funkce. Ve světě digitálního zpracování signálu je často známa pod jednodušším názvem impuls, nebo jednotkový impuls. Výstup nebo odpověď, kterou dostaneme od digitálního elementu (funkce, součástky), nazýváme odpověď na jednotkový impuls.

Matematická definice:

$$\delta(n) = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$



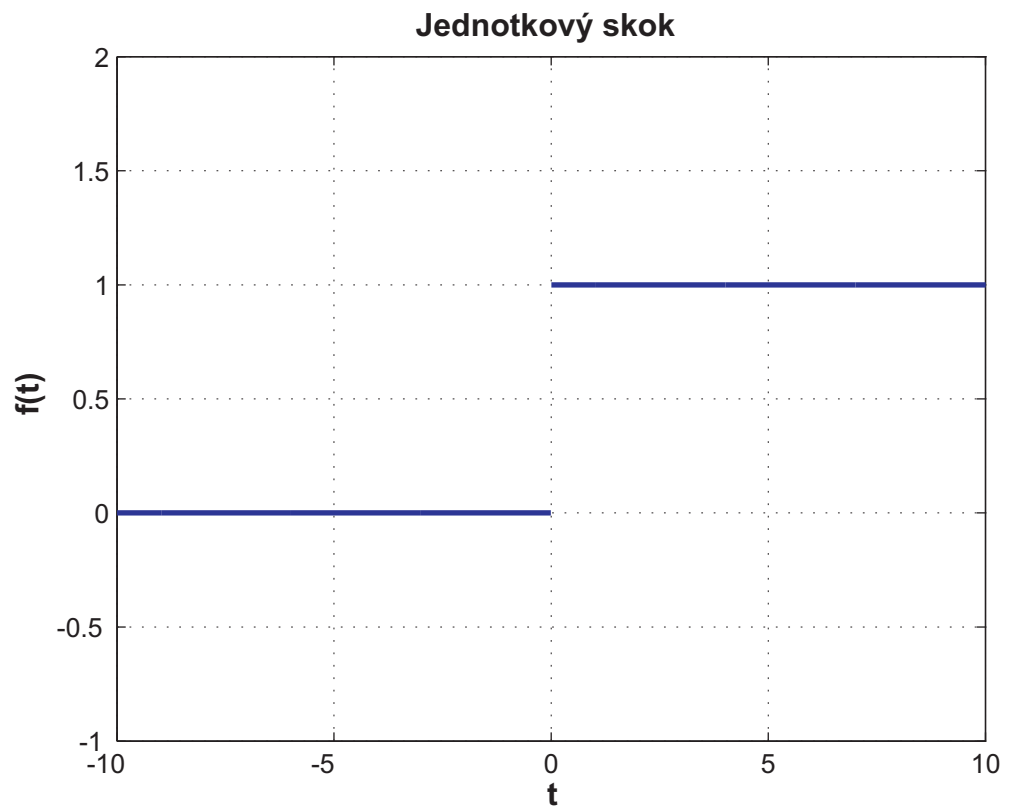
Kroneckerova delta funkce

Funkce jednotkového skoku, obvykle označená jako u , je nespojitou funkcí, jejíž hodnota je nulová pro všechny záporné hodnoty proměnné a jedna pro všechny pozitivní hodnoty. Funkce reprezentuje signál, který se zapne ve specifikovaném čase a zůstane donekonečna zapnutý. Funkce jednotkového skoku je integrálem Diracovy delta funkce.

$$u(x) = \int_{-\infty}^x \delta(s) ds$$

Matematická definice:

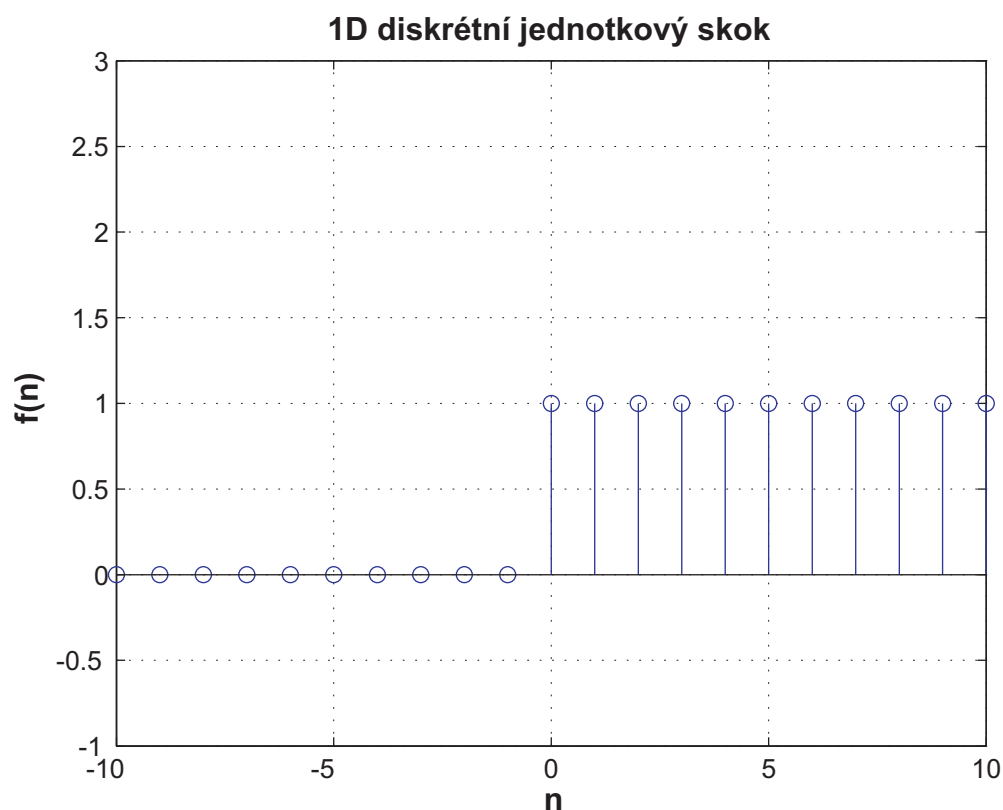
$$u(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases}$$



Jednotkový skok (spojitý)

Diskrétní forma jednotkové funkce:

$$u(n) = \begin{cases} 0 & n < 0 \\ 1 & n \geq 0 \end{cases}$$



Jednotkový skok (diskrétní)

Další množinou signálů jsou periodické signály. Periodické funkce (které popisují opakující se signály) jsou funkce, které opakují svoje hodnoty v pravidelných intervalech. Patří sem všechny trigonometrické funkce (sinus, kosinus, tangens, kotangens – všechny mají periodu 2π). Jestliže periodu označíme P , potom matematická definice periodické funkce je:

$$f(x) = f(x + P)$$

Dvojměrné signály

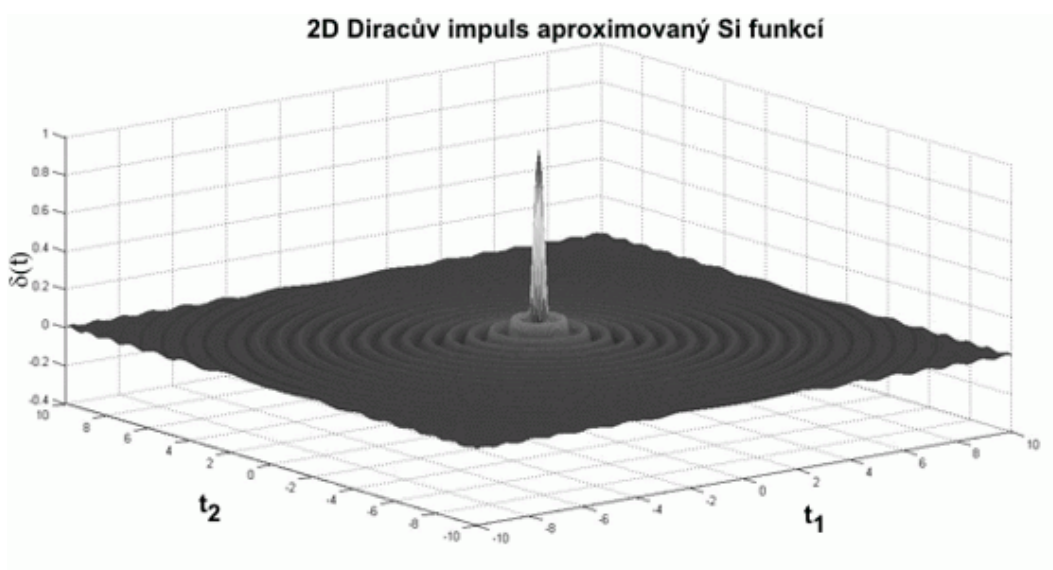
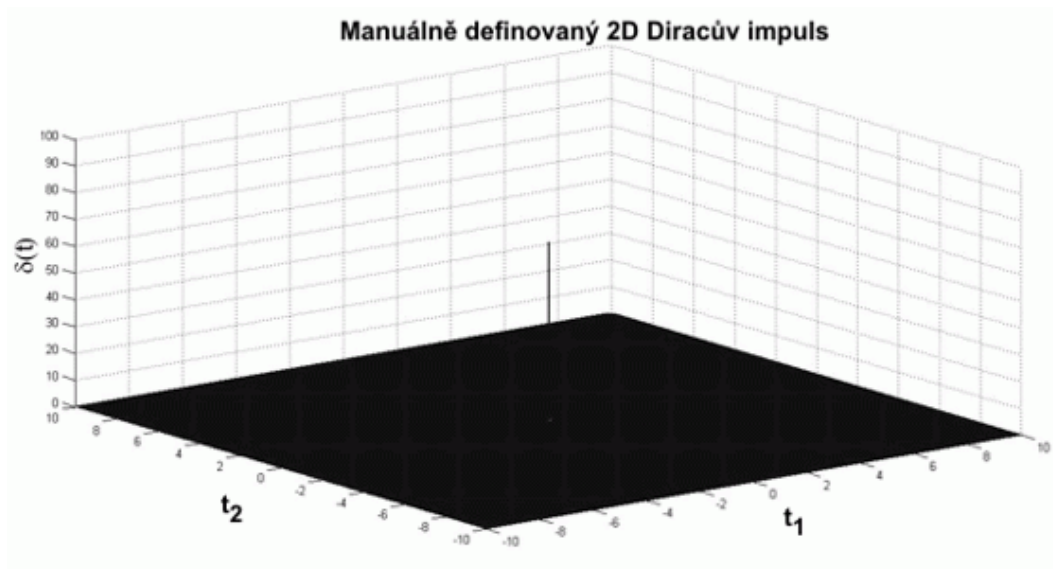


Signál, který je funkcí dvou nezávislých proměnných, se nazývá dvojměrný signál. Typickým příkladem dvojměrného signálu je obrázek. Obrázek se skládá ze světelné a luminiscenční složky. Také 2D obraz může být spojitý, například klasická fotografie nebo obraz, nebo diskrétní, například digitální fotografie.

Všechny důležité signály uvedené pro jednorozměrné signály je možno definovat též pro dva rozměry. Uvádíme jen matematické definice.

Diracova delta funkce

$$\delta(x_1, x_2) = \begin{cases} +\infty & x_1, x_2 = 0 \\ 0 & x_1, x_2 \neq 0 \end{cases}$$

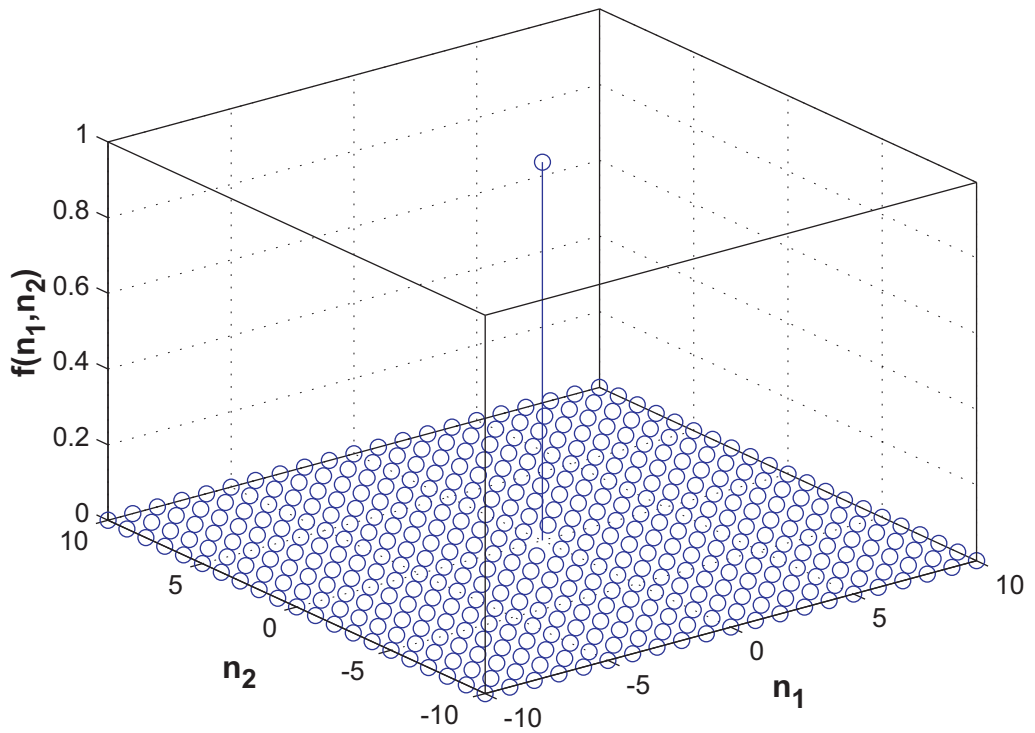


2D Diracova delta funkce – ideální a aproximovaná funkcí *sinc()*.

Kroneckerova delta funkce

$$\delta(n_1, n_2) = \begin{cases} 1 & n_1, n_2 = 0 \\ 0 & n_1, n_2 \neq 0 \end{cases}$$

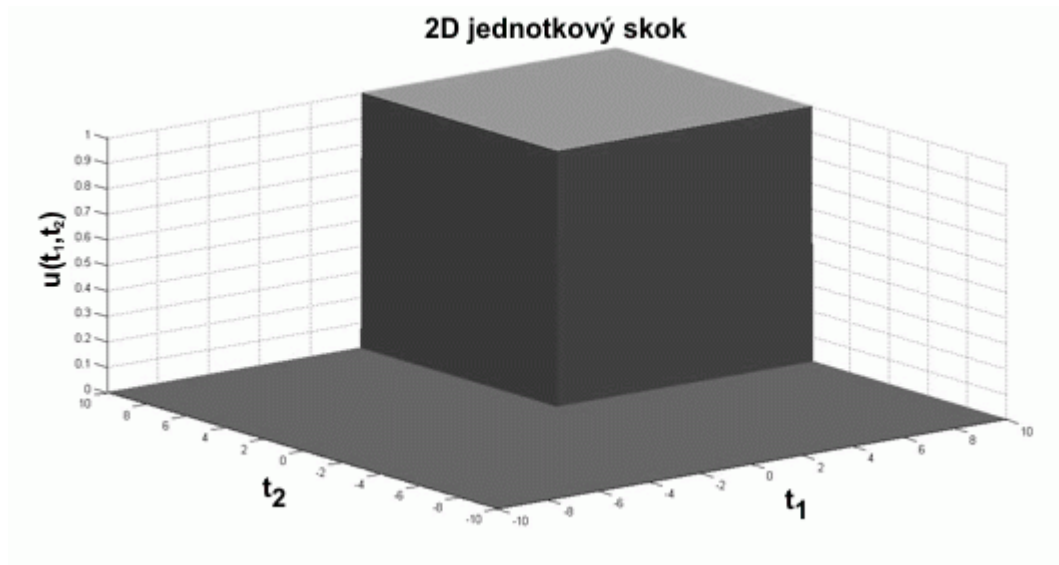
2D Kroneckerův impuls



2D Kroneckerova delta

2D jednotkový skok (spojitý)

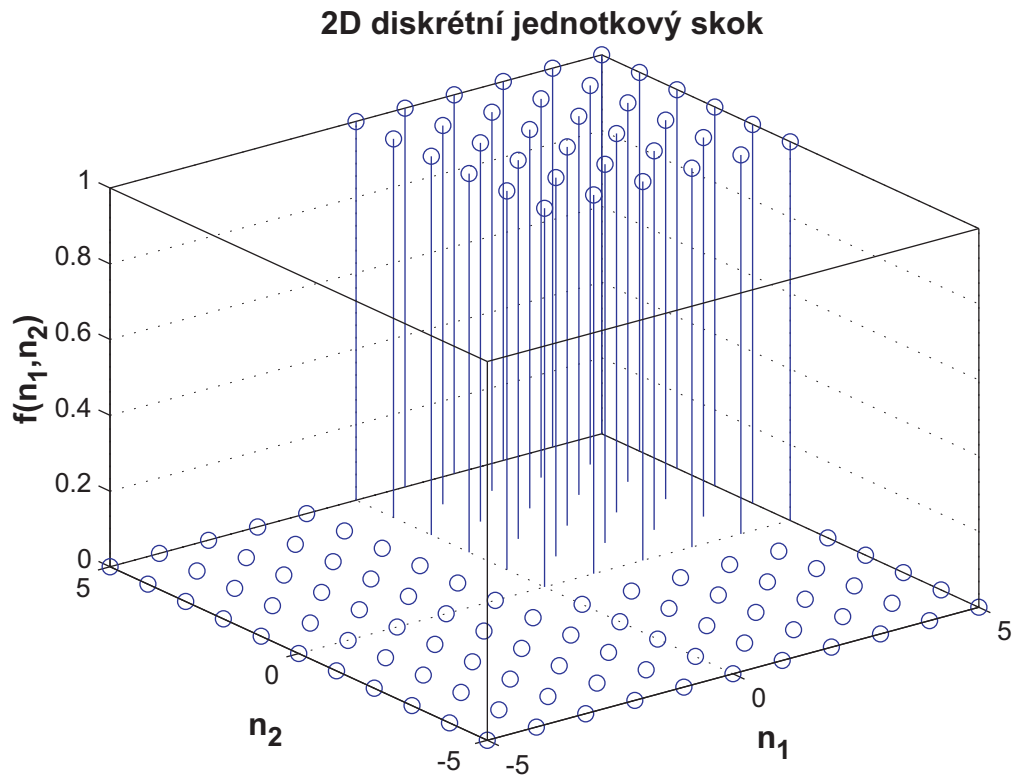
$$u(x_1, x_2) = \begin{cases} 0 & x_1, x_2 < 0 \\ 1 & x_1, x_2 \geq 0 \end{cases}$$



2D Jednotkový skok (spojitý)

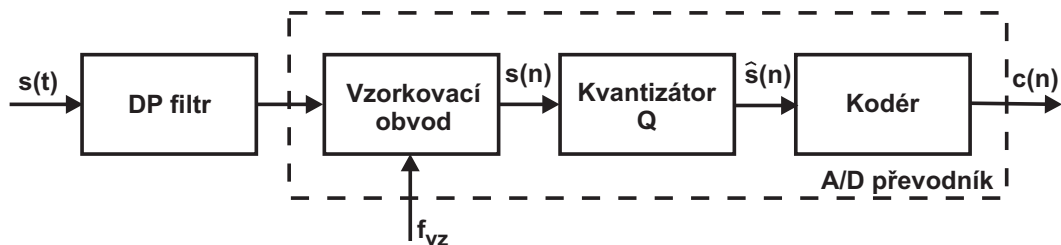
Jednotkový skok (diskrétní)

$$u(n_1, n_2) = \begin{cases} 0 & n_1, n_2 < 0 \\ 1 & n_1, n_2 \geq 0 \end{cases}$$



1.3 Digitalizace analogového signálu

Digitalizace analogového signálu je proces, ve kterém je analogový signál (reprezentující například zvuk, video, obraz,...) transformovaný do digitální formy. Signál je nejprve navzorkován, kvantován a kódován. Výsledkem je sekvence (posloupnost) binárních čísel, které se mohou dále zpracovávat.



Digitalizace analogového signálu (PCM metoda)

Mezi základní metody kódování multimediálních signálů v časové oblasti, používané v multimediálních telekomunikacích je možné zařadit:

- pulsně kódovou modulaci (PCM),
- diferenční PCM (DPCM),
- adaptivní DPCM (ADPCM).

Signál se po omezení DP filtrem (antialiasing filtr) podrobí vzorkování ve vzorkovacím obvodu, čímž se získá posloupnost jeho vzorků.



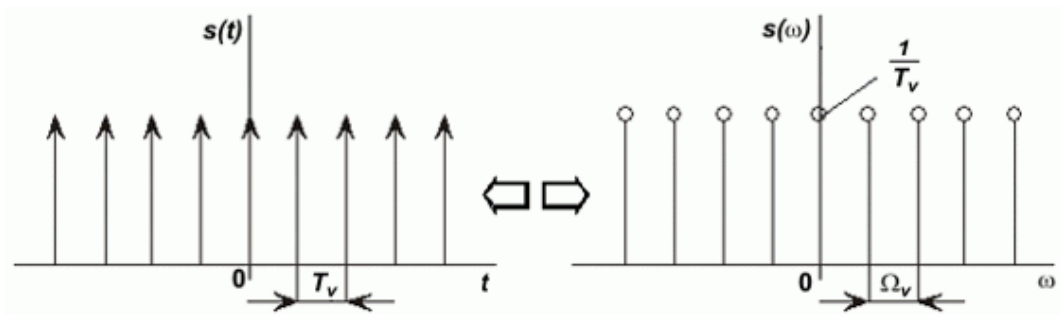
Vzorkování je převod spojitého signálu na diskrétní signál. Vzorkuje se v definovaných časových okamžicích, takže vzorky odpovídají hodnotám v čase a/nebo prostoru. Vzorkovací rychlost je určena ze vzorkovacího teorému (známého jako Shannon-Kotělnikovův teorém), tj. vzorkovací frekvence musí být minimálně dvakrát větší než je hodnota nejvyšší frekvence signálu.

Hodnota vzorkovací frekvence F_{vz} je dána šířkou pásma vzorkovaného signálu, např. pro telefonní hovorový signál s frekvenčním pásmem 300 - 3400 Hz (do 4 kHz) se volí $F_{vz} = 8$ kHz.

Vzorkovací teorém může být matematicky vyjádřen jako $T_{vz} \leq \frac{1}{2F_m}$.

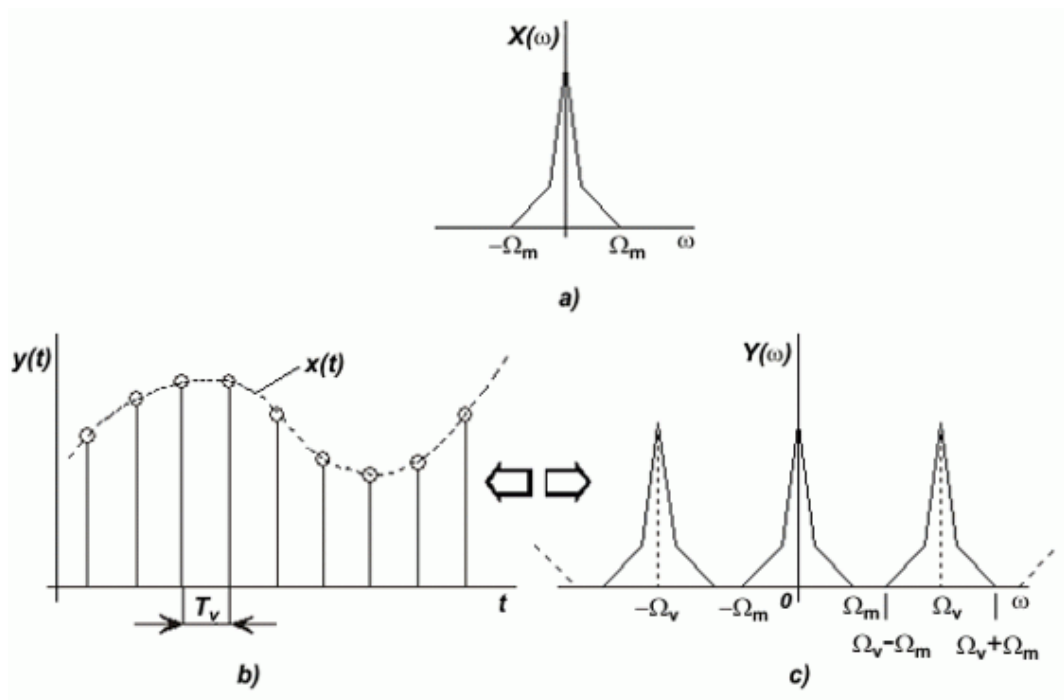
Minimální vzorkovací frekvence $F_{vz\min} = 2F_m$ se nazývá Nyquistova frekvence.

Vzorkovaný signál, označme ho jako $y(t)$, je možné vyjádřit jako součin průběhu $x(t)$ a vzorkovací funkce $s(t)$, která je reprezentována nekonečnou sérií jednotkových (Diracových) impulsů. Spektrum vzorkovací funkce, jejíž impulsy jsou od sebe vzdálené o dobu $t = T_{vz}$, je též nespojitý, frekvenčně neomezený a vzdálenost jednotlivých složek na frekvenční ose je Ω_{vz} .



Vzorkovací funkce v časové a frekvenční oblasti

Spektrum $Y(\omega)$ vzorkovaného signálu $y(t)$ je výsledkem konvoluce spekter signálů $X(\omega)$ a $S(\omega)$.



a) Spektrum nevzorkovaného signálu $X(\omega)$ b) Časový průběh ideálně vzorkovaného signálu $y(t)$ c) Spektrum ideálně vzorkovaného signálu $Y(\omega)$

Jestliže pro “periodu“ opakování spekter platí $\Omega_{vz} < 2\Omega_m$, dochází k překrývání složek spektra. Tento jev se nazývá aliasing. K překrývání spekter dochází též v důsledku nikoliv konečné šířky spektra signálu $x(t)$. To způsobí, že rekonstruovaný signál ze vzorků je jiný než původní signál.

Dalším krokem digitalizace je kvantování. Vzorky signálu jsou nahrazeny příslušnými kvantizačními úrovněmi (pevně určenými hodnotami jako například přirozená čísla), čímž se získá posloupnost kvantovaných vzorků signálu. Kvantizační úrovně se získají dělením amplitudy na menší intervaly. Délka intervalu se nazývá kvantovací krok. V případě, že všechny kroky kvantování mají stejnou délku, hovoříme o lineárním kvantování, v opačném případě jde o nelineární kvantování.

-
- Největší nevýhodou tohoto procesu je tzv. kvantizační chyba nebo kvantizační šum. Je to rozdíl mezi analogovým vstupem do analogovo-digitálního převodníku (**ADC** *analog-digital convertor*) a digitálním výstupem. Chyba je nelineární a závislá na konkrétním signálu. Chyba kvantování způsobuje problém též při zpětné konverzi digitálního signálu na analogový. Proto signál nikdy není konvertován do identické podoby, je pouze aproximován z kvantizačních úrovní.
-

Dalším krokem digitalizace je proces kódování.



V tomto procesu se přiřadí k danému kvantovanému vzorku binární kód, čímž vzniká posloupnost kódových slov.

PCM metoda je mezinárodní standard pro kódování a přenos multimediálních signálů. Základní princip této metody je znázorněn na obrázku níže.

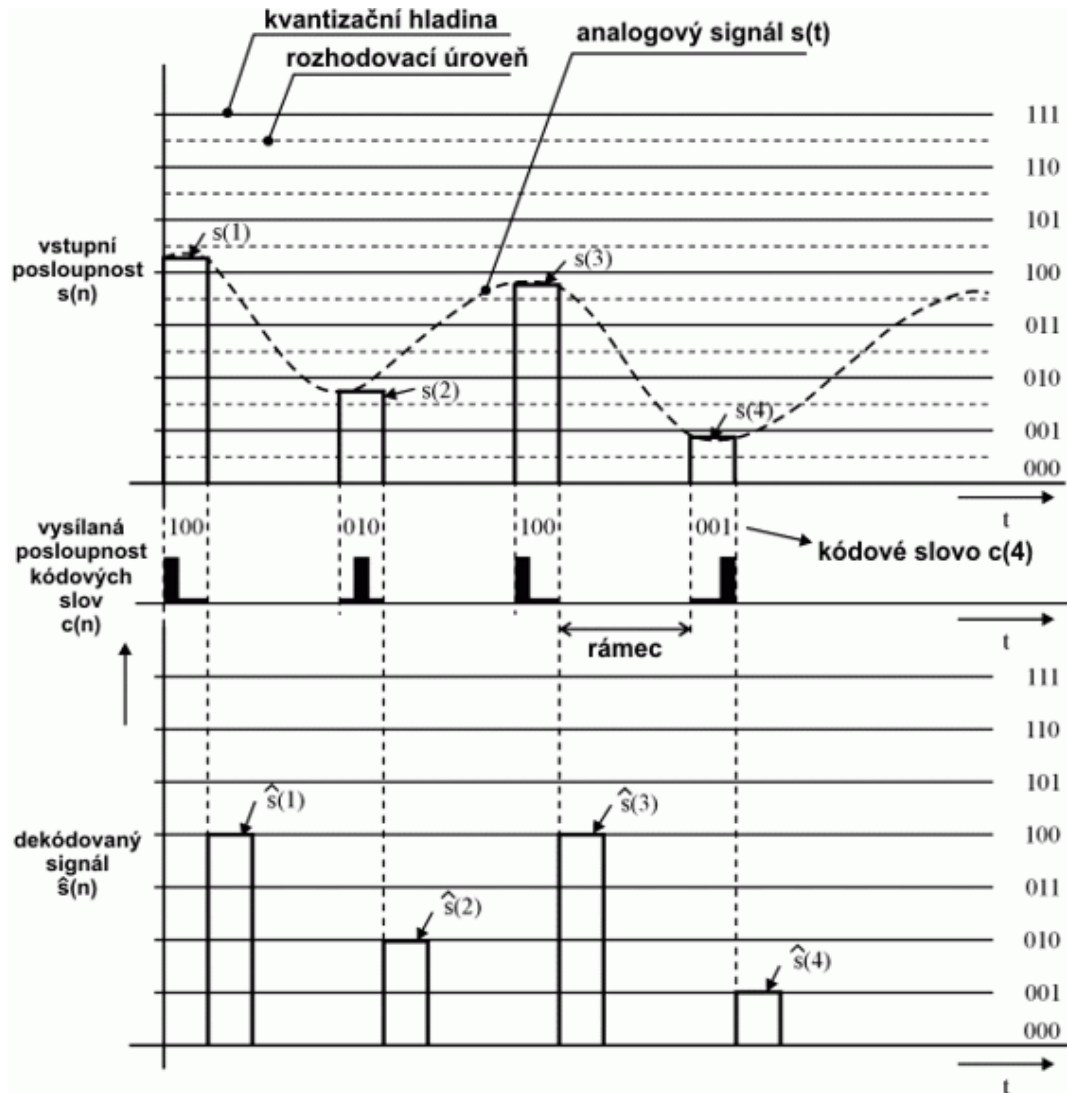


Prvé systémy na bázi PCM používaly 7 bitovou délku kódového slova N , tj. počet kvantizačních úrovní byl 128. Když předpokládáme, že vzorkovací frekvence $F_{vz} = 8$ kHz a $N = 8$, potom potřebná přenosová rychlost pro přenos hovoru v telefonním pásmu je $8 \cdot 10^3 \cdot 8 = 64$ kbit/s.

- + Metoda PCM v porovnání s analogovými metodami přenosu signálů má výhodu zejména v odolnosti přenášeného digitálního signálu vůči rušení.
-

- Na druhé straně, nevýhodou je požadavek na širší frekvenční pásmo pro přenos signálu.
-

Časové průběhy signálu kódovaného PCM jsou znázorněny na obrázku.



Časové průběhy v PCM metodě kódování multimediálních signálů

Lineární PCM používá stejné kvantizační kroky v celém pracovním rozsahu kvantování. Z toho vyplývá, že rozsah (dynamika) zpracovatelného vstupního signálu závisí na počtu a velikosti kvantizačních kroků. Počet kvantizačních kroků (stupňů) při stejném rozsahu vstupního signálu ovlivňuje též velikost kvantizační chyby. Při větším počtu kvantizačních kroků se sice zmenšuje velikost kvantizační chyby, ale zvyšují se nároky na přenosovou rychlost. Tento přístup má tedy omezení, které je možné odstranit nelineárním rozložením kvantizačních kroků, což je podstatou nelineární PCM.

Nelineární PCM používá nelineární rozložení kvantizačních kroků tak, že jejich velikost se směrem k vyšším amplitudám vstupního signálu zvětšuje. Modifikovaným přístupem je komprese dynamiky vstupního signálu na vysílací straně a expanze dynamiky na přijímací straně. Kompresor na vysílací straně malé hodnoty vzorků zesílí a velké zeslabí. Expandér na přijímací straně naopak vrátí vzorkům původní rozsah a správný poměr jejich hodnot.

Dekódování je stejný proces jako kódování, ale v opačném pořadí. Výstup dekodéru je posloupnost kvantovaných vzorků.

1.4 Modulace

Analogový signál, tak jak byl definován v první části, je proměnlivý signál spojitý v čase i amplitudě. Kdybychom měli například graficky znázornit střídavý proud, vypadal by jako vlna, s napětím měnícím se ze záporných na kladné hodnoty. Existují tři faktory, které je potřebné posoudit: frekvence, amplituda a fáze.



Frekvence je rychlost jakou proud osciluje nad a pod nulovou úroveň, resp. střídá svoji polaritu.

Když proud vystoupí nad nulovou hodnotu, potom jde zpět pod nulovou hodnotu a následně se vrátí na nulu, říkáme, že vykonal jeden „cyklus“. Frekvence se udává v počtech těchto cyklů za jednu sekundu a její jednotkou je Hertz (Hz). Takže když má analogový signál 500 cyklů za sekundu, říkáme, že frekvence je 500 Hertzů (500 Hz).



Za amplitudu je považována hodnota největší výšky a hloubky grafické vlny. Jinak řečeno je to maximální absolutní hodnota signálu.

Postupně, jak se analogový signál šíří prostorem, amplituda vlny se zmenšuje. Tato vlastnost se nazývá útlum. Analogové vlny nejsou až tak ovlivněné problémy s útlumem, avšak někdy musí být zesilovány. Amplituda analogových vln se měří ve wattech, ampérech nebo voltech. Jednotkou měření však často bývá decibel. Decibel (dB) umožňuje jednoduché porovnání dvou různých hodnot pomocí podílu (v tomto případě jde o podíl amplitud signálu). Decibel je bezrozměrná logaritmická jednotka.



Fáze popisuje rozdíl mezi počátečním stavem cyklu jednoho signálu v porovnání s počátečním stavem cyklu druhého signálu, tj. popisuje jejich posun. Jeden signál je vždy referenční, druhý signál je fázově posunutý.

Fázově posunutý signál vzniká malým zpožděním signálu, které způsobí, že vrcholy a hloubky nejsou synchronní ve srovnání s referenčním signálem. Úroveň nesynchronizovanosti je měřena ve stupních. Když je fázový posun signálu 180 stupňů, znamená to, že posunutý signál začíná tam, kde se referenční signál po sestupu z vrcholu dotýká nuly.

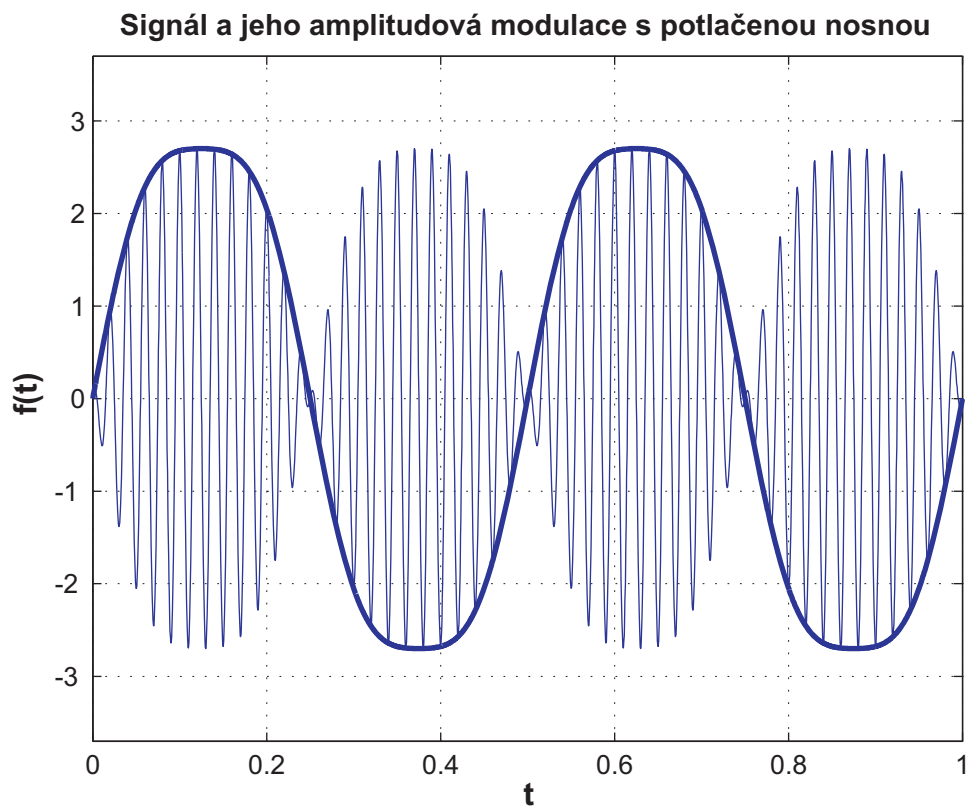
Je důležité dobře znát amplitudu, frekvenci i fázi, když víme, že signál dokáže přenášet informace právě prostřednictvím změny těchto vlastností v čase.



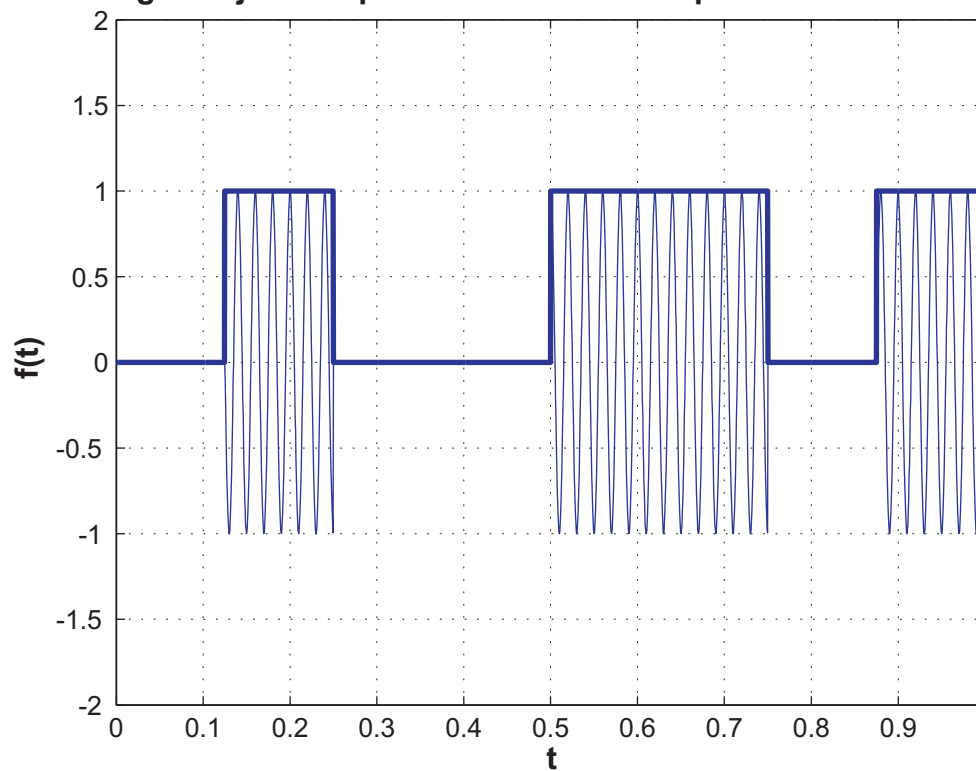
Modulace je proces vkládání zprávy, například digitálních bitů nebo analogového audio signálu, do jiného signálu, který může být fyzicky vysílaný.

Jedním z důvodů modulace je možnost přenosu více signálů v jednom kanálu s omezenou šířkou pásma. Každý signál je modulovaný tak, aby se „vešel“ do přenosového subpásma. Výhodou tedy je, že po seskupení těchto signálů (což je možné, když každý má individuální subpásmo) můžeme použít pouze jednu přenosovou cestu (jehož pásmo je složeno ze subpásem).

Při analogové modulaci se modulace aplikuje kontinuálně podle spojitého analogového signálu. Existuje mnoho druhů analogových modulací, nejjednodušší jsou *amplitudová modulace (AM)*, *fázová modulace (PM)* a *frekvenční modulace (FM)*.

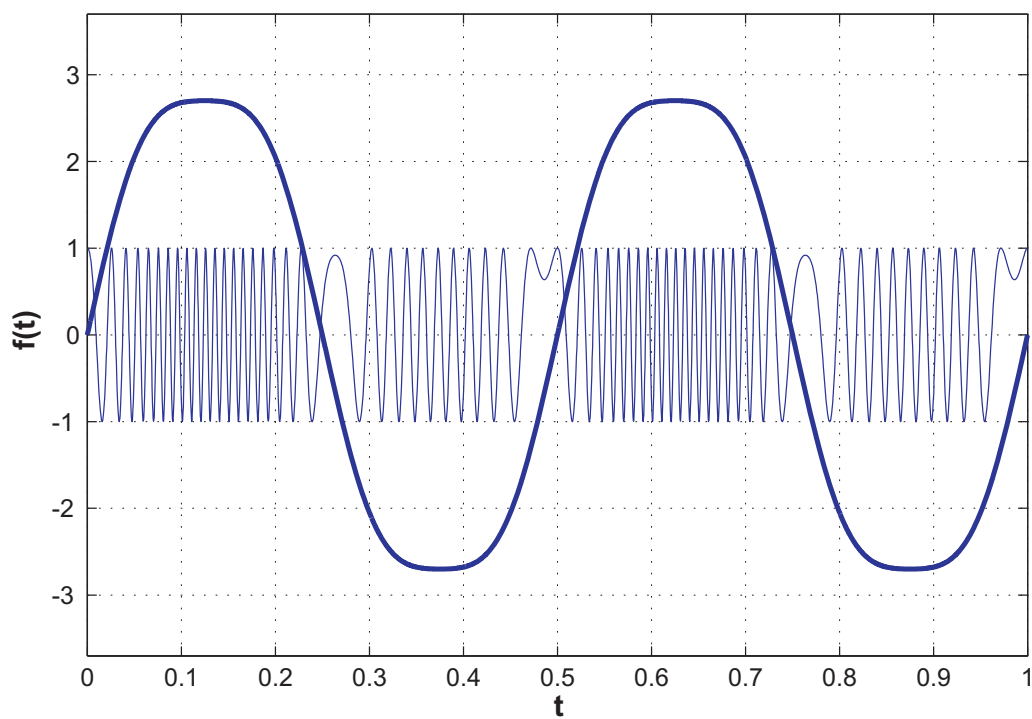


Signál a jeho amplitudová modulace s potlačenou nosnou

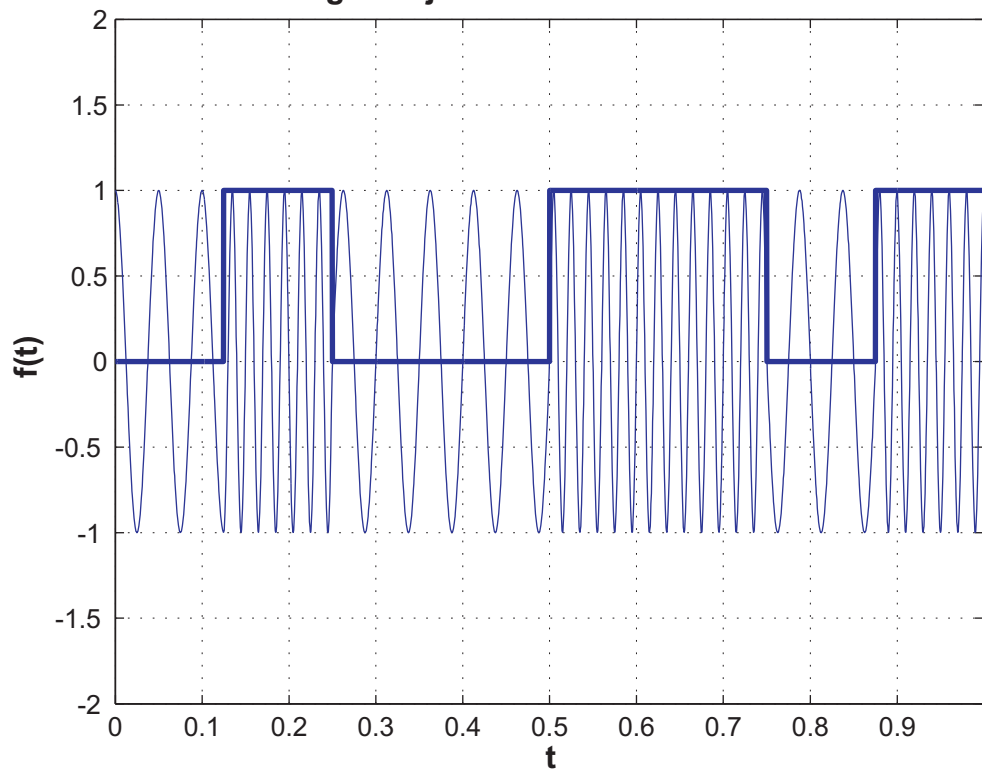


Amplitudová modulace analogového a digitálního signálu

Signál a jeho frekvenční modulace

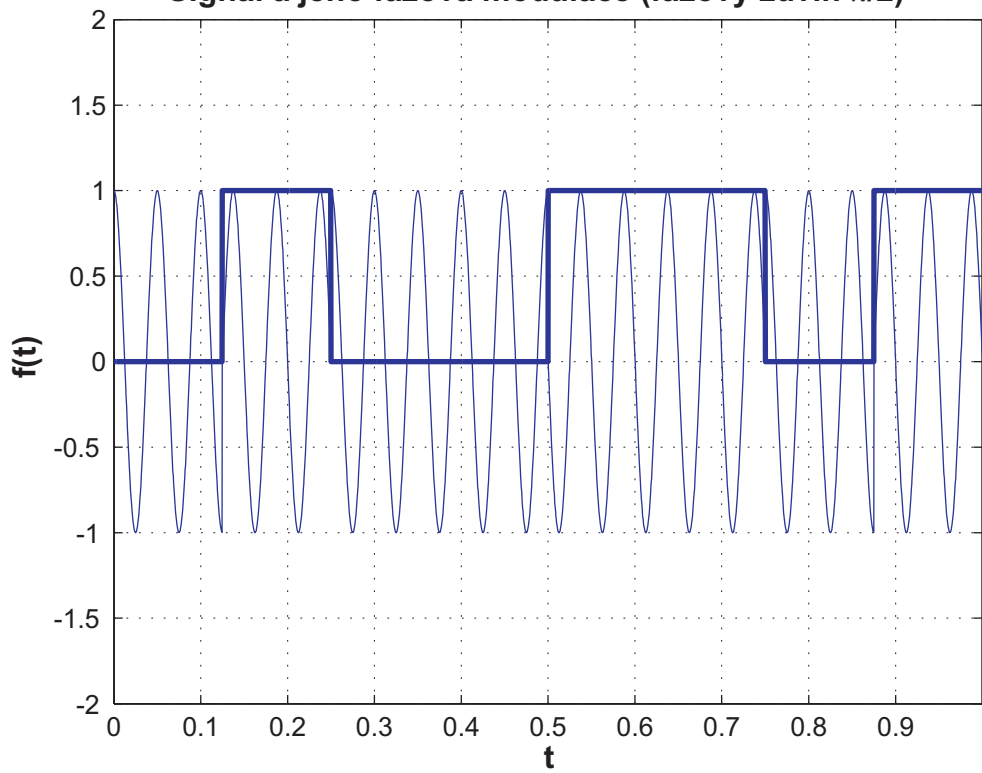


Signál a jeho frekvenční modulace



Frekvenční modulace analogového a digitálního signálu

Signál a jeho fázová modulace (fázový zdvih $\pi/2$)



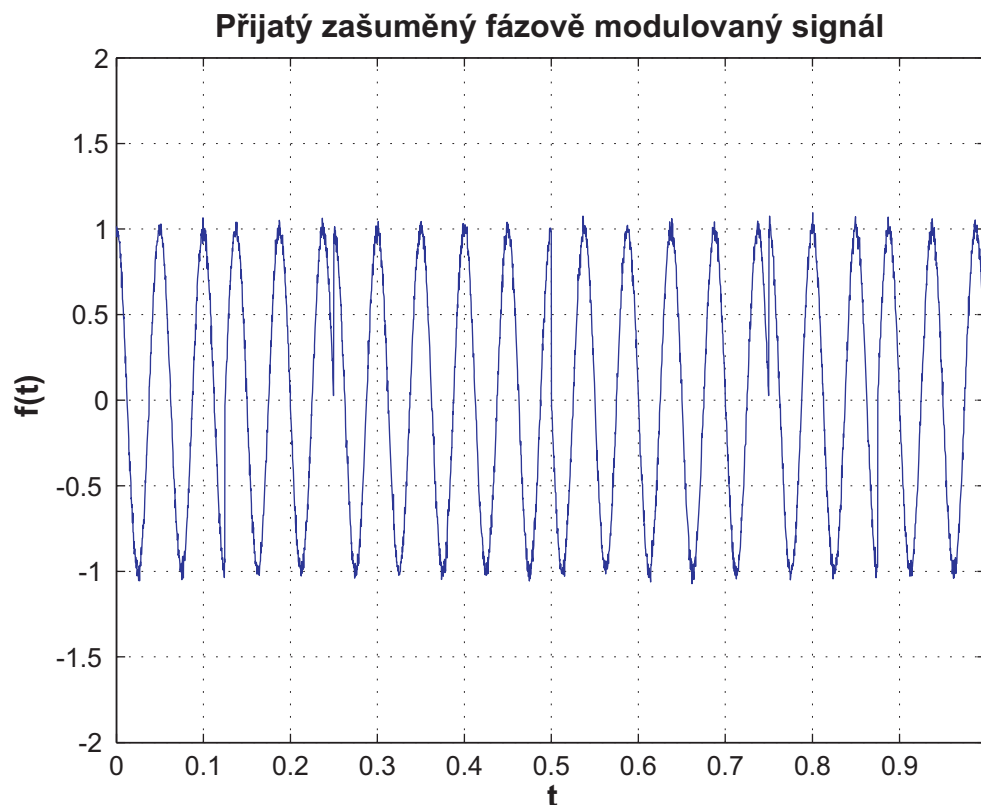
Fázová modulace digitálního signálu

AM funguje na základě změn amplitud přenášeného signálu na základě vysílané informace – amplituda je přímo úměrná amplitudě informačního signálu. Nosná vlna je modulovaná (modifikovaná) vstupním signálem za účelem přenosu informace. Nosná vlna má obvykle mnohem vyšší frekvenci než vstupní informační signál. Amplituda nosné vlny je modulovaná vstupním signálem (tj. přenášenou informací) před samotným přenosem. Vstupní vlna modifikuje amplitudu nosné vlny a tím pádem definuje “obálku” celého průběhu.

FM přenáší informaci přes nosnou vlnu modifikací okamžité frekvence. Při analogové modulaci je změna mezi momentální a standardní frekvencí nosné vlny přímo úměrná okamžité hodnotě amplitudy vstupního signálu.

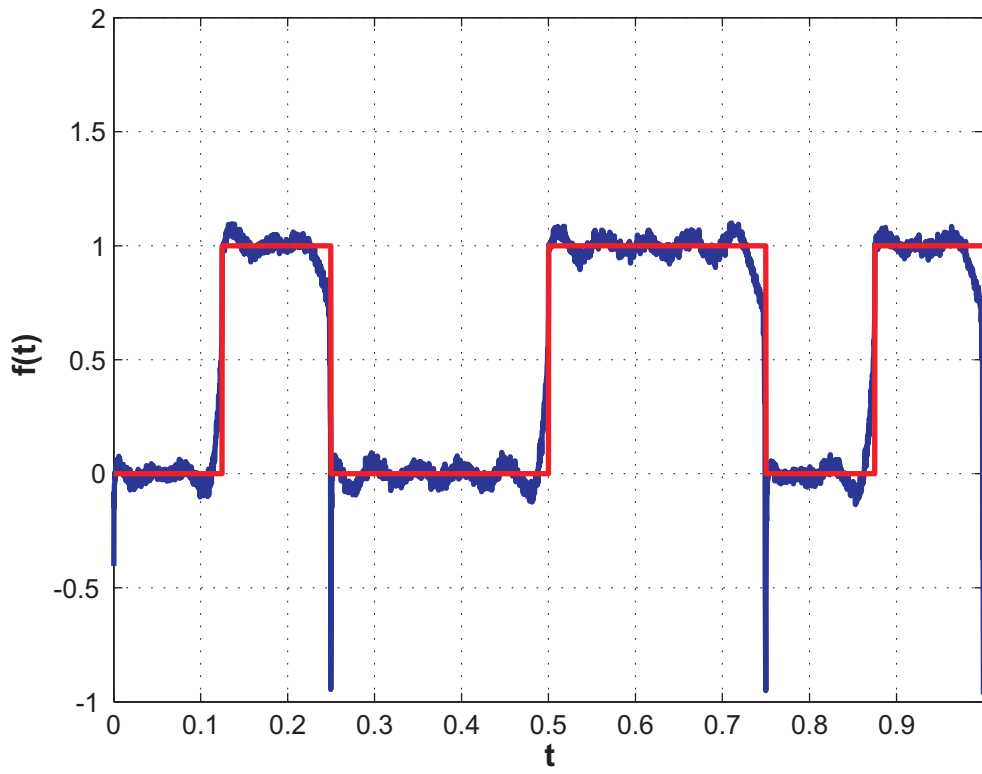
PM je forma modulace, kde je informace reprezentována jako změna okamžité fáze nosné vlny. Zjednodušeně řečeno, modifikace fáze podle datového signálu je fázová modulace. PM není široce užívanou technologií při rádiových přenosech z důvodu potřeby komplexního přijímacího aparátu. Při PM je též potřebné ošetřit nejednoznačnosti, jako například při změnách fáze o $+180^\circ$ nebo -180° . PM se však úspěšně používá v digitálních hudebních syntetizátorech.

Demodulace je proces získávání informace z modulovaného nosného signálu. Demodulace samozřejmě závisí na tom, jaký parametr signálu byl modifikovaný, a tedy jaká modulace byla použita (amplitudová, frekvenční, nebo fázová). Například pro signál s amplitudovou modulací je možné použít synchronní detektor. Avšak pro jiné druhy, například FM a PM modulace je potřebné použít příslušný FM, resp. PM demodulátor.



Přijatý modulovaný signál

Demodulace přijatého zašuměného signálu a porovnání s originálem



Demodulovaný signál porovnaný s původním signálem

2 Časová a frekvenční reprezentace

2.1 Fourierova transformace

Fourierova transformace, pojmenovaná podle Josepha Fouriera, je důležitá v matematice, technických vědách a fyzice.



Jednoduše řečeno, Fourierova transformace představuje matematické vyjádření funkce času v závislosti na frekvenci. Tato funkce je známa jako frekvenční spektrum.

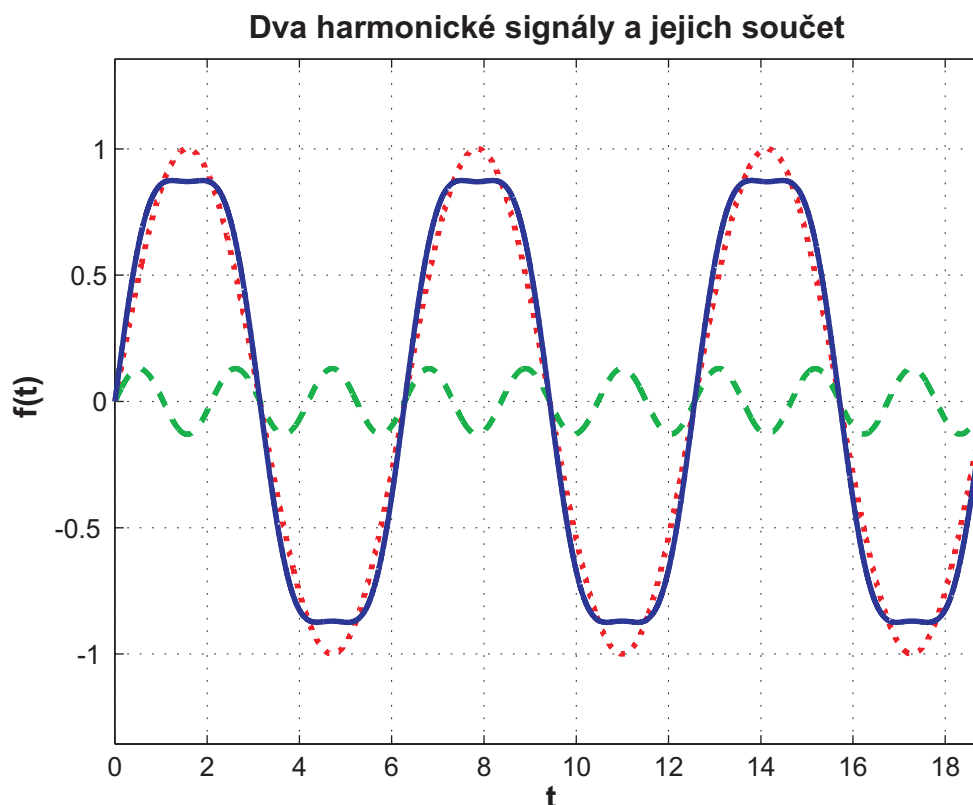


Je důležité poznamenat, že Fourierova transformace se používá jen pro neperiodické analogové signály. Pro periodické analogové signály se používá Fourierova řada.

Řekněme, že máme funkci $f(t)$, která mapuje nějaké časové hodnoty t na nějaké hodnoty $f(t)$.

Nyní zkusíme aproximovat f jako sumu jednoduchých harmonických kmitů, tj. sinusových vln určité kruhové frekvence ω . Samozřejmě, jsou tam některé frekvence, které se dobře hodí na aproximaci f a jiné které jí aproximují méně dobře. Potřebujeme tedy nějakou vhodnou funkci $f(\omega)$, která nám řekne, do jaké míry jsou dané oscilace s frekvencí ω reprezentované v aproximaci f .

Vezměme si například funkci (zobrazená černou čarou) odtud:



Dvě harmonické frekvence tvořící signál

kteřá je definována jako $f(t) = \sin(t) + 0.13\sin(3t)$. Kmitání (tečkovaná čára) s $\omega=1$ má největší vliv na výsledek, takže můžeme říci, že se $F(1)=1$. Druhá vlna ($\omega = 3$, přerušovaná čára) má malý vliv a její amplituda je mnohem menší. Tedy říkáme, že se $F(3)=0.13$. Další frekvence nemusí být vůbec uvedeny v aproximaci, tedy píšeme, že pro tyto se $F(\omega)=0$.

Nyní, když známe $F(\omega)$ nejen pro některé, ale pro všechny možné frekvence ω , můžeme perfektně aproximovat naši funkci $f(t)$. A to je to, co dělá spojitá Fourierova transformace.

Fourierova transformace tedy vezme nějakou funkci času $f(t)$ a vrátí k ní příslušnou funkci $F(\omega)=\text{FT}(f)$, tj. její *Fourierovu transformaci*. FT popisuje, do jaké míry je některá z daných frekvencí reprezentována ve funkci f . Jde pouze o jinou reprezentaci $f(t)$ se stejnou informací, ale se zápisem v jiné doméně (například frekvenční). Častokrát však může být problém vyřešen jednoduše v jiném zastoupení (spočívá v nalezení vhodného souřadnicového systému).

Danou Fourierovu transformaci můžeme integrovat přes všechny frekvence, sestavit váhované sinusové vlny a dostat opět naše f . Tento proces nazýváme **zpětná Fourierova transformace IFT**.

Nejdůležitější je to, že Fourierova transformace má mnoho výhodných matematických vlastností (např. konvoluce je ve frekvenční doméně jenom

násobením). Častokrát je mnohem jednodušší pracovat s Fourierovou transformací jako s funkcí samotnou. Takže funkci transformujeme, provedeme jednoduchou matematickou operaci a následně transformujeme nazpátek.



Řekněme, že chceme redukovat šum v digitálním obraze. Místo manipulace s funkcí obrazu: Pixel→Jas, raději transformujeme celek a pracujeme s F (obraz): Frekvence→Amplituda. Ty části vysokých frekvencí, které zapříčiňují šum, mohou být jednoduše odstraněny – $F(\text{image})(\omega)=0, \omega>...Hz$.



Fourierova transformace (obvykle známá jako dopředná transformace) je definována jako:

$$F(\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t} dt$$

a inverzní Fourierova transformace je definována:

$$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega)e^{j\omega t} d\omega$$

Fourierova funkce $F(\omega)$ je frekvenční reprezentace signálu $f(t)$, též nazývaná spektrální funkce. Spektrální funkce závisí na reálné proměnné ω , takže může být definována jako:

$$F(\omega) = A(\omega) + jB(\omega) = \text{Re}\{F(\omega)\} + j\text{Im}\{F(\omega)\} = |F(\omega)|e^{j\varphi(\omega)}$$

kde $|F(\omega)|$ je absolutní hodnota a $\varphi(\omega)$ je fázové spektrum.

V každé rovnici je j definováno jako $j = \sqrt{-1}$. Komplexní mocnina je srdcem transformace. Komplexní mocnina je v podstatě komplexní číslo, kde obě složky, reálná a imaginární, jsou sinusoidy. Exaktní vztah se nazývá Eulerova rovnice $e^{j\varphi} = \cos\varphi + j\sin\varphi$, která vede k zajímavé (a krásné) rovnosti $e^{j\pi} + 1 = 0$. Je jednodušší pracovat s komplexními mocninami než s trigonometrickými funkcemi, poskytují též mnohem jednodušší způsob zápisu sinusoid (je jednodušší napsat $e^{j\varphi}$ než napsat $\cos\varphi + j\sin\varphi$).



Komplexní mocniny (nebo siny a cosiny) jsou periodické funkce a soubor komplexních mocnin je kompletní a ortogonální. Proto Fourierova transformace může reprezentovat jakoukoliv částečnou spojitou funkci a chyba, měřená metodou nejmenších čtverců (least-square error) mezi funkcí a její Fourierovou reprezentací, je minimální.

Existuje též jiný úplný a ortogonální soubor periodických funkcí; například Walshova funkce (čtvercové vlny), která je užitečná pro digitální elektronická zařízení (více o tom v kapitole Ortogonální transformace).



Proč se vždy střetáváme s komplexními mocninami při řešení fyzikálních problémů? Proč jsou monochromatické vlny sinusového charakteru a ne periodické sledy čtvercových nebo trojúhelníkových vln? Důvod spočívá v tom, že derivace komplexních mocnin jsou jednoduše přepočítané komplexní mocniny. Jinými slovy, komplexní mocniny jsou funkce diferenciálních operátorů (operátor rozdílů). Většina fyzikálních systémů se řídí podle pravidel diferenciálních rovnic. Tedy analogové elektronické filtry budou konvertovat sinusovou vlnu na jinou sinusovou vlnu se stejnou frekvencí (ale nikoliv nevyhnutelně se stejnou amplitudou a fází), čímž filtrovaná čtvercová vlna už více nebude čtvercovou vlnou. Tato vlastnost komplexních mocnin dělá Fourierovu transformaci užitečnou v rozmezí od rádiového šíření až po kvantovou mechaniku.



Fourierova transformace je definována též pro dvojrozměrné signály a matematicky je definována jako:

$$F(\omega_1, \omega_2) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x_1, x_2) e^{-j\omega_1 x_1} e^{-j\omega_2 x_2} dx_1 dx_2$$

a inverzní transformace je definována:

$$f(x_1, x_2) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\omega_1, \omega_2) e^{j\omega_1 x_1} e^{j\omega_2 x_2} d\omega_1 d\omega_2$$

2.2 Diskrétní Fourierova transformace

Diskrétní Fourierova transformace DFT se používá pro získávání spektra diskretních signálů. Abychom popsali signál ve frekvenční doméně, potřebujeme konečný počet sinusoid.

Pro znázornění co dělá DFT použijeme následující příklad. MP3 přehrávač posílá reproduktoru audio informace v podobě kolísavého napětí elektrického signálu. Výsledkem je pohyb vzduchových částic a produkce zvuku. Kolísání audio signálu v čase může být graficky znázorněno: osa x představuje čas a osa y napětí elektrického signálu. Toto vypadá jako nepravidelný vlnový hlavolam, který je ve skutečnosti sumou několika pravidelných hlavolamů, které představují různé frekvence zvuku. Frekvence znamená jen rychlost, s jakou se molekuly vzduchu pohybují tam a zpět, nebo kolísání napětí.



DFT vytváří matematicky to, co fyzikálně dělá lidské ucho: rozkládá signál na jednotlivé složky frekvence. Na rozdíl od analogového signálu z gramofonu, digitální signál z MP3 přehrávače je pouze sled čísel, které reprezentují velmi krátké vzorky skutečného zvuku: například CD-kvalita digitálně zvukového přehrávače sbírá 44 100 vzorků za sekundu. Jestliže vybereme několik po sobě jdoucích hodnot z digitálního signálu: 8, 128 anebo 1 000 – DFT je reprezentuje jako vážený součet stejného počtu frekvencí. („Vážený“ jen znamená, že některé frekvence se významněji podílí na celku než jiné.)



Diskrétní Fourierova Transformace (DFT) je matematicky vyjádřena jako:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-n2\pi jk/N}$$

a inverzní diskretní Fourierova transformace (IDFT) je definována jako:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{n2\pi jk/N}$$

kde N je počet vzorků diskretního signálu a $n = 0, 1, 2, \dots, N-1$, $e^{2\pi jk/N}$ je velmi často nahrazené jako Ω ($\Omega = e^{2\pi jk/N}$).

DFT sledu N -bodových časových vstupů je N -bodové frekvenční spektrum s Fourierovou frekvencí k s rozsahem od $-(N/2 - 1)$, přes nultou frekvenci, nazývanou jednosměrná složka, až po nejvyšší Fourierovy frekvence $N/2$. Každá spektrální složka představuje celé číslo sinusových period prezentovaných v časovém sledu. Amplitudy a fáze jsou reprezentovány amplitudami A_k a fázemi \varnothing_k těchto sinusoid. Pro shrnutí, každá spektrální složka může být popsána jako $X(k) = A_k e^{j\varnothing_k}$.

Diskrétní Fourierova transformace je definována též pro dvojrozměrné signály a může být reprezentována jako série rozšíření funkce obrazu (přes 2D prostor).



Definice dopředné a inverzní **2D FT** jsou:

$$X(\Omega_1, \Omega_2) = \sum_{n_1=-\infty}^{\infty} \sum_{n_2=-\infty}^{\infty} x(n_1, n_2) e^{-j\Omega_1 n_1} e^{-j\Omega_2 n_2}$$

$$x(n_1, n_2) = 1/(2\pi)^2 \int_{\Omega_1=-\pi}^{\pi} \int_{\Omega_2=-\pi}^{\pi} X(\Omega_1, \Omega_2) e^{j\Omega_1 n_1} e^{j\Omega_2 n_2} d\Omega_1 d\Omega_2$$

Spektrum $X(\Omega_1, \Omega_2)$ je komplexní když i posloupnost $x(n_1, n_2)$ je reálná. Spektrum může být definováno též jako suma reálných a imaginárních částí nebo jako součin magnitudy a fáze.

$$X(\Omega_1, \Omega_2) = |X(\Omega_1, \Omega_2)| e^{j\theta_x(\Omega_1, \Omega_2)} = X_R(\Omega_1, \Omega_2) + jX_I(\Omega_1, \Omega_2)$$

2.3 Spektrum



Frekvenční spektrum časového signálu je reprezentace tohoto signálu ve frekvenční doméně. Frekvenční spektrum může být generované přes Fourierovu transformaci (nebo diskrétní Fourierovu transformaci) signálu a výsledné hodnoty jsou obvykle prezentovány jako amplituda a fáze, obě vykreslené poměrně k frekvenci.

Signál může být reprezentován jako amplituda, která se mění s časem, má odpovídající frekvenční spektrum. Toto zahrnuje známé pojmy jako viditelné světlo (barva), hudební noty, rádio/TV programy a též pravidelná rotace Země. Když jsou tyto fyzikální jevy reprezentovány ve formě frekvenčního spektra, určitý fyzikální popis jejich vnitřních procesů se stává mnohem jednodušší. Častokrát frekvenční spektrum jasně ukazuje harmonické složky, znázorněné jako viditelné špičky nebo čáry, které poskytují jasný náhled do mechanismu, který vytváří celý signál.

Spektrální analýza je technický proces dekompozice celého signálu na jednodušší části. Protože je popis vyšší, mnoho fyzikálních procesů je lépe popsáných jako suma mnohých individuálních frekvenčních komponentů. Jakýkoliv proces, který kvantifikuje různá množství (tj. amplitudy, výkon, intenzita nebo fáze) v poměru k frekvenci, se může nazývat spektrální analýza.

Spektrální analýza může být prováděna na celém signálu (obvyčejně periodickém signálu). Případně může být signál (většinou neperiodický nebo kvaziperiodický) rozdělený na kratší části, nazývané rámce, a spektrální analýza může být aplikována na tyto individuální části.

Fourierova transformace funkce vytváří frekvenční spektrum, které obsahuje všechny informace o originálním signálu, ale v jiné formě. To znamená, že originální funkce může být úplně zrekonstruovaná (syntetizovaná) inverzní Fourierovou transformací.

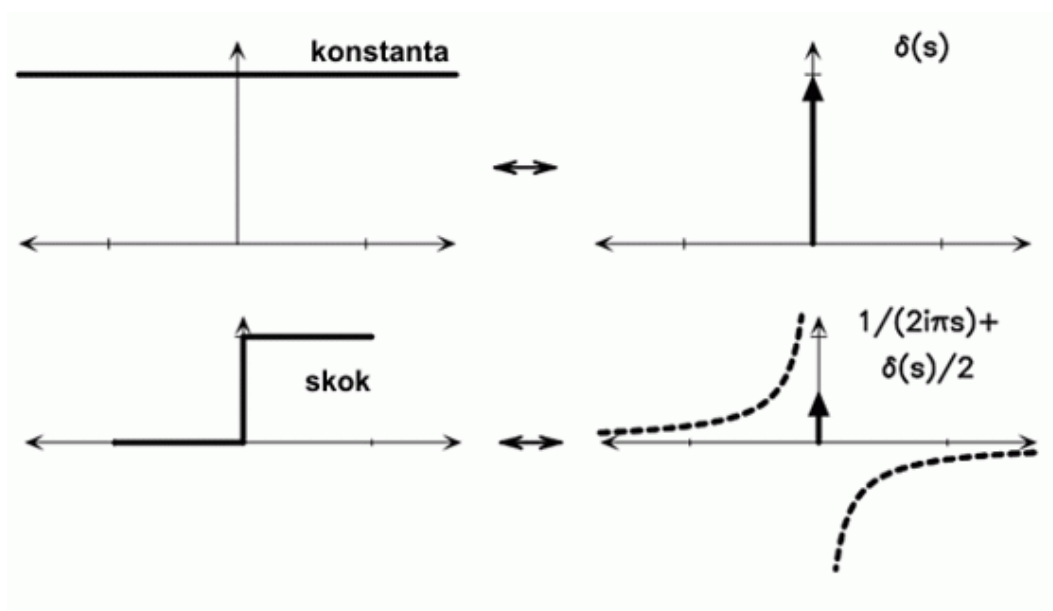


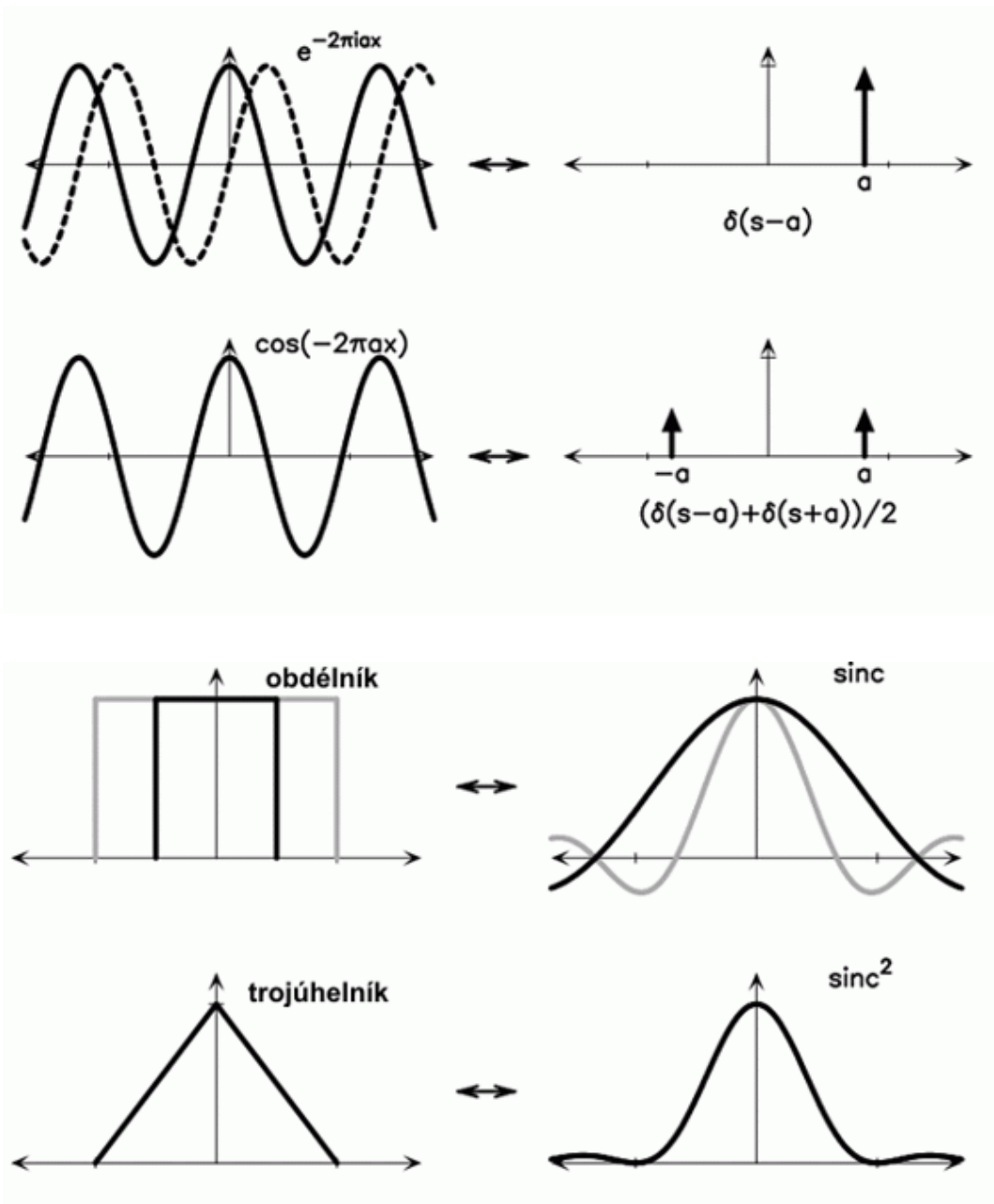
Pro dokonalou rekonstrukci musí spektrální analýza zachovat amplitudu i fázi každé frekvenční komponenty. Tyto dvě informační části mohou být reprezentovány jako dvojrozměrný vektor, jako komplexní číslo nebo jako magnituda (amplituda) a fáze v polární souřadnicové soustavě. Známa technika v oblasti zpracování signálu je brát v úvahu mocninu amplitudy nebo výkonu; v tomto případě výsledný graf se odvolává na spektrum výkonu.

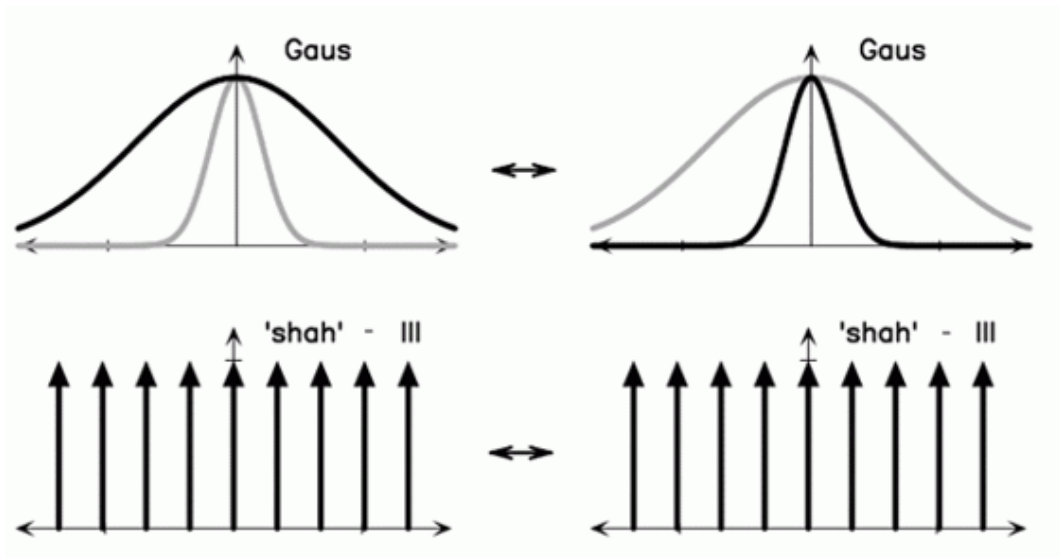
Následující tabulka shrnuje typy signálu a jejich spektrum.

| Signál | Spektrum |
|------------------------|------------------------|
| spojitý periodický | diskrétní neperiodické |
| spojitý neperiodický | spojité neperiodické |
| diskrétní periodický | diskrétně periodické |
| diskrétní neperiodický | spojité periodické |

Následující obrázek ukazuje základní páry Fourierovy transformace (na obrázku jsou zobrazeny jen amplitudy). Tyto mohou být kombinovány použitím níže uvedené teorie o Fourierově transformaci na generování Fourierových transformací různých funkcí.

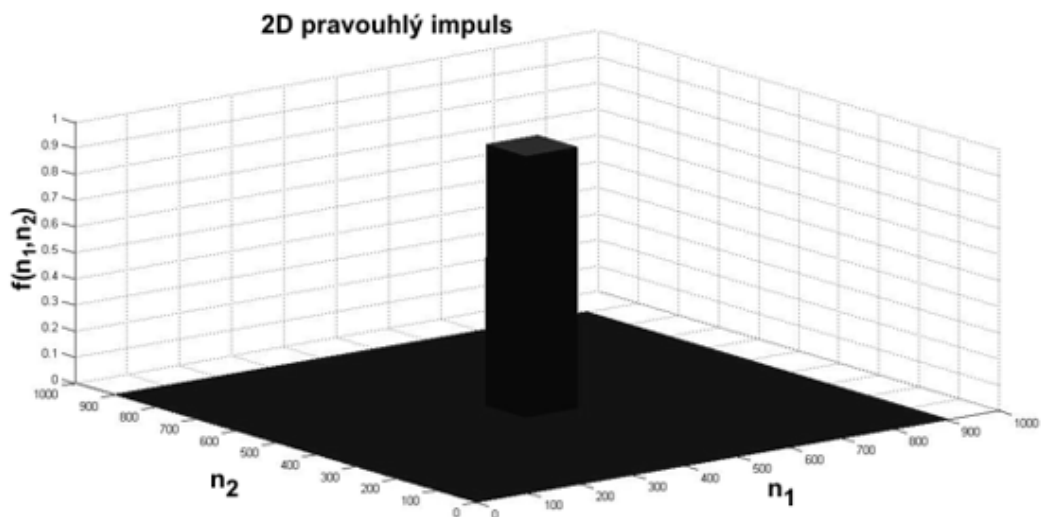




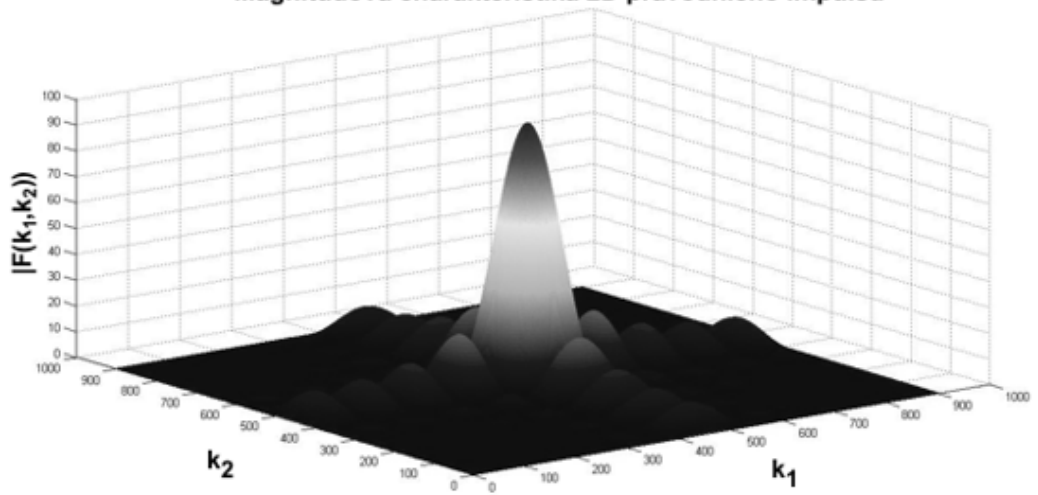


Základní páry Fourierových transformací

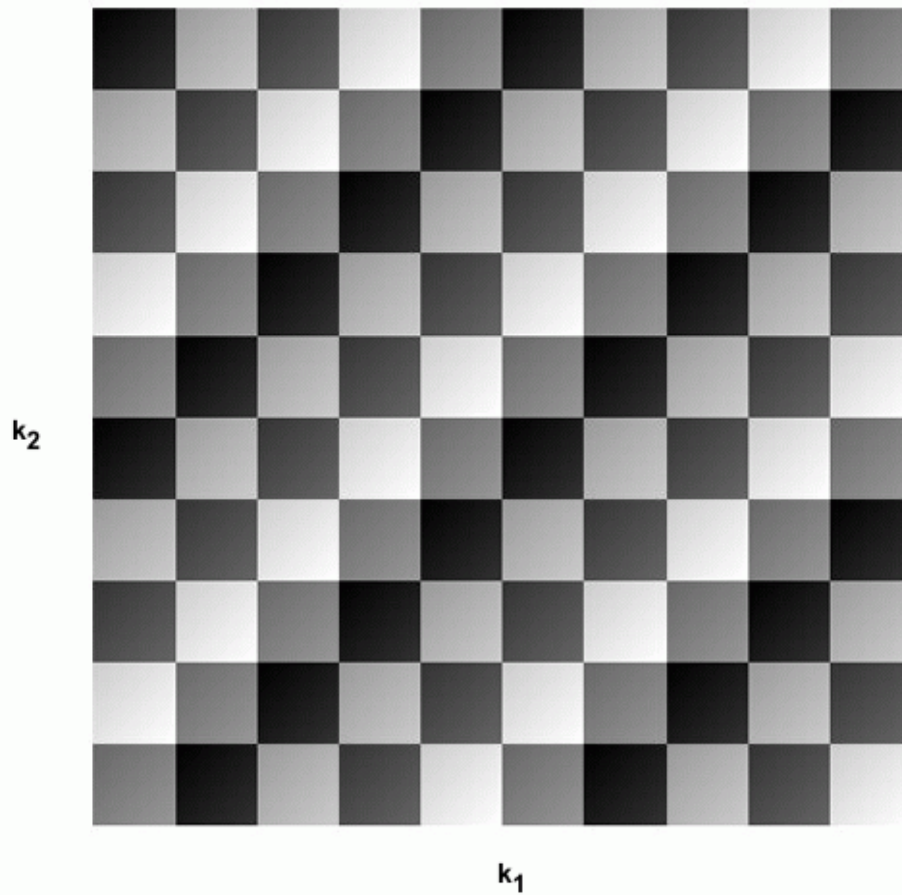
Obrázek níže zobrazuje dvojrozměrné signály a jejich spektra (magnitudovou a fázovou charakteristiku).



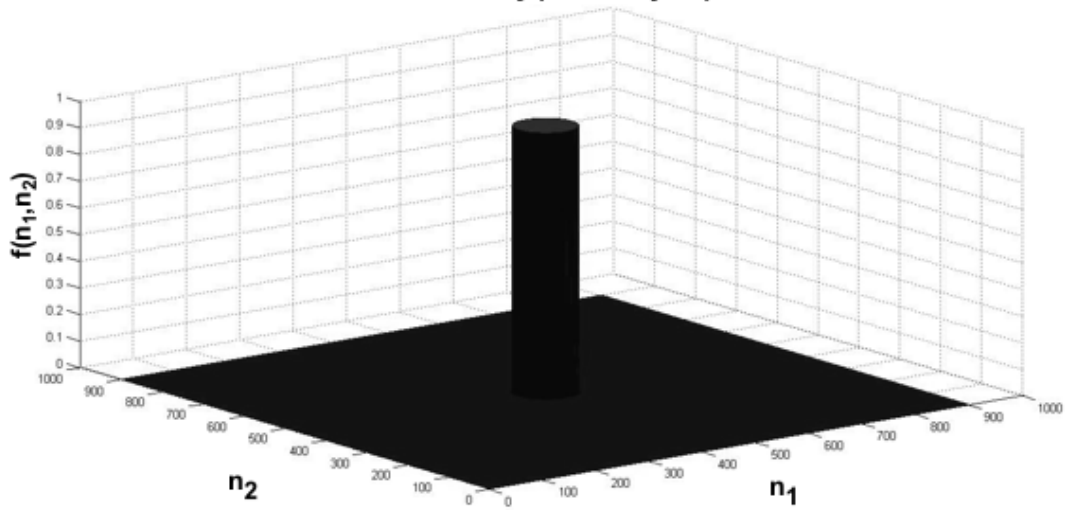
Magnitudová charakteristika 2D pravouhlého impulsu



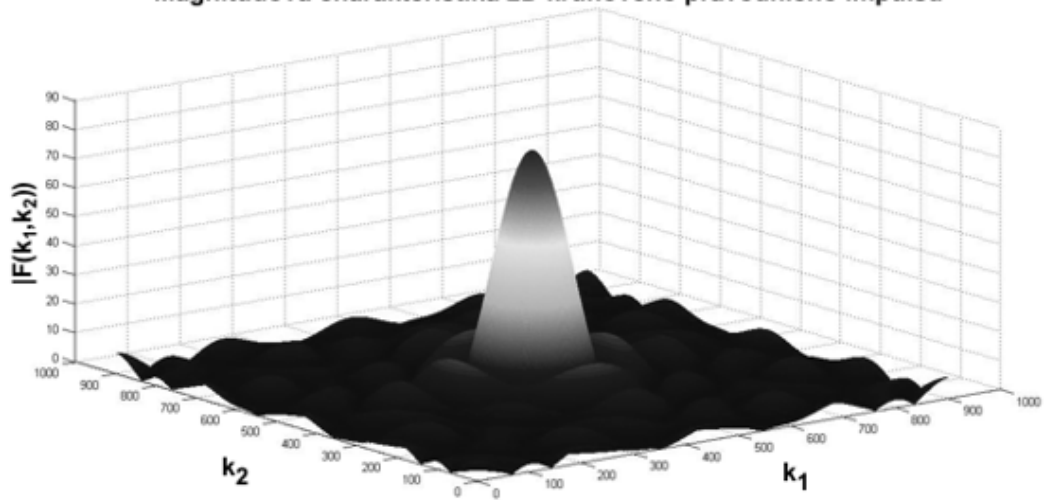
Fázová charakteristika 2D pravouhlého impulsu



2D kruhový pravouhlý impuls

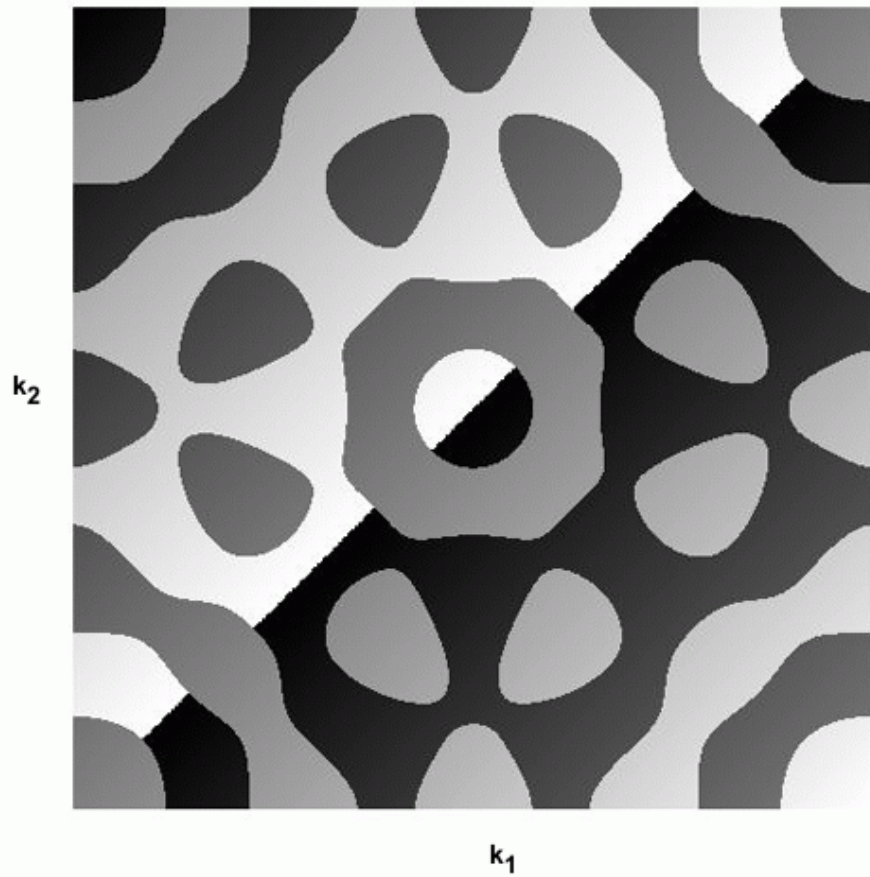


Magnitudová charakteristika 2D kruhového pravouhlého impulsu



2D čtvercová funkce, magnitudová a fázová frekvenční charakteristika

Fázová charakteristika 2D kruhového pravouhlého impulsu



2D kruhová funkce, magnitudová a fázová frekvenční charakteristika

2.4 Ortogonální transformace



Ortogonalní transformace umožňují reprezentovat časový průběh signálu ve formě jeho zevšeobecněného spektra. Transformace se používá, protože ve spektrální oblasti je potom možno vykonávat potřebné matematické operace, které umožňují např. rychlejší vyhodnocení signálů, dokáží vyloučit redundanci, atd.

Diskrétní ortogonální transformace nacházejí široké využití v mnohých oblastech při kompresi údajů, rozpoznávání obrazů, analýze a syntéze hovorových signálů, atd. Naším hlavním cílem v tomto kurzu jsou jednorozměrné ortogonální funkce a transformace.

Nejjednodušším způsobem matematického vyjádření jednorozměrných signálů je lineární kombinace (součet násobků) některých základních (bázových) funkcí. Takové vyjádření jednorozměrných signálů je velmi výhodné zejména v lineárních soustavách, protože umožňuje řešit mnohé úlohy na základě principu superpozice. Vyžaduje se, aby základní funkce $u(k, t)$ se lehce vyčíslovaly a měly jednoduchý tvar a těmto požadavkům nejvíce vyhovují ortogonální funkce.



Matematická definice ortogonálních funkcí $u(0, t), u(1, t), \dots, u(N-1, t)$ v intervalu $\langle t_1, t_2 \rangle$ je následující:

$$\int_{t_1}^{t_2} u(k, t)u(m, t) dt = 0, \quad k \neq m$$

$$\int_{t_1}^{t_2} u^2(k, t) dt = U_k, \quad k \neq m$$

V případě, že $U_k = 1$, funkce nazýváme ortonormální (tj. mají velikost jedna).

Příklad matematického vyjádření jednorozměrného signálu $x(t)$, pomocí lineární kombinace některých základních (bázových) funkcí:

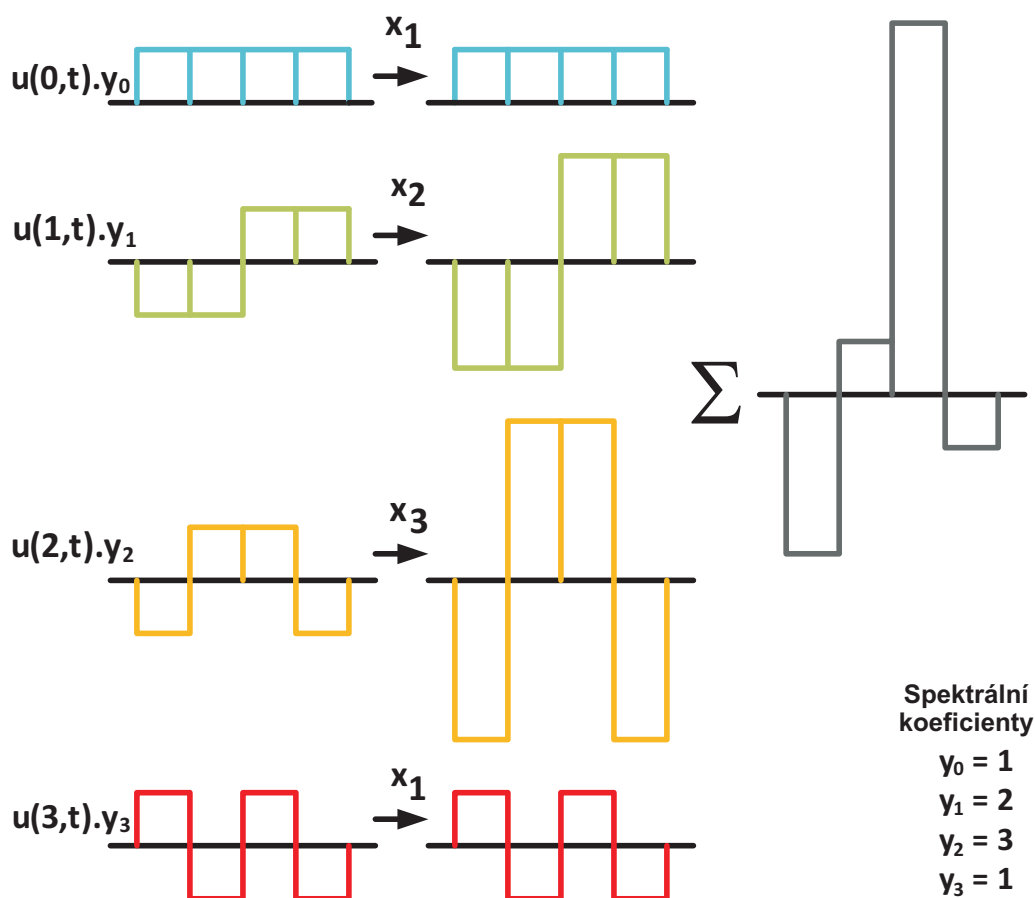
$$x(t) \cong \sum_{k=0}^{N-1} y_k u(k, t)$$

kde y_k jsou spektrální koeficienty definované jako:

$$y_k = \frac{1}{U_k} \int_{t_1}^{t_2} x(t)u(k, t) dt$$

Prostudujte si následující příklad.

System prvních 4 Walshových funkcí



Příklad matematického vyjádření jednorozměrných signálů, pomocí lineární kombinace některých báзовých funkcí

Mezi nejvíce používané ortogonální funkce v oblasti zpracování signálů patří funkce Walsh, Haar a Rademacher.



Rozklad (aproximace) diskretního signálu $x(nT)$, kde T je diskretizační perioda s počtem vzorků M podle některé diskretní ortogonální báze, je dán vztahem:

$$x(nT) = \sum_{k=0}^{M-1} y_k u(k, nT) \quad n = 0, 1, \dots, M-1$$

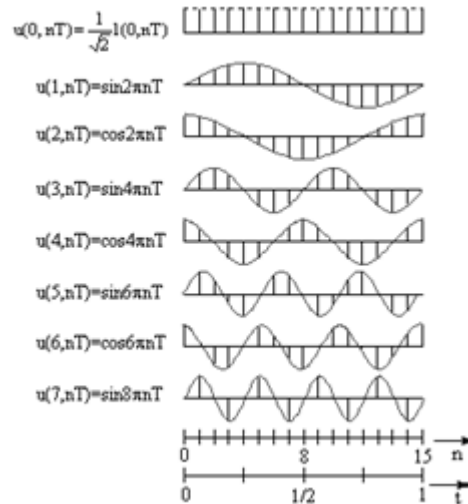
přičemž optimální koeficienty:

$$y_k = \sum_{n=0}^{M-1} x(nT) u(k, nT)$$

U harmonických signálů je parametrem funkce frekvence. V případě neharmonických signálů se používá termín sekvence, která je určena počtem

průsečíků nulové úrovně za sekundu. V souvislosti s diskretními funkcemi se sekvence určí z počtu změn znaménka vzorků za sekundu.

Diskretní ortogonální báze funkce můžeme získat diskretizací funkcí spojité ortogonální báze harmonických funkcí.



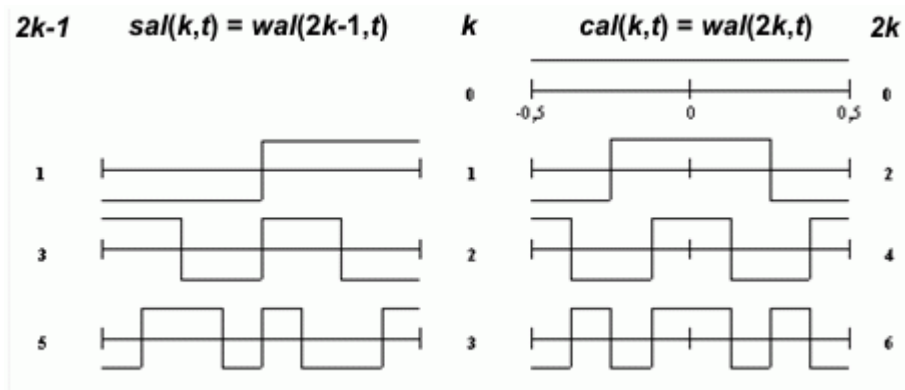
System Walshových funkcí



Walshovy funkce tvoří uspořádanou množinu pravoúhlých impulsů, které mají jen dvě možné hodnoty amplitudy (+1 anebo -1) a představují úplný systém ortogonálních funkcí. Jsou závislé na dvou argumentech, a to času (t) a na pořadovém čísle (k). Značíme je $wal(k, t)$ a můžeme rozlišovat sudé (cosinové-Walshové) $cal(k, t)$ a liché (sinusové-Walshové) $sal(k, t)$ funkce, které jsou dány:

$$cal(k, t) = wal(2k, t), \quad k = 0, 1, 2, \dots$$

$$sal(k, t) = wal(2k - 1, t), \quad k = 0, 1, 2, \dots$$



Příklad sudých a lichých bázeových Walshových funkcí

Podle uspořádání Walshových funkcí tyto dělíme do třech skupin:

1. uspořádání Walshovo (sekvenční) $walw(k, t)$
2. uspořádání Paleyho (diadické) $walp(k, t)$
3. uspořádání Hadamardovo (přirozené) $walh(k, t)$

Jednotlivá uspořádání mají svá realizační opodstatnění. Báze obsahují ty samé funkce, které jsou však odlišně seřazené. Jako příklad je možné uvést Hadamardovo (přirozené) uspořádání.

+

Výhodou tohoto uspořádání je jednoduchá tvorba bázových funkcí větších rozměrů.

Na obrázku níže je prvých osm spojitých a diskrétních Walshových funkcí s přirozeným uspořádáním, přičemž maticový zápis diskrétních Walshových funkcí dává v tomto případě Hadamardovu matici $U_h(3)$ s rozměrem 8×8 .



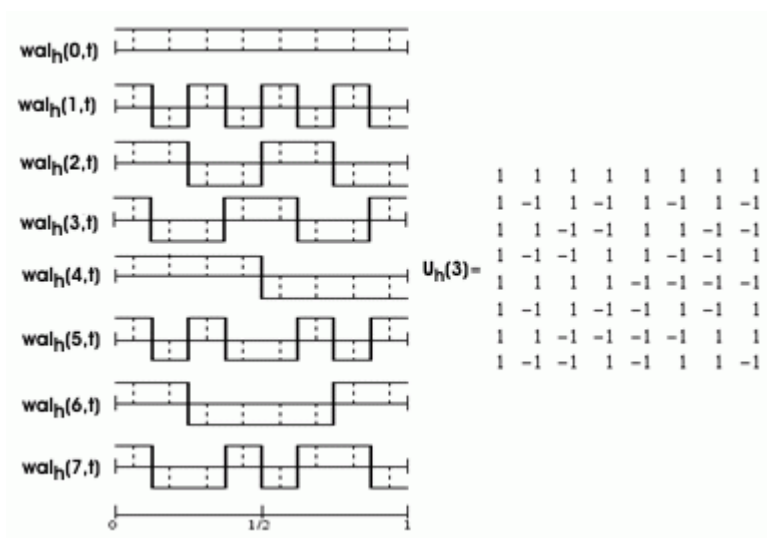
Obecně můžeme Hadamardovu matici $U_h(r)$ s rozměrem $M \times M$, kde $M=2^r$, vypočítat pomocí kroneckerovského (přímého) součinu matic z $U_h(r-1)$.

$$U_h(r) = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes U_h(r-1) = \begin{bmatrix} U_h(r-1) & U_h(r-1) \\ U_h(r-1) & -U_h(r-1) \end{bmatrix}$$

kde $U_h(0)=1$



Kroneckerovský součin, označený jako \otimes , je operace se dvěma maticemi libovolných rozměrů, jejímž výsledkem je matice s maticovými prvky.



Přirozené uspořádání spojitých, diskrétních a matice Walshových funkcí

Existují také další transformace obsahující harmonické bázové funkce. Kromě diskrétní Fourierovy transformace sem patří *diskrétní kosinová transformace* (*discrete cosines transformation* **DCT**), *diskrétní sinusová transformace* (*discrete sinus transformation* **DST**), *diskrétní Hartleyho transformace* (*discrete Hartley transformation* **DHYT**).

3 Analogové a digitální technologie

3.1 Multiplexování

Multiplex spojuje několik analogových nebo digitálních signálů do jednoho telekomunikačního spoje. Multiplexor (mux) se používá zejména na zvýšení objemu dat, které se mohou přenést v daném pásmu s určitou šířkou a v určitém čase.

Naopak, demultiplexor (demux) je zařízení s jedním vstupem, které přijme vstupní signál a rozdělí ho do mnoha výstupních signálů. Multiplexor se obvykle používá s doplňkovým demultiplexorem na straně přijímače.

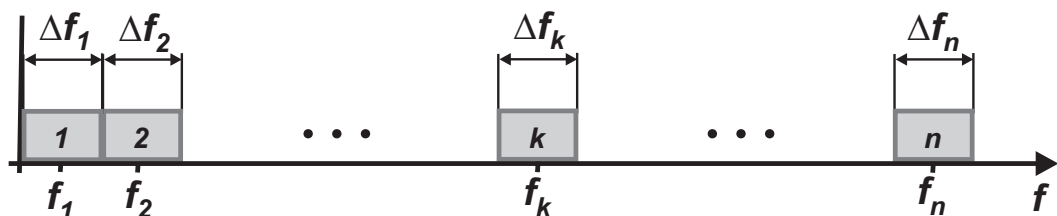
Frekvenční multiplex

Frekvenční multiplex (Frequency division multiplexing FDM) je analogová technologie. Frekvenční multiplex spojuje několik signálů do jednoho společného zdroje tak, že se vysílají v odlišných frekvenčních pásmech.

- + Důvodem využívání systémů s frekvenčním dělením (frekvenčním multiplexem) je možnost současného vysílání více signálů z různých kanálů, a také to, že mohou být vysílány signály z různých prostorově odlehlých míst.
- Nedostatkem je vzájemné ovlivňování kanálů, které vzniká překrýváním spekter signálů, neideálním filtrováním a objevením se nežádoucích frekvenčních složek vlivem přeslechů v elektrických obvodech.



Při frekvenčním multiplexování jsou jednotlivým signálům na frekvenční ose přiděleny navzájem se nepřekrývající frekvenční pásma $\Delta f_1, \Delta f_2, \dots, \Delta f_n$. Obvykle $\Delta f_1 = \Delta f_2 \dots = \Delta f_k = \Delta f_n$. Spektra signálů příslušných kanálů se musí vejít do vyhrazených frekvenčních pásem Δf_n .



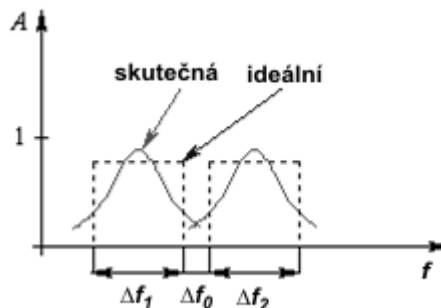
Rozdělení frekvenčního spektra příslušným kanálem

Frekvenční pásmo přenosové cesty F určují hraniční frekvence kanálů, čili minimální frekvence intervalu Δf_1 ($\Delta f_{1_{min}}$) a maximální frekvence intervalu Δf_n ($\Delta f_{n_{max}}$).

$$F = \Delta f_{n\max} - \Delta f_{l\min}$$

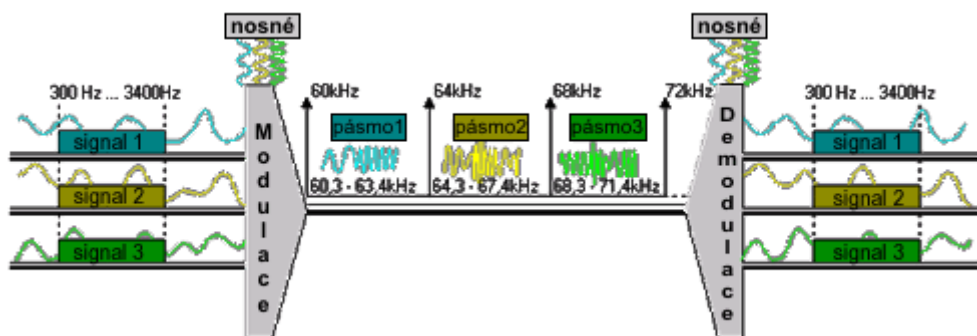
Důležitou součástí frekvenčního multiplexu na straně vysílače i přijímače jsou pásmové filtry (PP), které na vysílací straně ohraničují frekvenční pásma příslušných kanálů. Na přijímací straně zase filtry rozdělují signály, které po průchodu demodulátorem mohou být přijaty přijímacím převodníkem. Při výběru frekvenčního pásma filtrů se především bere v úvahu požadovaná přenosová rychlost systému.

V reálných filtrech se útlum za hranicemi frekvenčního pásma nemění skokem, ale spojitě, proto se mezi frekvenčními pásmy jednotlivých filtrů vynechává ochranné frekvenční pásmo Δf_0 . Poměr $\Delta f_0 / \Delta f = k_f$ závisí na amplitudově-frekvenčních charakteristikách použitých pásmových filtrů (viz obrázek dole). Více informací o filtrech je v kapitole Filtry.



Amplitudově-frekvenční charakteristiky (ideální a skutečná) pásmových filtrů

Prostřednictvím modulace různými signály (nosný signál, nosná, angl. carrier) se posouvají jednotlivá (např. hovorová) pásma do subpásem, a tak se přenášejí. Po demodulaci na přijímací straně jsou k dispozici jednotlivá hovorová pásma ve své původní formě. Tímto postupem se dá přenášet, prostřednictvím jednoho vedení více hovorových pásem současně.



Princip frekvenčního multiplexu

Frekvenční multiplex se používal zejména v analogových telekomunikačních systémech. V současnosti jej využívají hlavně TV společnosti (každý kanál je na

mírně odlišné frekvenci – přepínáním kanálu na TV přeladíme na jiné frekvenční pásmo) a v optických komunikacích.

Časový multiplex

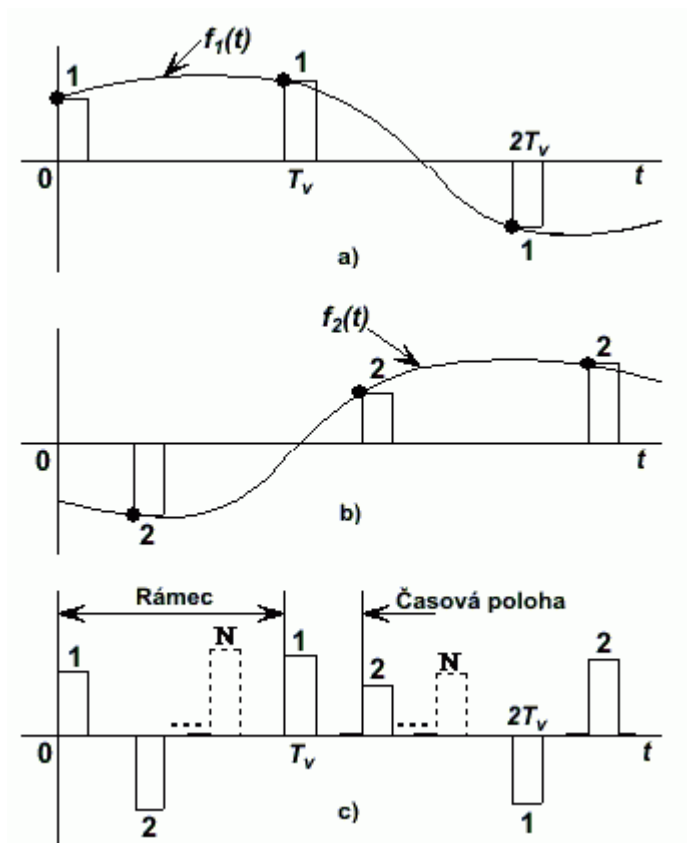


Časový multiplex (Time division multiplex TDM) se využívá v síťových i telefonních systémech a dělá přesně to, co říká samotný název: periodicky vybere vzorky z několika signálů s nižší rychlostí, spojené je přeneseno přes jeden přenosový kanál s vyšší rychlostí a na konci obnoví původní signál. Vzorky vybírá periodicky na základě času.

Vstupní zařízení, multiplexor, vybírá vzorky z různých signálů, spojí je do jednoho a vysílá je jedním kanálem. Tímto způsobem je více vzorků z různých zdrojů spojeno a přeloženo do vysokorychlostního kanálu. Toto je možné realizovat díky tomu, že zdroje vysílají data relativně malou modulační rychlostí (jako 300 Bd) a výsledný kanál sdružuje modulační rychlosti ze všech zdrojů (1200 Bd). Na konci přenosové cesty jiný multiplexor rozdělí přenášená data a pošle je do cíle, znova tou samou menší modulační rychlostí jakou vstoupili do TDM systému.

Tuto technologii využívají zejména telefonní společnosti, které musí přepojit obrovské množství telefonních hovorů přes omezený počet kabelů. V případě, že hovor je rozdělený na vzorky a je obnovený rychleji než to dokáže lidské ucho zjistit, nikdo to nezjistí. Právě proto telefonní přenosy využívají TDM s vysokorychlostními nosiči, a takto přenesou několik hovorů najednou.

— Nevýhodou multiplexů využívajících časový multiplex je skutečnost, že přidělují časové polohy i tehdy, když zdroj nic nevysílá. To vede k neefektivitě.



Časový multiplex a) informace 1, b) informace 2, c) multiplexované posloupnosti impulsů

Synchronizace

Synchronizace vzorkovacího procesu je nezbytná pro to, aby se udržely správné pozice posloupností impulsů. Zabezpečuje se posloupností hodinových impulsů, která je referenční pro všechny vzorkovací obvody. V přijímači musí existovat podobná posloupnost hodinových impulsů, která zabezpečí správné přebírání jednotlivých impulsů příslušných informací. Jinak by přijímač/demultiplexor nemohl určit, do kterého cílového kanálu patří jednotlivé zdrojové signály a jak signál zrekonstruovat zpětně. Posloupnost hodinových impulsů ve vysílači a přijímači musí mít definovány fázové poměry a vzájemně se synchronizují. Časová synchronizace je proto nezbytně nutná pro správné fungování TDM technologie.



Synchronizace hodinových impulsů sama o sobě ještě negarantuje správné přebírání informací. Je k tomu nezbytná ještě i rámcová synchronizace. Pro synchronizaci hodinových impulsů i rámcovou synchronizaci se obvykle vymezuje jedna časová poloha, ve které se umístí speciální impulsy (např. Větší než je předpokládaná největší hodnota v následující časové poloze). Začátek každého rámce je tak jednoznačně detekovatelný jednoduchými prahovacími obvody.

3.2 Lineární diskrétní časově – invariantní systém

Lineární, konečný, časově – invariantní systém (LKI) se přímo využívá v seismologii, zpracování signálů, obvodech, teorii kontroly a jiných technických oblastech. Analýza spojitého LKI a diskrétního LKI je velmi podobná. Diskrétní oblast je méně technicky náročná, proto se zaměříme na tyto systémy.



LDKI (*Lineární, diskrétní, konečný, časově – invariantní systém*) má na vstupu a na výstupu jeden signál, pro který platí tyto vlastnosti:

- Systém je lineární. To znamená, že jak vstup $x_1(n)$ a $x_2(n)$ generuje výstup $y_1(n)$ a $y_2(n)$, a jak a_1 a a_2 jsou konstanty, potom vstupní signál $a_1x_1(n) + a_2x_2(n)$ generuje výstupní signál $a_1y_1(n) + a_2y_2(n)$.
- Systém je neměnný v čase. To znamená, že jak vstupní signál $x(n)$ generuje výstupní signál $y(n)$, potom pro každé reálné číslo s platí, že posunutý vstupní signál v čase $\hat{x}(t) = x(t-s)$ generuje posunutý výstupní signál v čase $\hat{y}(t) = y(t-s)$.

Diferenční rovnice

První důležitý poznatek o chování lineárního, diskrétního, konečného, časově invariantního systému je, že odpověď systému na libovolný vstup je přesně dán odpovědí systému na jeden specifický vstup v čase 0, a tím je Kroneckerův impuls (definovaný v kapitole Důležité signály). Tuto odezvu systému nazýváme impulsní odezvou. Nyní si vydefinujeme vztah, který definuje výstup generovaný vstupním signálem $x(n)$.



$$y(n) = \sum_{k=0}^N a_k x(n-k) - \sum_{k=1}^N b_k y(n-k)$$

Rovnice popisuje rekurzivní LDKI (se zpětnou vazbou). Vzorok výstupního signálu jsou dány jako lineární kombinace váhovaných vzorků vstupního signálu. Váhové koeficienty jsou označeny jako a_k a b_k . Systémy popsané touto rovnicí nazýváme systémy s nekonečnou impulsní odpovědí (*infinite impulse response IIR*).

Speciálním případem je diferenční rovnice pro nerekurzivní systém. V tomto případě vstupní signál závisí jen na vzorcích vstupního signálu, není závislý na předcházejících vzorcích výstupního signálu. Systém je s konečnou impulsní odezvou (*finite impulse response FIR*) a je definován jako:



$$y(n) = \sum_{k=0}^N a_k x(n-k)$$

Konvoluce



Konvoluce je další možností, jak popsat LDKI systém. Jestliže známe impulsní odezvu systému označenou jako $h(n)$ a vstupní signál je dán jako $x(n)$, potom výstupní signál můžeme vyjádřit jako:

$$y(n) = \sum_{k=0}^{D_y-1} x(k)h(n-k) = x(n) * h(n)$$

kde operátor $*$ je konvoluční součin. Délka výstupního signálu je definována vztahem $D_y = D_x + D_h - 1$, kde D_x je délka vstupního signálu a D_h je délka impulsní odezvy.

Podstata činnosti je založena na principu superpozice lineárních systémů. Výstupní signál soustavy je dán součtem vážených a posunutých impulsních charakteristik. Pro správné porozumění si prostudujte následující příklad.



Mějme FIR systém s impulsní odezvou $h(n) = \{1, 2, 3\}$. Na jeho vstup přivedeme signál

$$x(n) = \{x(0), x(1), x(2), x(3)\}.$$

Výstupní signál určíme pomocí konvoluce.

Délka impulsní odezvy $h(n)$ je $D_h = 3$ a délka vstupního signálu $x(n)$ je $D_x = 4$, potom délka výstupu je $D_y = 6$. Konvoluci velmi jednoduše vypočítáme pomocí tabulky.

Konvoluce

| n | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|------|------|--------|--------|--------|--------|-------|---|
| x(0) | x(0) | 2 x(0) | 3 x(0) | 0 | 0 | 0 | 0 |
| x(1) | | x(1) | 2 x(1) | 3 x(1) | 0 | 0 | 0 |
| x(2) | | | x(2) | 2 x(2) | 3 x(2) | 0 | 0 |
| x(3) | | | | x(3) | 2 x(3) | 3x(3) | 0 |
| y(n) | y(0) | y(1) | y(2) | y(3) | y(4) | y(5) | 0 |

V jednotlivých řádcích tabulky se nacházejí příslušné vzorky vstupního signálu vážené impulsní charakteristikou. Posouvání řádků směrem doprava odpovídá zpoždění příslušného vzorku vstupu. V posledním řádku jsou vzorky výstupního signálu, které dostaneme jejich superpozicí v jednotlivých sloupcích pro $n = 0, 1, 2, \dots$. Například:

$$y(1) = 2 \cdot x(0) + x(1)$$

$$y(2) = 3 \cdot x(0) + 2 \cdot x(1) + x(2), \text{ atd.}$$

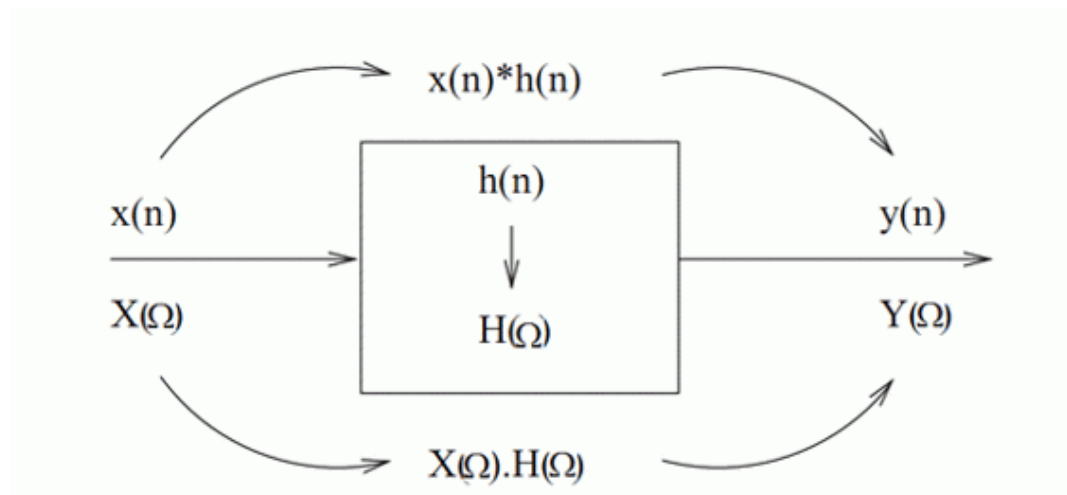
Konvoluční součin dvou posloupností dostaneme tak, že jednu z nich uspořádáme v opačném pořadí a potom jí podsouváme pod druhou zprava a v každém kroku určíme součet vzniklých součinů.

Přenosová funkce

Přenosová funkce $H(\Omega)$ reprezentuje vztah mezi vstupním a výstupním signálem LDKI systému při nulových počátečních podmínkách ve frekvenční oblasti. Přenosová funkce je odvozena z diferenciální rovnice nebo impulsní charakteristiky. V obou případech se použije DFT na transformaci do frekvenční oblasti. Jak bylo definováno výše, platí $y(n) = h(n) * x(n)$ a po DFT dostaneme vztah:



$$Y(\Omega) = H(\Omega) \cdot X(\Omega) \Rightarrow H(\Omega) = \frac{Y(\Omega)}{X(\Omega)}$$



Souvislost mezi vstupním, výstupním signálem a jejich obrazy

Je zřejmé, že $h(n)$ a $H(\Omega)$ charakterizují stejný systém v různých doménách. Jestliže aplikujeme DFT na diferenciální rovnici, přenosová funkce bude popsána pomocí váhových koeficientů a_k a b_k , které jsou identické s koeficienty v diferenciální rovnici:

$$H(\Omega) = \frac{a_0 + a_1 e^{-j\Omega} + a_2 e^{-j2\Omega} + \dots + a_N e^{-jN\Omega}}{1 + b_0 e^{-j\Omega} + b_1 e^{-j2\Omega} + \dots + b_M e^{-jM\Omega}}$$

Vztah zapsaný jako racionální lomená funkce je výhodnější, protože dělením čitatele jmenovatelem dostaneme přímo vzorky impulsní odezvy. Pro IIR systémy je počet vzorků nekonečný. V případě FIR systému není přenosová funkce ve tvaru zlomku, protože jmenovatel je rovný 1.

$$H(\Omega) = a_0 + a_1 e^{-j\Omega} + a_2 e^{-j2\Omega} + \dots + a_N e^{-jN\Omega}$$

Přenosová funkce je velmi důležitá a pomocí $H(\Omega)$ jsou určeny frekvenční charakteristiky. Ty se využívají hlavně v teorii filtrů.

Frekvenční charakteristiky

Frekvenční charakteristiky určují dynamiku systému.



Je to poměr amplitudy (nebo magnitudy) a fáze výstupu jako funkce frekvence ke vstupu.

Jednoduše řečeno, jak je na vstup systému přivedena funkce sinus s danou frekvencí, odezva LDKI bude s tou samou frekvencí, amplituda a fáze budou mít hodnoty poměrné ke vstupu.

Frekvenční charakteristika přenosové funkce $H(\Omega)$ je definována:



$$H(\Omega) = |H(\Omega)| \cdot e^{j\varphi(\Omega)} = \text{Re}\{H(\Omega)\} + j\text{Im}\{H(\Omega)\}$$

$$\text{kde } |H(\Omega)| = \sqrt{\text{Re}\{H(\Omega)\}^2 + \text{Im}\{H(\Omega)\}^2} \text{ a } \varphi(\Omega) = \arctan \frac{\text{Im}\{H(\Omega)\}}{\text{Re}\{H(\Omega)\}}.$$

Absolutní hodnota přenosové funkce $|H(\Omega)|$ se nazývá magnitudová frekvenční charakteristika a $\varphi(\Omega)$ se nazývá fázová frekvenční charakteristika. Funkce $\varphi(\Omega)$ není spojitá funkce, ale vykazuje 180 stupňové skoky. Když odstraníme tyto skoky, dostaneme fázovou frekvenční charakteristiku $\Theta(\Omega)$, která bude spojitá. Odstranění skoků umožní změna znaménka magnitudové frekvenční charakteristiky vždy při každém skoku fázové frekvenční charakteristiky $\varphi(\Omega)$. Vztah mezi amplitudovou a magnitudovou frekvenční charakteristikou je:

$$A(\Omega) = \pm M(\Omega).$$

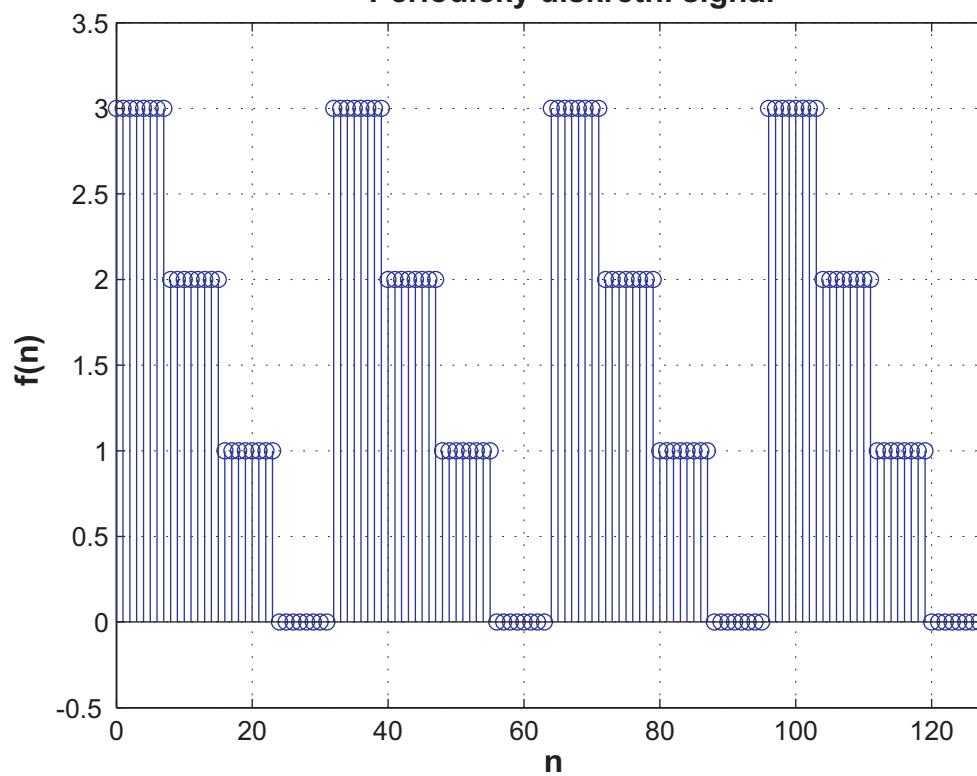
Zohledněním výše uvedených informací můžeme definovat pro frekvenční charakteristiky:



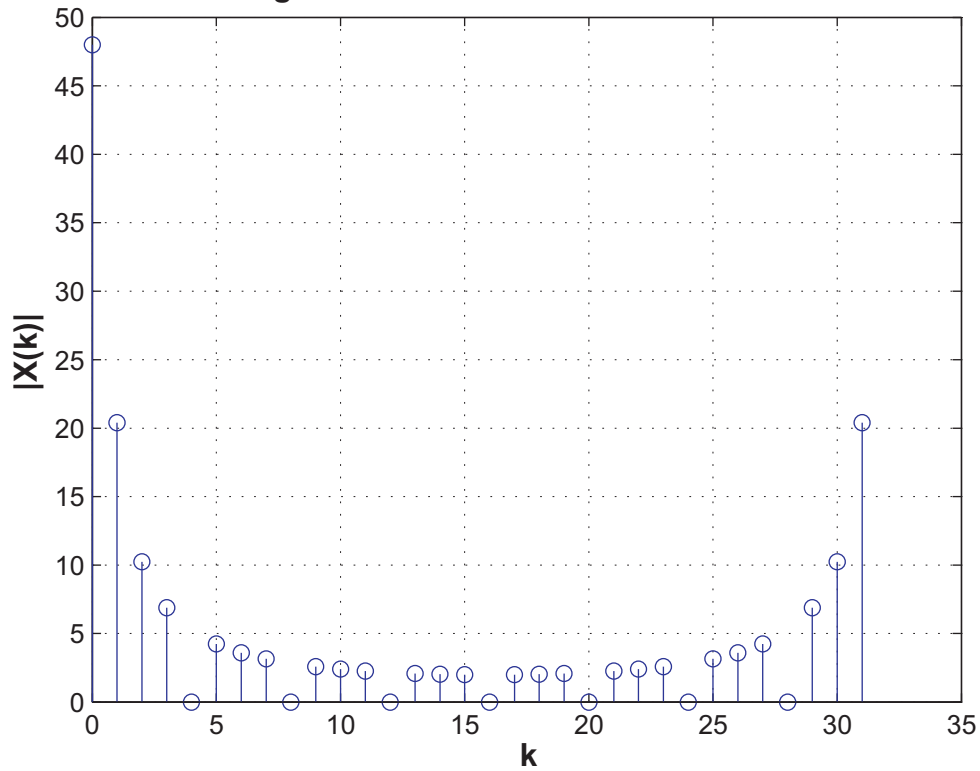
$$H(\Omega) = A(\Omega) \cdot e^{j\varphi(\Omega)}$$

$$H(\Omega) = M(\Omega) \cdot e^{j\varphi(\Omega)}$$

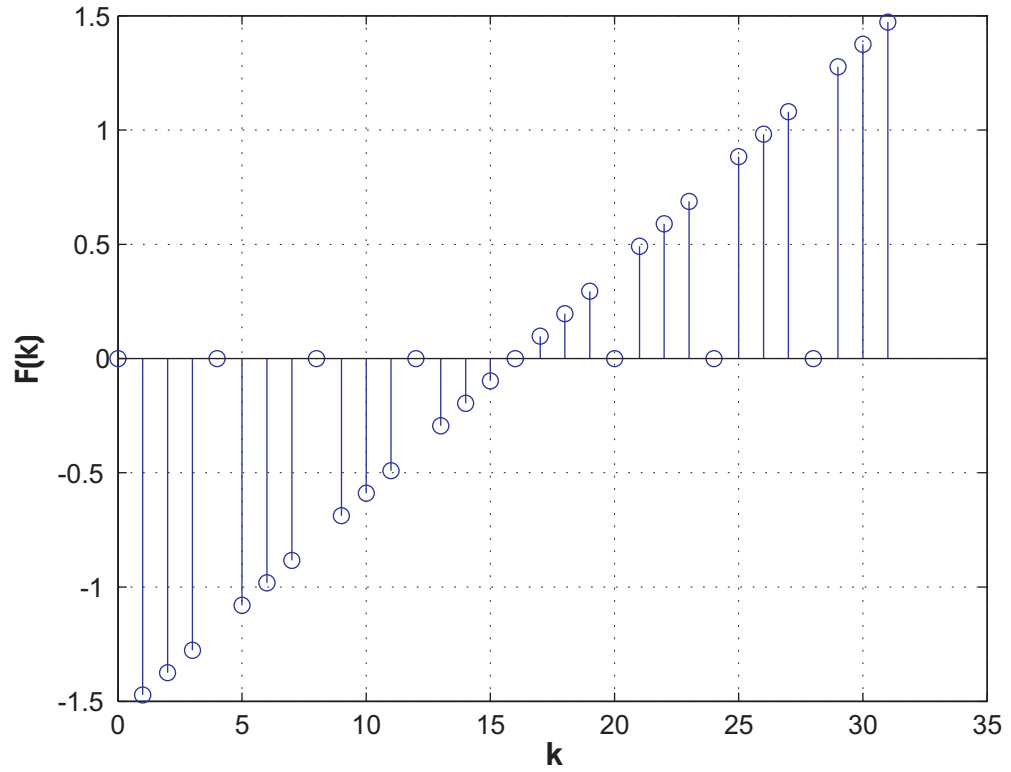
Periodický diskrétní signál



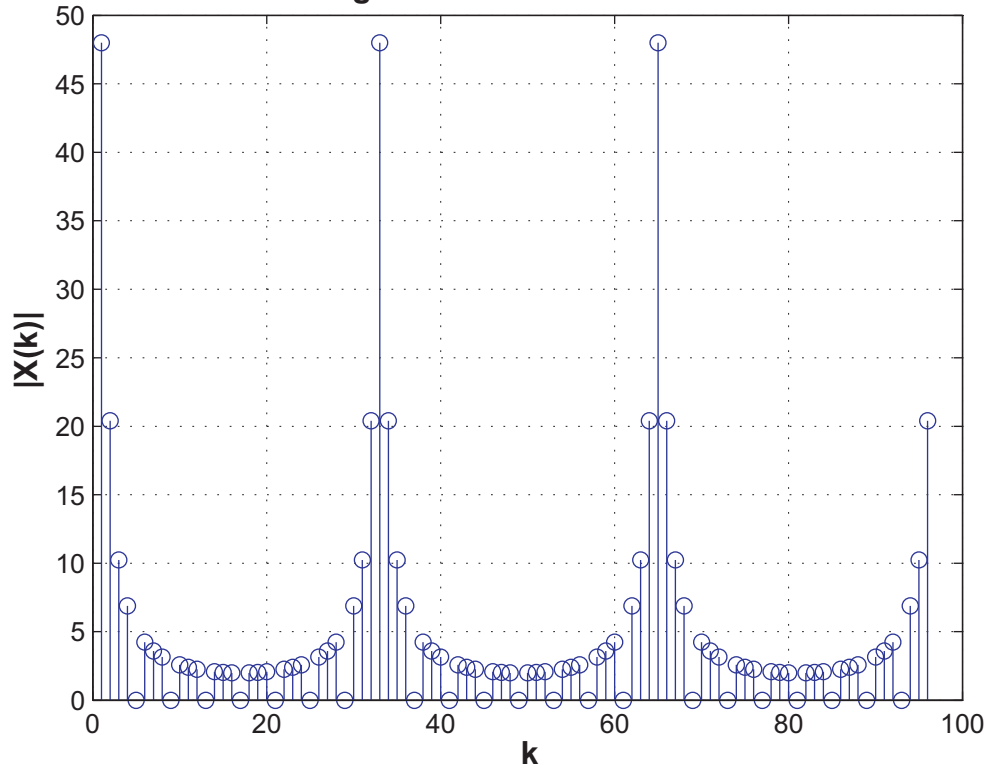
Magnitudová frekvenční charakteristika

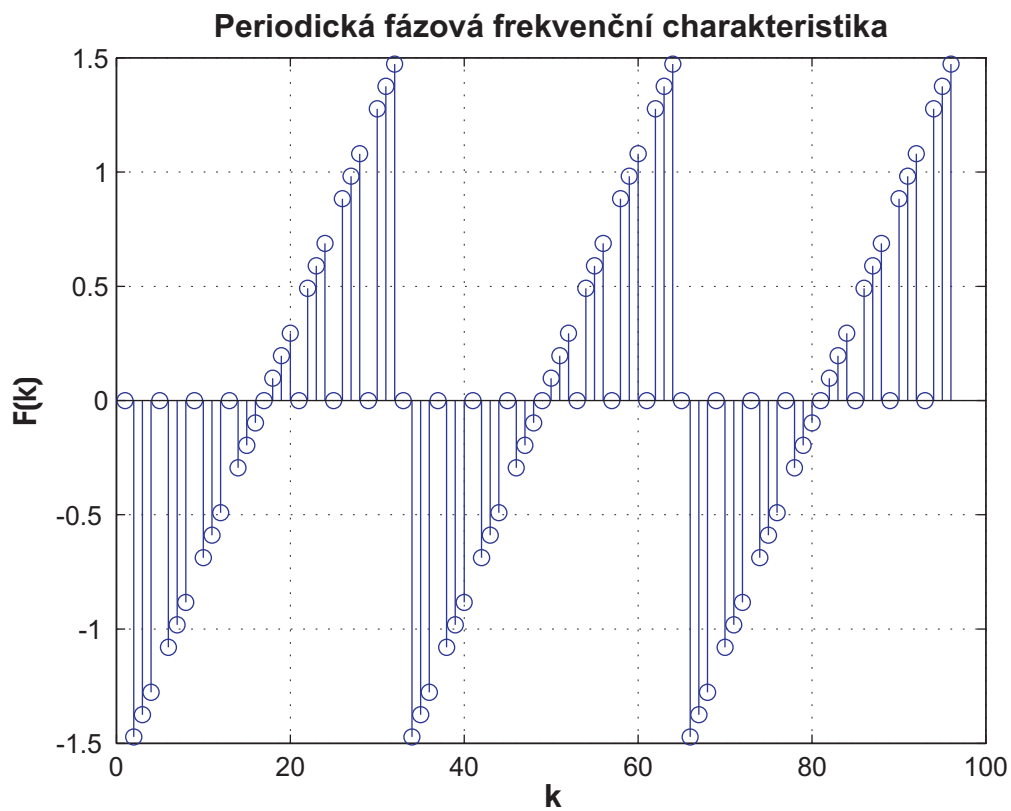


Fázová frekvenční charakteristika



Periodická magnitudová frekvenční charakteristika





Příklad přenosové funkce LDKI systému, magnitudová a fázová frekvenční charakteristika (jedna a více period)

3.3 Filtry



Při zpracování signálů, je filtr zařízení nebo proces, který odstraní ze signálu nechtěnou část nebo vlastnost.

Nejčastěji to znamená odstranění některých frekvencí kvůli potlačení rušení signálů a snížení šumu v pozadí. Avšak filtry nepracují jen ve frekvenční doméně, obzvláště v oblasti zpracování obrazu je více veličin, které se mohou filtrovat.

Nedostatkem filtrace je ztráta informace spojené s odfiltrovanou částí. Kombinace signálu s Fourierovým prostorem je alternativní přístup na odstranění pouze určitých frekvencí ze signálu.



Existuje několik různých klasifikací filtrů, které se v mnohých kritériích překrývají. Neexistuje tedy jednoznačné dělení. Filtry tedy můžeme dělit na:

- analogové nebo digitální
- diskrétní v čase nebo diskrétní v hodnotě
- lineární nebo nelineární
- s nekonečnou impulsní odezvou (IIR) nebo s konečnou impulsní odezvou (FIR) patřící mezi diskrétní a digitální filtry

Důležité pojmy používané v klasifikaci lineárních filtrů:

- Dolní propust – nízké frekvence se propouštějí, vysoké frekvence se tlumí.
- Horní propust – vysoké frekvence se propouštějí, nízké frekvence se tlumí.
- Pásmová propust – propouštějí se pouze frekvence definovaného pásma.
- Pásmová zadrž – tlumí se pouze frekvence definovaného frekvenčního pásma.
- Úzkopásmový filtr – tlumí pouze jednu definovanou frekvenci (speciální typ pásmové zadrž).

Filtry mohou být implementovány různými technologiemi. Stejná přenosová funkce se může realizovat více způsoby. To znamená, že matematické vlastnosti a parametry jsou stejné, ale fyzická realizace je odlišná. Komponenty v jednotlivých technologiích jsou však obvykle stejné a v příslušných filtrech plní stejný úkol.

- Elektronické filtry byly původně vyrobeny pouze z pasívních prvků (odporníky, indukčnosti a kondenzátory). Aktivní technologie zjednodušily výrobu a otevřely i nové možnosti ve specifikaci filtrů.
- Digitální filtry pracují se signály definovanými v digitální podobě. Podstata digitálních filtrů spočívá v tom, že mají přímo v mikro kódu implementován matematický algoritmus odpovídající požadované přenosové funkci.

Lineární analogové filtry



Lineární obvod popsaný spojitými funkcemi je pravděpodobně nejčastěji myšlený jako filtr v oblasti zpracování signálů. Pro zjednodušení se zavedl pojem filtr. Úkolem těchto filtrů je určité frekvence tlumit a jiné propouštět.

Takový filtr musí být nutně lineárním filtrem. Nelinearita by mohla způsobit to, že se ve výstupním signálu budou vyskytovat i frekvence, které se ve vstupním signálu nenacházely.

Nejznámější skupiny lineárních, časově spojitých filtrů, dělené podle způsobu návrhu jsou:

- Chebyshev filtr, má nejlepší aproximaci vzhledem k ideální charakteristice libovolného filtru pro dané uspořádání a zvlnění.
- Butterworth filtr, má maximálně vyrovnanou frekvenční charakteristiku.
- Bessel filtr, má maximálně vyrovnané fázové zpoždění.

Rozdíl mezi těmito skupinami filtrů spočívá v tom, že každý typ používá jinou polynomickou funkci pro aproximaci ideální frekvenční charakteristiky. Výsledkem jsou různé přenosové funkce. Takže podle toho, jakou frekvenční charakteristiku potřebujeme (zvlněnou, vyrovnanou) si vybereme typ filtru.



Filtry se používají především v oblasti telekomunikací – jsou nezbytnou součástí mnohých technologických pokroků a byly zdrojem značných zisků pro telekomunikační společnosti. Nemělo by tedy být žádným překvapením, že vývoj prvních filtrů byl úzce spojen s vývojem přenosových tras.

Digitální filtry



V elektronice, výpočetní technice a matematice se za digitální filtr považuje systém, který pracuje jen s navzorkovaným signálem (diskrétním v čase). Jeho podstatou je zredukovat nebo zvýraznit některé vlastnosti signálů.

To je hlavním rozdílem v porovnání s analogovými filtry, které reprezentuje elektrický obvod pracující se signálem spojitým v čase. Analogový signál může být filtrován digitálním filtrem v případě, když se nejdříve navzorkuje a je reprezentován jako posloupnost vzorků. Potom může projít digitálním filtrem a zpětně se zrekonstruovat na profiltrovaný analogový signál. U analogových filtrů je vstupní signál přímo přiveden do elektrického obvodu.



Digitální filtr je charakterizován svojí přenosovou funkcí anebo diferenční rovnicí. Matematickou analýzou přenosové funkce se dá vyjádřit odpověď na libovolný vstup.

Filtry s konečnou impulsní odezvou



Filtr s konečnou impulsní odezvou, se zkratkou FIR – je filtr, který má konečnou impulsní odezvu, protože se po určité době ustálí na nule.

FIR jsou typické filtry bez zpětné vazby, takže výstupní signál závisí pouze na vstupním signálu. Jestliže máme N vzorků vstupního signálu, potom $N - 1$ je stupeň filtru.



Diferenční rovnice popisující systém v časové oblasti:

$$y(n) = \sum_{k=0}^N a_k x(n-k)$$

Definice přenosové funkce:

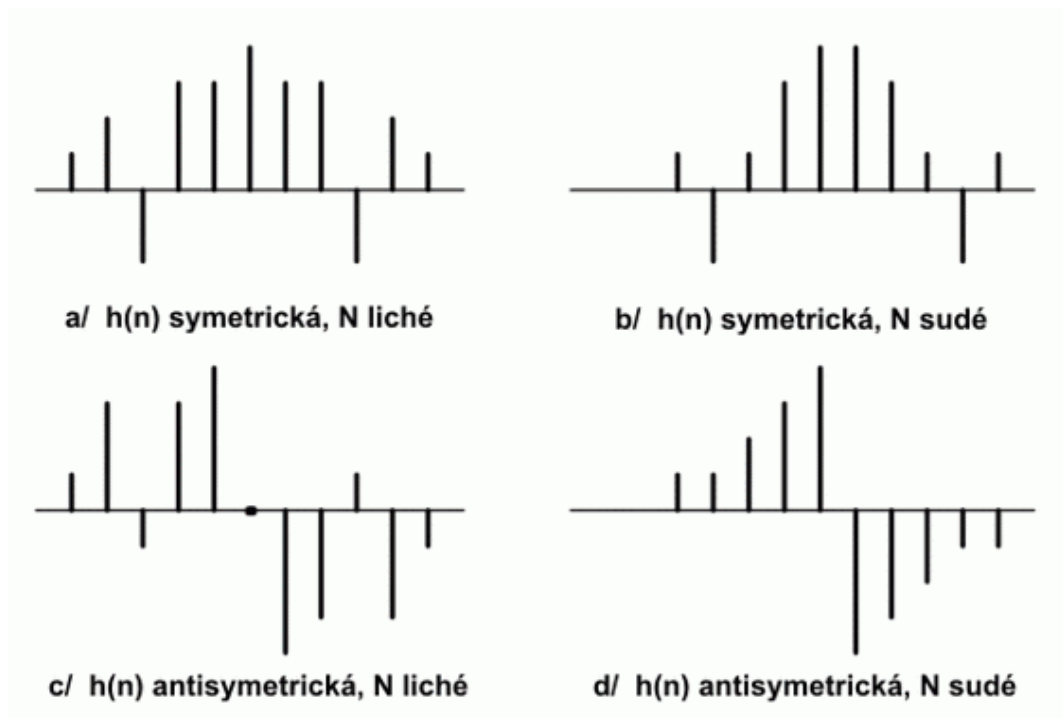
$$H(\Omega) = a_0 + a_1 e^{-j\Omega} + a_2 e^{-j2\Omega} + \dots + a_N e^{-jN\Omega}$$



Hlavní výhody FIR jsou tyto:

- Nedochozí k akumulaci chyby při sčítání vzorků, protože nemají zpětnou vazbu.
 - Jednoduchá implementace.
 - Stabilita. Jsou-li bez zpětné vazby, pak všechny póly* jsou umístěny v začátku (zjednodušeně řečeno „jakoby póly nebyly“). Systém je stabilní právě tehdy, když absolutní hodnota každého pólu je menší než jedna.
 - Jednoduše se dá zabezpečit lineární fázovou charakteristiku. Ta je zabezpečena tehdy, když je impulsní odezva systému symetrická anebo antisymetrická.
-

*Poznámka: póly jsou kořeny jmenovatele přenosové funkce.

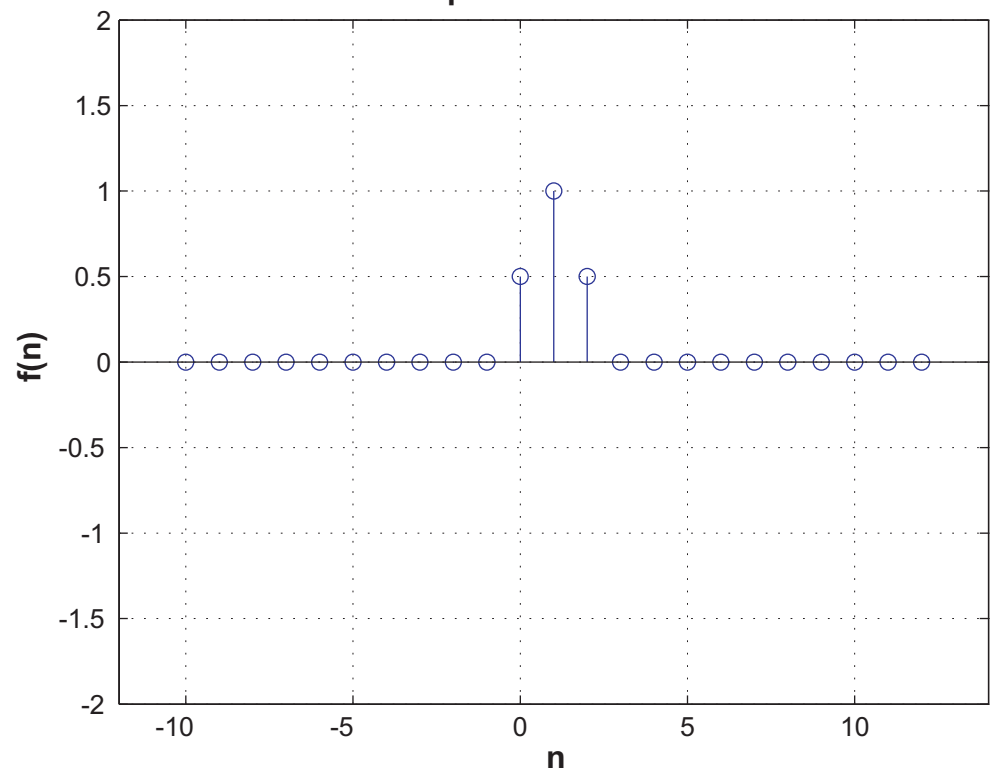


Impulsní odezva (pro zabezpečení lineární fázové charakteristiky)

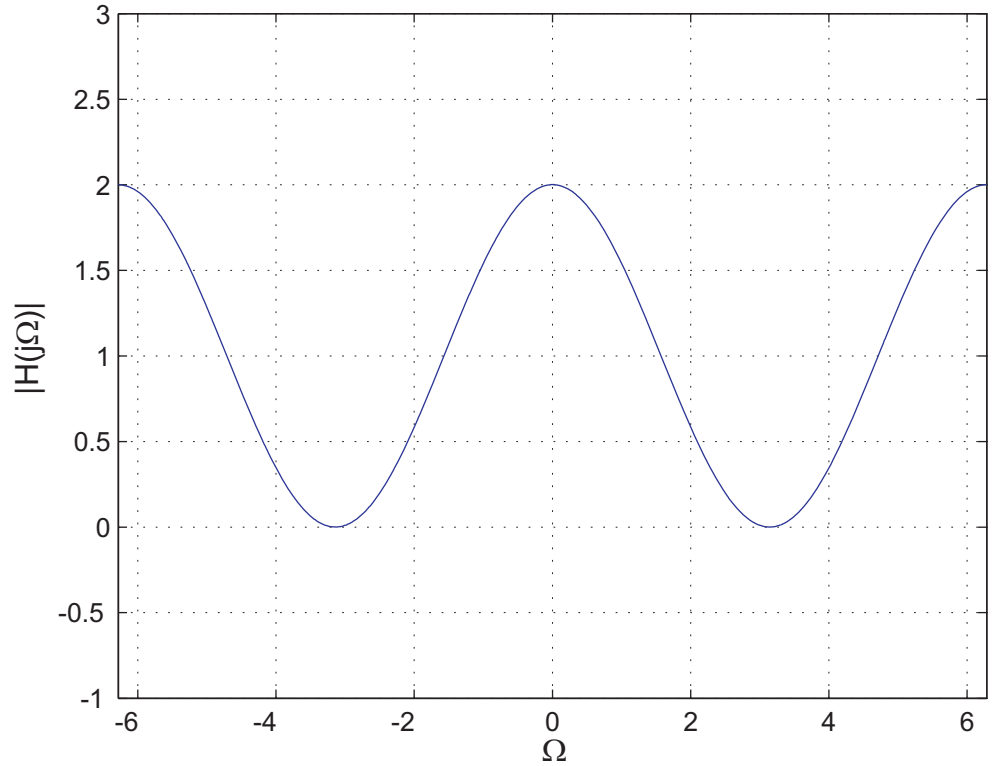
Při návrhu filtru musíme vybrat takové koeficienty, aby filtr měl požadované charakteristiky. Ty jsou obvykle uvedeny ve specifikaci filtru, která nejčastěji představuje jeho frekvenční charakteristiku. Existuje několik metod na hledání koeficientů - mezi nejvíce používané patří:

- Intuitivní metoda
- Metoda frekvenčního vzorkování
- Metoda s okenními funkcemi
- Metoda vážením nejmenších čtverců

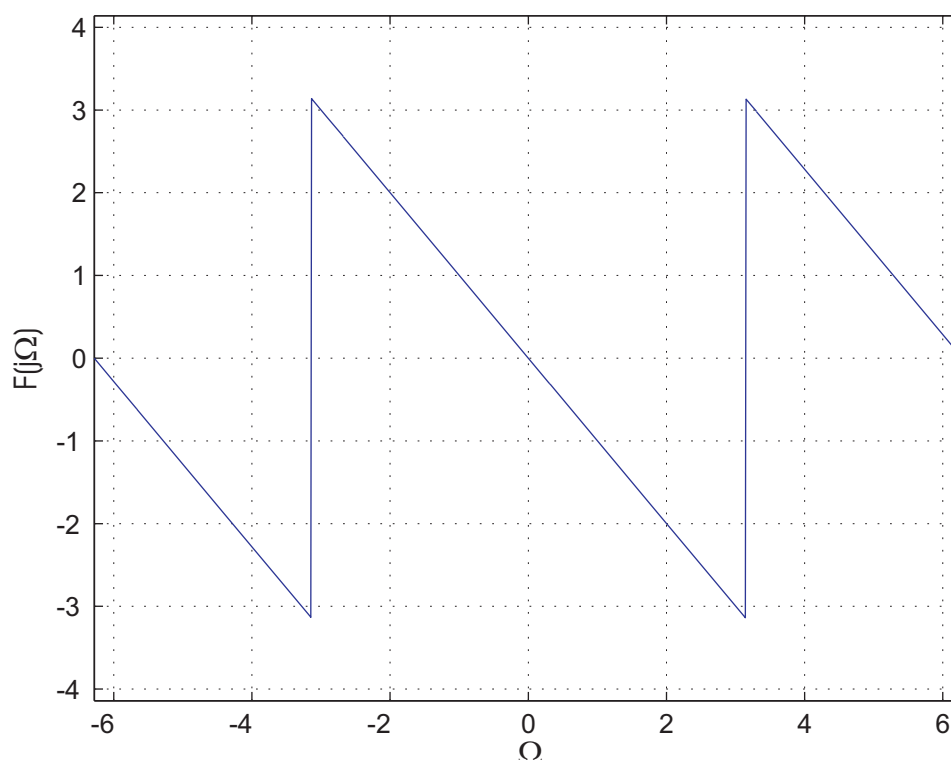
Impulsová odezva



Magnitudová frekvenční charakteristika



Fázová frekvenční charakteristika



Impulzová odpověď, magnitudová a fázová frekvenční charakteristika FIR filtru

Filtry s nekonečnou impulsní odezvou



Filtr s nekonečnou impulsní odezvou, se zkratkou IIR – je filtr, jehož impulsní odezva nikdy nenabyde nulové hodnoty.

Vzorek výstupního signálu je dán sumou N vzorků vstupního signálu, váženými koeficienty a_k a výstupního signálu, váženými koeficienty b_k . Je to zřejmé také diferenční rovnice, která popisuje IIR systém.



Diferenční rovnice popisující IIR filtr:

$$y(n) = \sum_{k=0}^N a_k x(n-k) - \sum_{k=1}^N b_k y(n-k)$$

a definice přenosové funkce:

$$H(\Omega) = \frac{a_0 + a_1 e^{-j\Omega} + a_2 e^{-j2\Omega} + \dots + a_N e^{-jN\Omega}}{1 + b_1 e^{-j\Omega} + b_2 e^{-j2\Omega} + \dots + b_M e^{-jM\Omega}}$$

Přenosová funkce je dána podílem dvou polynomů, proto stabilita soustavy není zaručena. Jak bylo již uvedeno, systém je stabilní tehdy, když absolutní hodnota každého pólu je menší než jedna. V případě nestability existuje několik způsobů,

jak soustavu stabilizovat, například pomocí fázovacích článků anebo PLSI algoritmem.

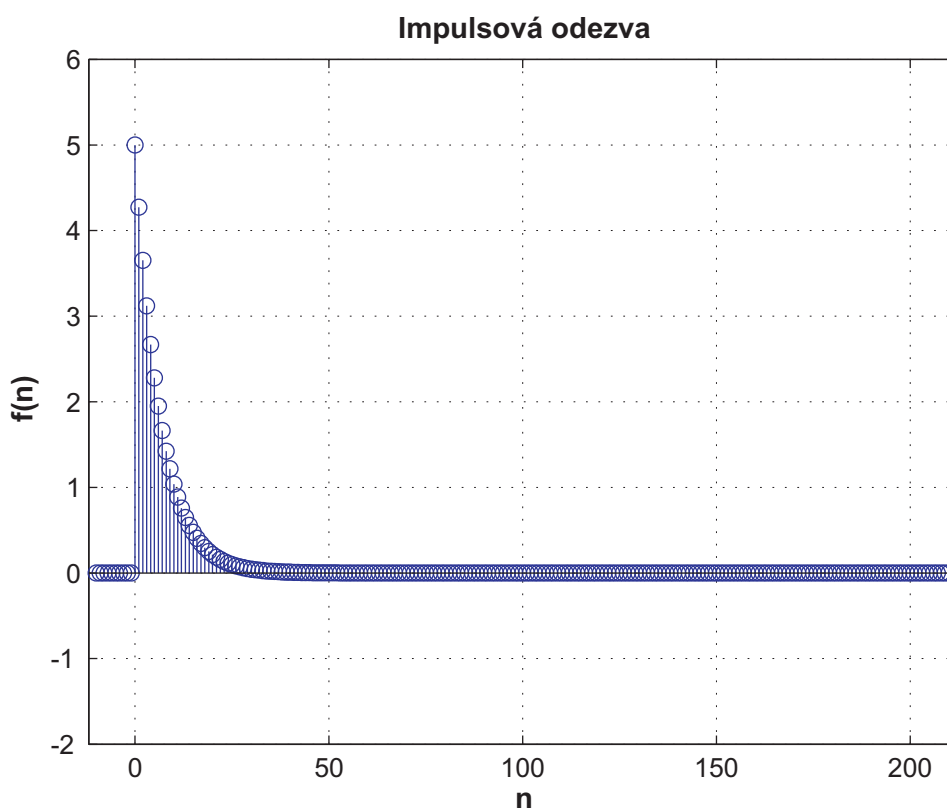
Při návrhu filtru musíme zvolit koeficienty tak, aby systém měl předepsanou frekvenční charakteristiku. V případě IIR filtru to znamená určit stupeň čitatele, jmenovatele a koeficienty a_k a b_k . Metody návrhu IIR filtrů se dělí do dvou skupin.

Do první skupiny patří přímé metody návrhu:

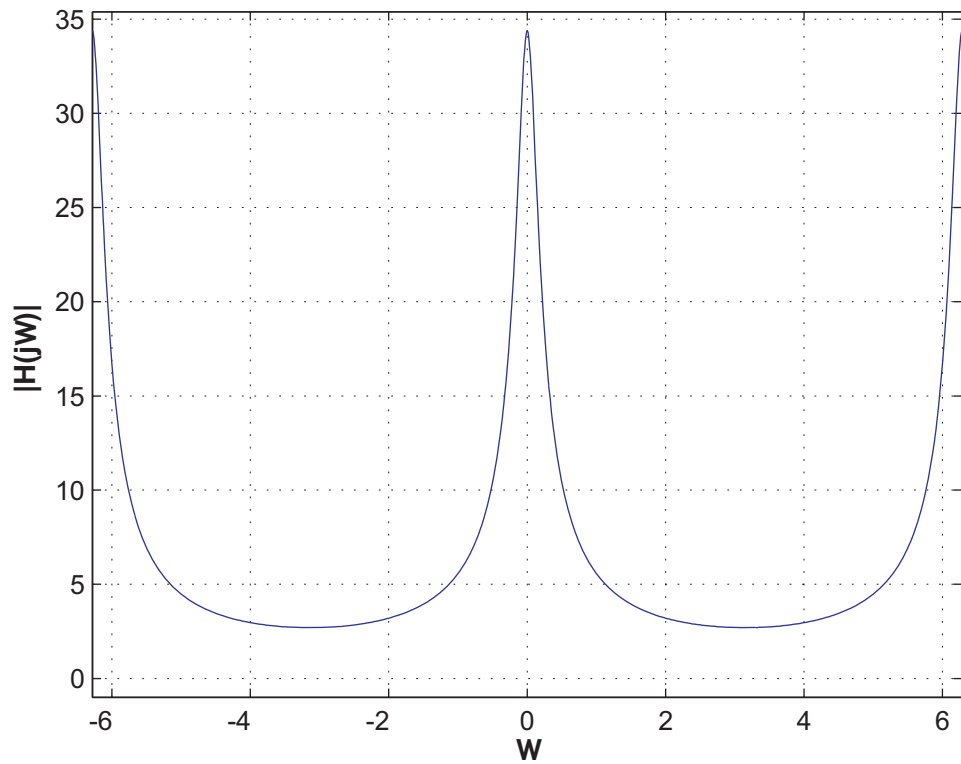
- Intuitivní metoda
- Metoda frekvenčního vzorkování
- Pronyho metoda

Druhá skupina metod návrhu se nazývá nepřímé metody a vychází z přepracovaných metod návrhu analogových filtrů. Při návrhu jsou formulovány požadavky na IIR filtr. Potom se navrhne analogový filtr (Chebyshev, Butterworth, Eliptický filtr), který je přetransformován do digitální oblasti aplikováním diskretizačních technik. Mezi nejpoužívanější patří:

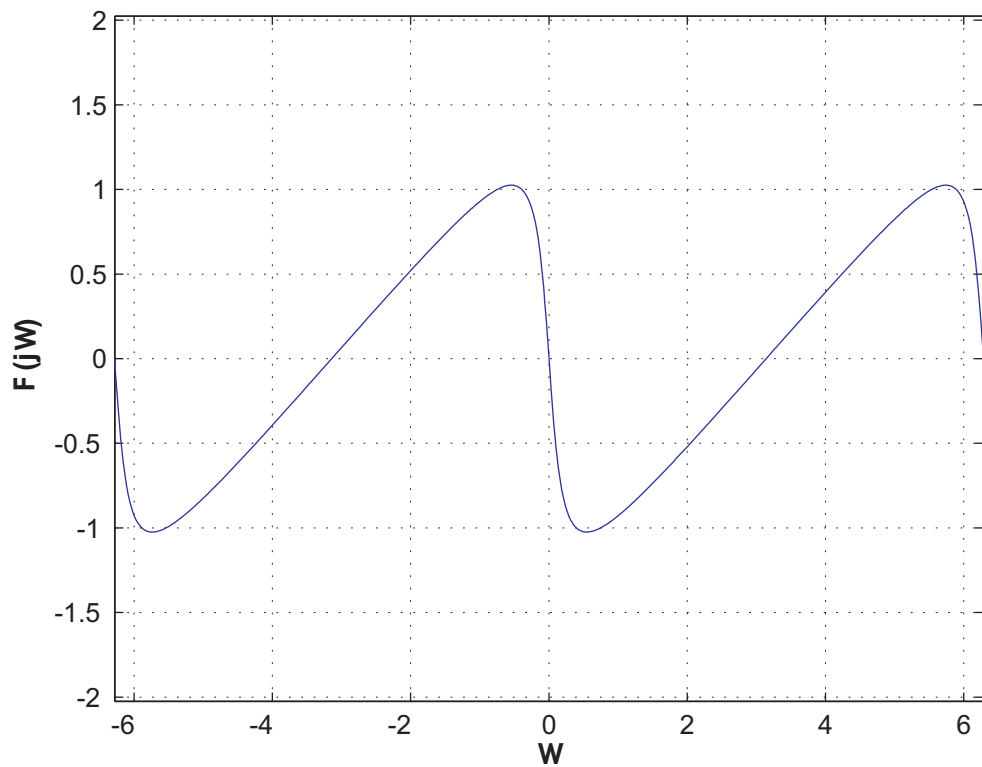
- Metoda bilineární transformace
- Metoda invariantnosti impulsní charakteristiky



Magnitudová frekvenční charakteristika

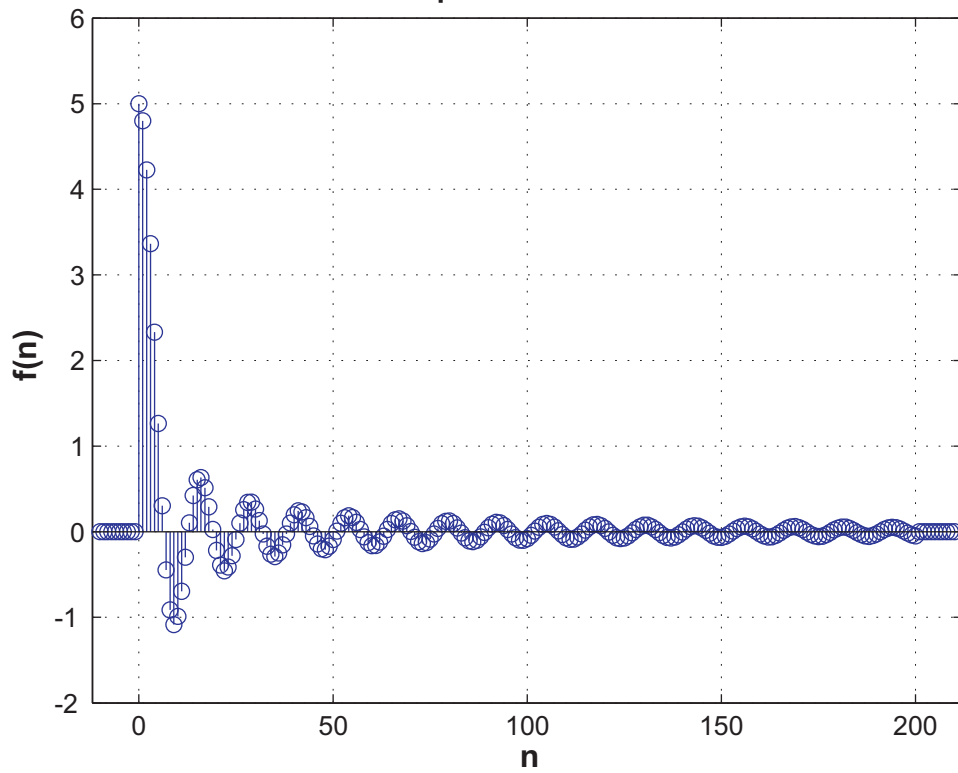


Fázová frekvenční charakteristika

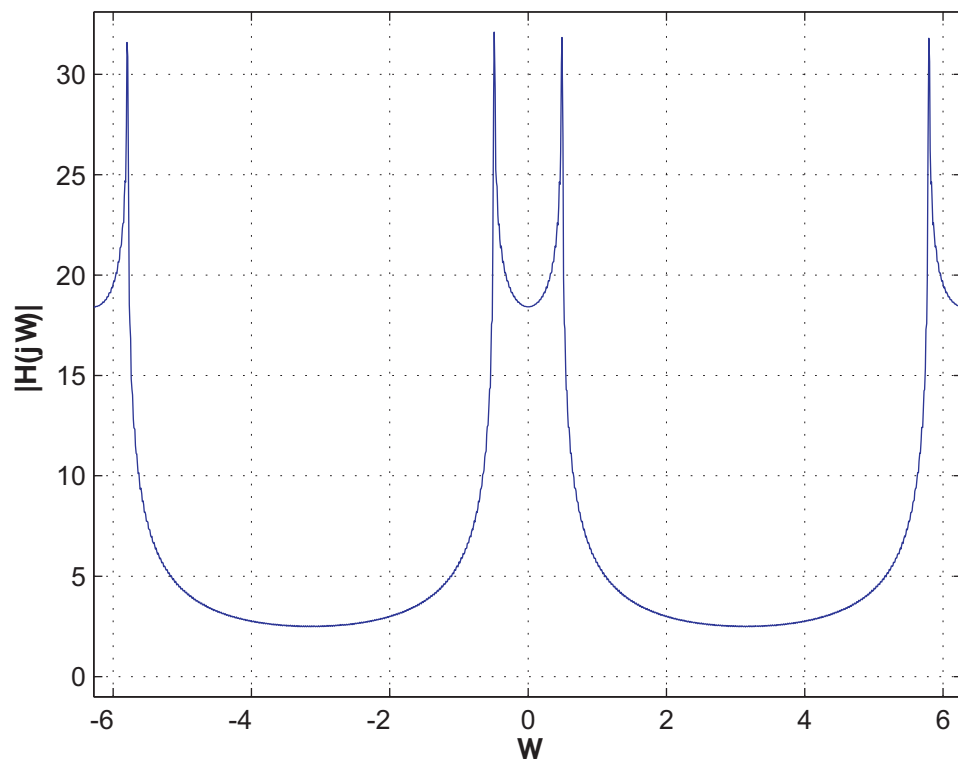


Impulsní odezva, magnitudová a fázová frekvenční charakteristika IIR filtru (exponenciální funkce)

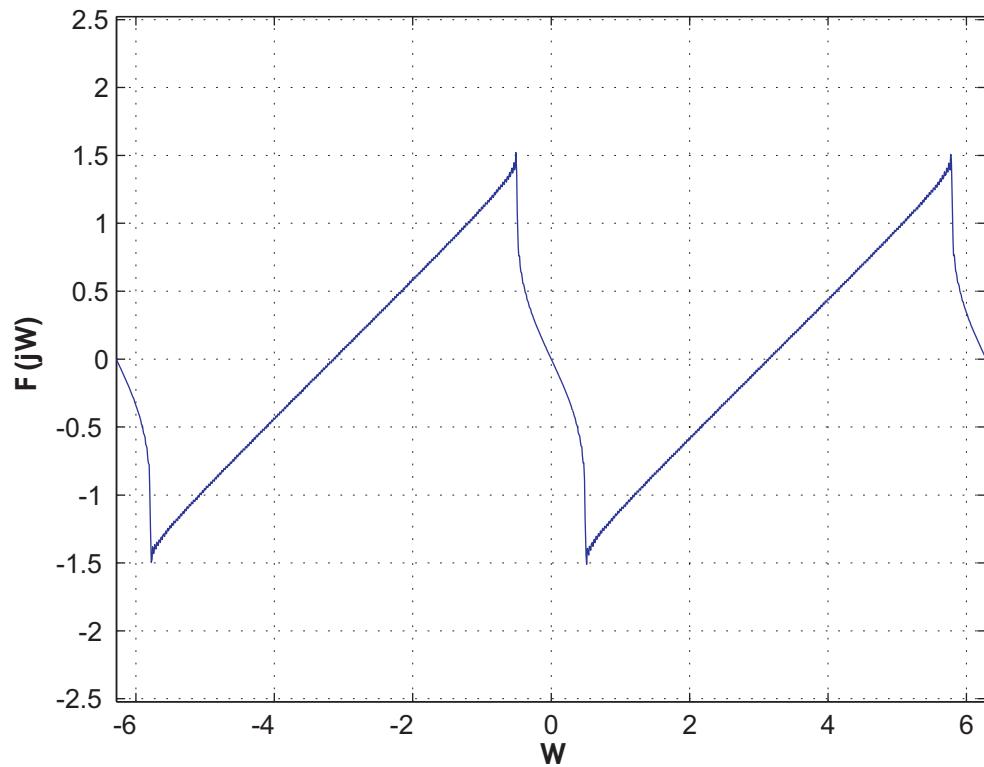
Impulsová odezva



Magnitudová frekvenční charakteristika



Fázová frekvenční charakteristika



Impulsní odezva, magnitudová a fázová frekvenční charakteristika IIR filtru (sinc funkce)

-
- + Hlavní výhodou IIR filtrů je skutečnost, že přes rekurzivitu mají méně členů, a tedy spotřebují méně operační paměti než ekvivalentní FIR filtry. Proto jsou výhodné v oblasti číslicového zpracování signálů.
-
- Nevýhodou IIR filtrů je okolnost, že mohou být nestabilní. Implementace IIR filtrů je komplikovanější než implementace FIR filtrů.
-

4 Telekomunikační spoj

4.1 Telekomunikační spoj

Typický komunikační systém obsahuje vysílací systém (vysílač), přijímací systém (přijímač) a přenosovou cestu (přenosové médium) což v souhrnu tvoří telekomunikační spoj. Telekomunikační kanál pak představuje soubor technických prostředků umožňujících pouze jednosměrný přenos signálu mezi dvěma místy bez ohledu na druh použitých prostředků.



Telekomunikační okruh tvoří pár vzájemně přiřazených protisměrných kanálů umožňující protisměrnou komunikaci. Protisměrná komunikace může být buď simplexní (přenos signálu okruhem střídavě v jednom či druhém směru) anebo duplexní (přenos signálu okruhem současně v obou směrech).

Vzhledem k přenášenému signálu rozlišujeme dva druhy komunikačních přenosů - analogový a digitální. V následující části budou popsány oba systémy a jejich základní funkční bloky.

Analogový komunikační přenos



Analogová komunikace je způsob přenášení informací, který využívá spojitý signál pro přenos zpráv jako hovor, hudba, obraz statický i video, atd.

Analogový přenos umožňuje přenášet zprávy od jednoho bodu k jinému bodu anebo k více bodům. Pro přenos dat pomocí analogových signálů se využívá proces modulace. Proto je nutné, aby vysílač i přijímač obsahoval kompatibilní zařízení – vhodný modulátor na straně vysílače a demodulátor na straně přijímače.



Zařízení, které se používá při vysílání i přijímání datového signálu, obsahuje v sobě modulátor i demodulátor a nazývá se MODEM (MOdulátor+DEModulátor).



Analogové obvody nezahrnují v sobě kvantizaci informace jako digitální obvody, z čehož plyne jejich velká nevýhoda náhodných chyb a poškození (degradace) signálu, které neumíme opravit. Výsledkem je velké zašumění zvuku anebo videa. Platí, čím větší vzdálenost, tím větší zašumění.

Digitální komunikační kanál

Vysílač

I když je to číslicový komunikační systém, je schopný zpracovat i analogový signál (například hovor), který je *analogově-číslícovým převodníkem (analog-digital convertor ADC)* konvertován do číslicového tvaru. V tomto analogově-číslícovém převodníku probíhá proces digitalizace signálu. Detailní postup je vysvětlen v kapitole Digitalizace analogového signálu.

Dalším blokem, který už pracuje s digitálním signálem, je zdrojový kodér, jehož obecnou vlastností je to, že efektivně konvertuje každý diskretní symbol do vhodné číslicové reprezentace (obvykle binární). Je nutné poznamenat, že v praxi obvykle bývá proces prvotního přiřazení číslicové reprezentace součástí ADC převodníku. Digitální zpráva ve většině případů v sobě zahrnuje redundanci (nadbytečnost).

- + Principiální vlastností zdrojového kodéru je to, že odstraňuje redundanci. Efektivnější kodér účinněji odstraňuje redundanci, což v konečném důsledku vede k menšímu počtu bitů (binárních číslic „0“ a „1“) potřebných pro přenos informace. Jako příklad odstranění nadbytečnosti je možno uvést: „a tak dále“, v psaném textu se často používá „atd.“.

V některých systémech, které nedisponují funkcí kanálového kódování, je přímo výstup zdrojového kodéru transformován do tvaru, který je vhodný pro přenos kanálem. Jak bylo uvedeno výše, je úlohou kodéru redukovat nadbytečnost v digitálním signálu. Naopak je to při kanálovém kódování, jehož úkolem je přidat nadbytečnost, která slouží na detekci a opravu chyb. Bez redundantní informace by nebylo možno zjistit a odstranit chyby způsobené šumem a interferencí. Nejjednodušší formy redundance jsou opakování a parita. Posledním zařízením na straně vysílače je modulátor.



V případě digitální modulace, je analogový nosný signál modulován diskretním signálem.

Druh použité modulace závisí na typu kanálu a přenášených dat. Některé modulátory jsou určeny pro typicky zašuměné kanály (např. rádiové), některé jsou naopak určeny pro kanály s minimálním rušením (např. optické kabely).

Přijímač



V přijímači se realizují operace inverzní k operacím vysílače. Velmi důležitou úlohou při návrhu a optimalizaci digitálního komunikačního systému je minimalizovat chyby vzniklé při přenosu.

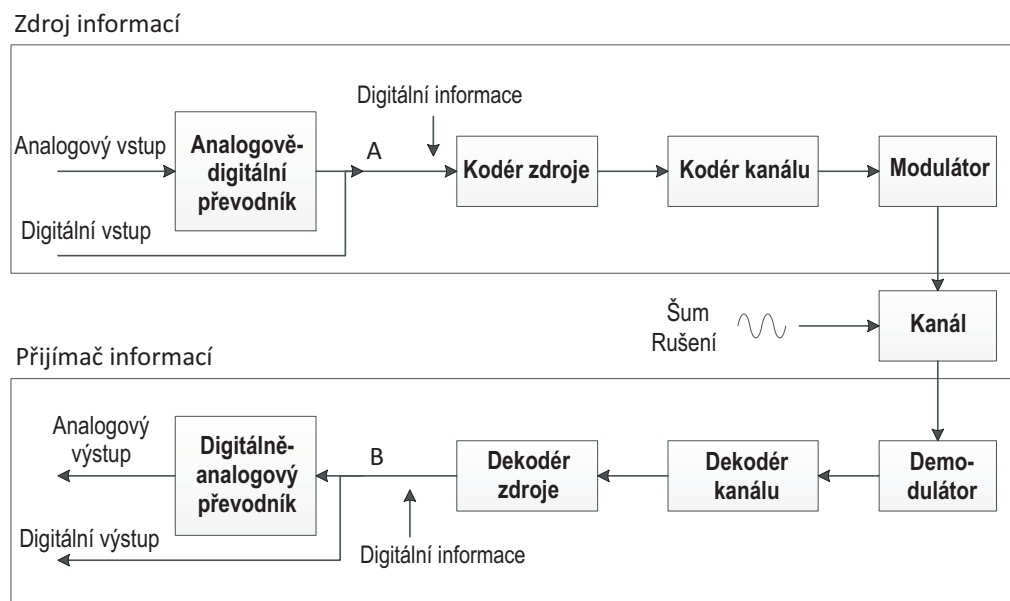
V přijatém signálu se obvykle vyskytují chyby zapříčiněné šumem kanálu a tyto chyby musí být opraveny na straně přijímače. Toto je úkolem demodulátoru, který mění numerické hodnoty na hodnoty signálu (např. číslo 1 odpovídá hodnotě

signálu 5, kde 5 může být ampér, volt, atd.). Podle toho, jak jsou přenášeny parametry (amplituda, fáze a frekvence) v nosném signálu, se vybere způsob demodulace. Když je signál modulován lineární modulací jako AM, použije se synchronní detektor. V případě úhlové modulace je nutné použít FM demodulátor anebo PM demodulátor. Tyto demodulátory jsou realizovány různými elektrickými obvody.

Posláním kanálového dekodéru je rekonstruovat co nejpřesněji, výstup zdrojového kodéru vysílače. V praxi se v přenášeném signálu zpravidla vždy vyskytnou chyby. Redundantní informace přidaná kanálovým kódem vysílače mu však umožňuje korigovat tyto chyby vnášené kanálem a demodulátorem (což není možné při analogovém přenosu).

Zdrojový dekodér realizuje přesně inverzní funkci, kterou vykonával zdrojový kódér vysílače, to znamená, že přidává předtím odstraněnou redundanci. V případě, kdy výstup má zůstat digitální, pak výstup ze zdrojového dekodéru je výstupem celého přijímače.

Když potřebujeme analogový výstup (např. hovor), pak údaje ze zdrojového dekodéru postupují do *číslicově-analogového převodníku* (*digital-analog convertor DAC*). Číslicově-analogový převodník v souladu s teorií vzorkování rekonstruuje původní analogovou zprávu.



Blokové schéma digitálního komunikačního systému

5 Kompresní metody



Cílem komprese je odstranění nepodstatných anebo nadbytečných informací z původních dat tak, abychom data mohli ukládat nebo přenášet efektivněji: ušetříme tak např. přenosový kanál anebo úložný prostor. Kompresí kódujeme užitečnou informaci menším počtem bitů, oproti původní reprezentaci.

Množství dat, které v současnosti vytvoří multimediální zařízení, je obrovské. Kdybychom chtěli uložit 6megapixelový statický obrázek (3000x2000 pixelů s 8 bity/barva) na USB klíč v nekomprimovaném tvaru, potřebovali bychom více než 17 megabyte prostoru (přesně 18 miliónů byte). Stejný obrázek komprimovaný pomocí kodéru JPEG by zabral okolo 3 Megabyte úložného prostoru (asi 3 200 000 byte) a nepostřehli bychom žádný rozdíl. Stejně jako pro statické obrázky to platí i pro audio a video signály, a to jak při ukládání, tak i při přenosu digitálními sítěmi.

Kompresní algoritmy můžeme dělit na:

- Bezeztrátové kódování
- Ztrátové kódování



Bezeztrátové kódování snižuje počet bitů hledáním a odstraňováním statistické nadbytečnosti, tedy informací, které se opakují anebo se mohou získat z jiných informací v signálu. Bezeztrátová komprese (označovaná též jako kompakce) umožňuje rekonstrukci původního signálu bez jakýchkoliv změn. Je proto vhodná pro archivační účely.



Na druhé straně ztrátová komprese se snaží snížit množství dat odstraňováním informací, které člověk kvůli nedokonalostem svých smyslů nemůže zaregistrovat. Při použití této metody komprese není rekonstruovaný signál nikdy stejný jako původní signál. Metody jsou však vhodné na aplikace, kde jsme omezeni velikostí úložného prostoru anebo šířkou přenosového kanálu.

Obecně jsou ztrátové algoritmy mnohem efektivnější než bezeztrátové algoritmy, ale zrekonstruovaný signál se vždy liší od toho původního. Kvalita komprimovaného signálu je tedy kompromisem mezi množstvím informací odstraněných z původního signálu a požadovanou velikostí souboru komprimovaného signálu.

Míra změn v signálu, které jsou ještě akceptovatelné bez pozorovatelného rozdílu oproti původnímu signálu, se nazývá právě pozorovatelné zkreslení (just-noticeable distortion) u zvuku a hovoru anebo právě pozorovatelný rozdíl (just-noticeable difference - jnd) u statických obrázků a videa.

Kompresní poměr, vyjadřující efektivitu kompresní metody, je dán vztahem:

$$C_r = \frac{N_n}{N_k},$$

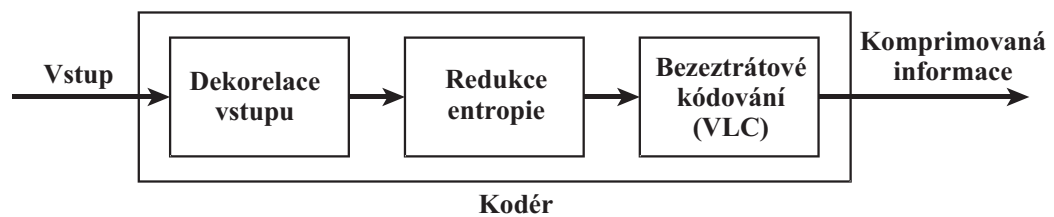
kde N_n je počet bitů nekomprimovaného signálu a N_k je počet bitů komprimovaného signálu.

Programové vybavení, provádějící kódování původního signálu do jeho komprimované podoby a dekódování komprimovaného signálu do původní podoby, se nazývá kodek (zkratka z **kodér-dekodér**).

Model komprese

Proces komprese se skládá z třech základních fází:

1. Dekorelace vstupních dat
2. Redukce entropie
3. Bezeztrátové kódování



Obecné kompresní schéma

Dekorelace vstupních dat odstraňuje ze signálu duplicitní informace. Může se provést různými způsoby, resp. v různých oblastech reprezentace signálu:

- Časová (anebo prostorová) oblast
- Parametrická oblast
- Rozklad signálu

V časové (anebo prostorové) oblasti jsou metody založené na lineární predikci, která předpokládá, že sousední vzorky v čase (anebo prostoru) jsou korelované (podobají se). Přístup pomocí rozkladu signálu rozkládá signál na subpásma, ve kterých je možné samostatně sledovat energii. Do parametrické oblasti přecházíme různými transformacemi tak, abychom dokázali získat charakteristické parametry pro kódování a přenos vstupních dat.

Proces kvantizace redukuje entropii. I když kvantizace vždy způsobuje ztrátu informací, dostatečně přesná kvantizace tuto ztrátu snižuje na minimum.

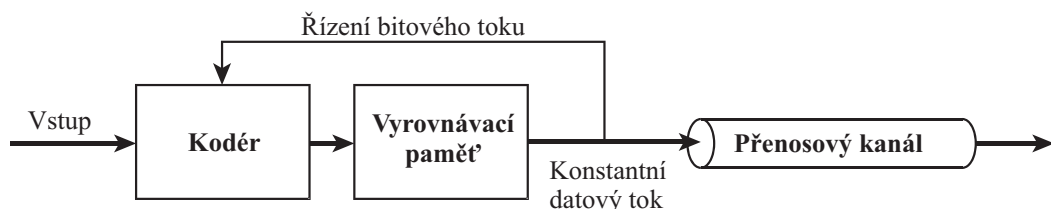
Posledním krokem v procesu kódování je bezeztrátové kódování, nazývané též entropické kódování. Během tohoto procesu se statisticky častým kombinacím

bitů (symbolům) přiřadí kratší kódové slovo a statisticky méně častým kombinacím bitů delší kódové slovo. Takové kódování se též nazývá kódování s variabilní délkou (variable length coding - VLC), protože kódová slova mají různou délku. Nejeftivnějším algoritmem k provádění bezztrátového VLC entropického kódování je tzv. Huffmanovo kódování.

Na přenos komprimované informace prostřednictvím přenosového kanálu existují dva režimy:

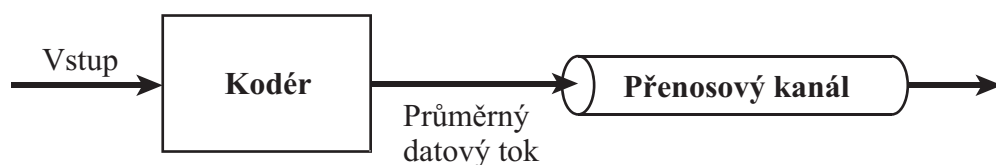
- Konstantní datový tok
- Proměnlivý datový tok

Při použití konstantního datového toku se výstup kodéru ukládá do vyrovnávací paměti, ze které se zabezpečuje konstantní datový tok do přenosového kanálu. Datový tok ovlivňuje kodér tak, aby produkoval stejný datový tok. Takovým způsobem může kodér ovlivňovat krok kvantování a tím i kvalitu rekonstruovaného signálu.



Kodér s konstantním datovým tokem

Při použití proměnlivého datového toku vyrovnávací paměť není potřeba a kodér tak může použít kvantovací krok podle požadavků kódovaného vstupního signálu. Kvalita rekonstruovaného signálu se potom nemění.



Kodér s proměnlivým datovým tokem

I když se může zdát proměnlivý datový tok nejlepší volbou, platí to jen v případě, když přenosová rychlost přenosového kanálu je dostatečná i pro maximální datový tok, který může kodér s proměnlivým datovým tokem produkovat. Jestliže je však k dispozici omezená přenosová rychlost přenosového kanálu, pro zachování nepřetržitého toku dat bude vhodnější kodér s konstantním datovým tokem.

5.1 Komprese audio signálů

Pro kompresi obecných audio signálů máme k dispozici velký počet kódovacích standardů a kompresních přístupů. Mnohé z nich se zaměřují na specifické druhy zvuku (např. řeč) anebo vlastnosti (výpočetní náročnost, zpoždění, atd.).



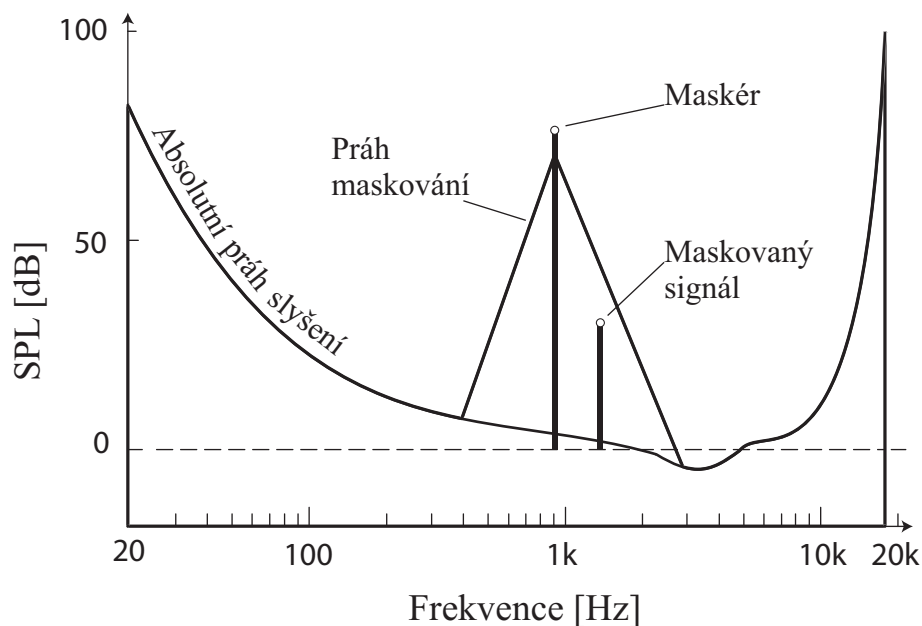
Vzorkovací frekvence popisuje, kolik vzorků bylo ze signálu získáno každou sekundu. Obecně platí, že čím vyšší vzorkovací frekvenci použijeme, tím přesnější, kvalitnější a věrnější nahrávku získáme. Mezi nejpoužívanější vzorkovací frekvence patří: 8 kHz, 16 kHz, 22,5 kHz, 32 kHz, 44,1 kHz anebo 48 kHz pro každý audio kanál.

Sluchové maskování

Sluchové maskování je jev, který se projevuje v důsledku nedokonalosti lidského sluchového aparátu. Uši nejsou schopné vnímat zvuky všech různých intenzit, což je vyjádřeno pomocí tzv. absolutního prahu vnímání zvuku. Kromě toho jeden hlasitý zvuk často překryje blízko se nacházející tišší zvuk. To se může stát v časové a frekvenční oblasti, a proto maskování dělíme na:

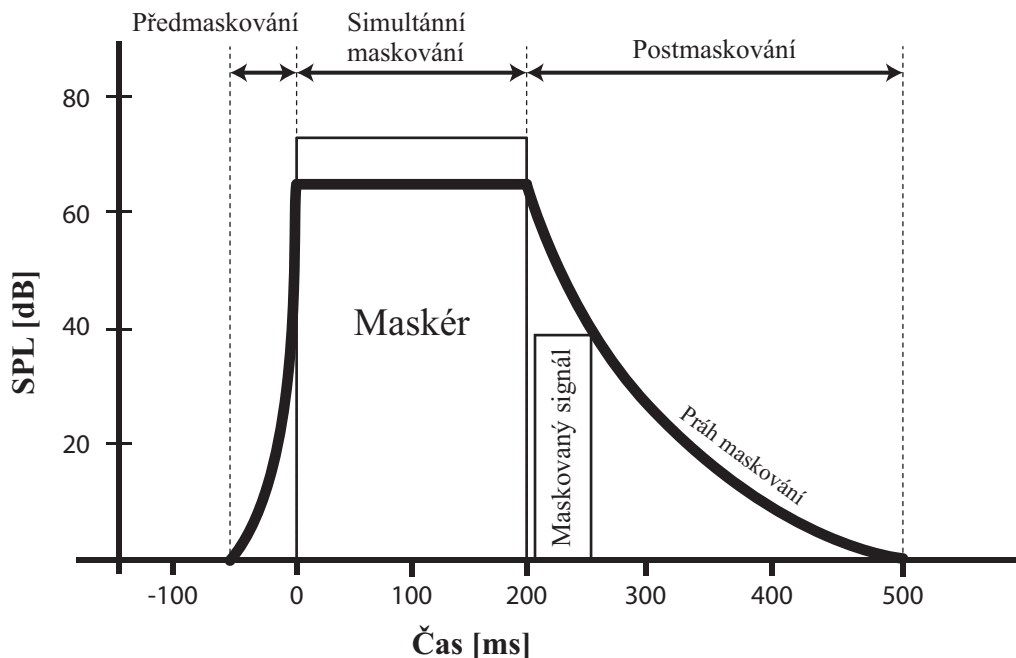
- časové (nesimultánní) maskování
- frekvenční (simultánní) maskování

Hlasitý zvuk se označuje též jako maskér. Když se dva zvuky vyskytnou ve stejnou dobu, může nastat tzv. simultánní maskování. Maskér vytvoří práh maskování, pod kterým nebude možno vnímat žádné jiné zvuky. Když signál v blízkosti maskéra nemá dostatečnou intenzitu a padne pod práh, bude maskován. Následující obrázek ukazuje, jak maskér dokáže překrýt tichý signál ve frekvenční oblasti. Kombinace prahů maskování více maskérů a absolutního prahu vnímání vytváří globální práh maskování, který se může v čase měnit. Proces určování maskovacích prahů se označuje jako psychoakustická analýza.



Maskování ve frekvenční oblasti. Když je intenzita maskéru dostatečně vyšší než intenzita signálu a signály jsou dostatečně blízko od sebe, signál nebude slyšitelný a slyšet bude jen maskér.

Během časového, nesimultánního maskování může maskér překrývat signál, který se vyskytne těsně před (předmaskování) anebo po (postmaskování) zaznění maskéru. Opět však platí, že intenzita maskéru musí být mnohem vyšší než intenzita signálu.



Maskování v časové oblasti

Frekvenční maskování je v současnosti už prozkoumáno dostatečně přesně a má, jak později ukážeme, široké využití v mnohých audio kodecích. Na druhé straně časové maskování zatím nebylo probádané tak důsledně, v důsledku jeho poměrně

krátkého trvání. Postmaskování trvá průměrně nejvíce 300 ms od skončení samotného maskéru a předmaskování trvá dokonce jen méně než 50 ms. Tyto časy jsou příliš krátké na to, aby je bylo možno důsledně analyzovat, protože kodeky pracují obvykle s rámci délky aspoň 20 ms, takže na předmaskování jsou použitelné jen 2 až 3 rámce.

MPEG

V současnosti je většina audio kodeků založena na práci Expertní skupiny pro pohyblivý (dynamický) obraz - *Motion Picture Experts Group* (**MPEG**), která je součástí Mezinárodní standardizační organizace (*International Standards Organization* - **ISO**). Během její existence skupina uvedla několik audio formátů, které se celosvětově používají.

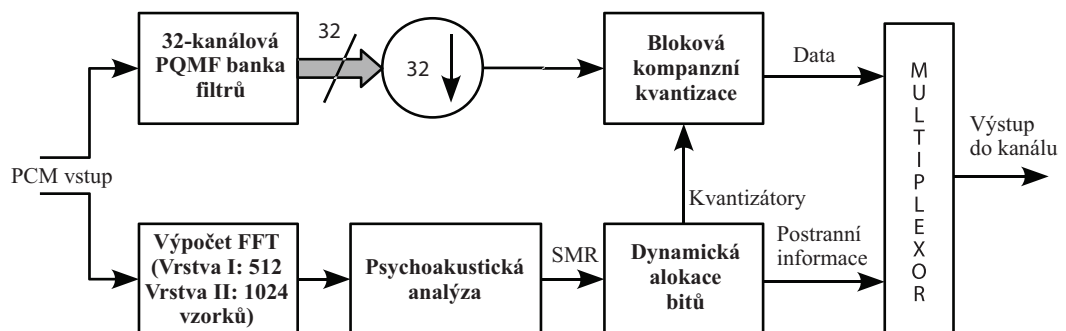
Jak bude dále zřejmé, kodeky z rodiny MPEG jsou založeny na ztrátovém kódování, což znamená, že modifikují původní audio signál a rekonstruovaný signál není nikdy shodný s původním.

MPEG-1



Standard MPEG-1 představuje flexibilní kódovací techniku, která využívá vícero metod, např. subpásmové kódování, analýzu bankou filtrů, transformační kódování, entropické kódování a psychoakustickou analýzu. Pracuje se vzorkovacími frekvencemi 32, 44,1 anebo 48 kHz se 16 bity/vzorek a výstupní datový tok se pohybuje od 32 do 192 kbit/s na jeden kanál. Standard nabízí 4 režimy kódování kanálů: mono, stereo, duální mono a spojené stereo (jen vrstva III).

Architektura standardu obsahuje 3 vrstvy, které se liší výpočetní náročností, zpožděním a kvalitou výstupu. Vrstvy I (mp1) a II (mp2) jsou si podobné a liší se jen v několika detailech. Obě používají rychlou Fourierovu transformaci (*fast Fourier transform* - **FFT**), avšak vrstva I využívá okno s velikostí 512 vzorků, zatímco vrstva II používá 1024vzorkové okno. Maximálně podporované rozlišení v subpásmové kvantizaci je v případě I vrstvy 15 bitů/vzorek a v případě II vrstvy 16 bitů/vzorek. I když se tyto rozdíly zdají být minimální, ukázalo se, že vrstva II poskytuje stejnou či dokonce vyšší kvalitu výstupu při bitovém toku 128 kbit/s než vrstva I s bitovým tokem 192kbit/s na kanál.



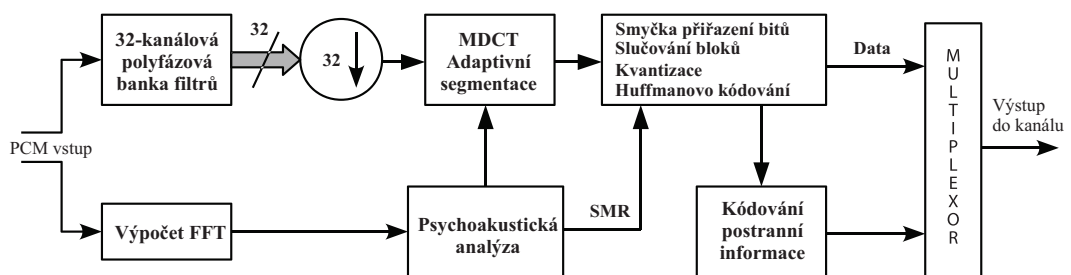
Zevšeobecněné schéma kodéru MPEG-1 vrstva I a II

Proces komprese v obou vrstvách I a II pracuje se vstupním PCM signálem, který rozkládá na 32 subpásem. Během rozkladu se provede FFT, jejíž výstup projde psychoakustickou analýzou a určením jnd. V závislosti na prahu maskování se pro každé subpásmo stanoví nejvhodnější krok kvantování tak, aby byl dodržen požadovaný datový tok a úroveň maskování. Výstup kodéru se na závěr zakóduje Huffmanovým entropickým kódováním.

- + I když MPEG-1 vrstva II poskytuje přijatelné výsledky, převládajícím formátem je MPEG-1 vrstva III, všeobecně známá svojí zkratkou mp3. Vychází z vrstev I a II, přidává však mnohé nové techniky, které vedou k nižšímu datovému toku (okolo 64 kbit/s na kanál) při zachování kvality svých předchůdců.

Algoritmus pracuje s oknem 1152 vzorků, které rozdělí na dvě tzv. granule po 576 vzorcích. Každá z těchto granulí projde hybridní bankou filtrů (soustavou pásmových propustí sloužících k rozdělení vstupu na subpásma: každé subpásmo se potom může zpracovávat samostatně), aby se zvýšilo frekvenční rozlišení. Každé subpásmo se transformuje do frekvenční oblasti použitím modifikované diskretní kosinové transformace (Modified discrete cosine transform -MDCT). Potom se iterativně provede přiřazení bitů a kvantizace: během každé iterace se procesem analýzy pomocí syntézy odhadne úroveň kvantizačního šumu.

- + Modifikovaná diskretní kosinová transformace vychází z diskretní Fourierovy transformace, je však speciálně navržena pro signály, jejichž bloky vzorků se překrývají. Vstupní signál rozkládá (transformuje) na soustavu kosinových funkcí. V porovnání s Fourierovou transformací, jejímž výstupem je soustava komplexních čísel, je výstupem MDCT soustava reálných čísel charakterizujících kosinové funkce. Kromě toho je výstupem DFT stejný počet koeficientů jako je počet vzorků vstupního signálu, zatímco u MDCT je, díky její práci s překryvem na výstupu, poloviční počet koeficientů.



Zevšeobecněné schéma kodéru MPEG-1 vrstva III

K původní vrstvě III vznikly dvě rozšíření, MP3pro a mp3 surround. MP3pro přidává techniku nazývanou replikace spektrálního pásma (*Spectral Band Replication* - **SBR**), která se používá u nižších datových toků na odstranění původních vysokých frekvencí. Ty je možné zrekonstruovat pomocí přidavných informací z komprimovaného signálu.

mp3 surround umožňuje zakódování 5.1 kanálového zvuku (5 plnopásmových a jeden nízkofrekvenční – basový – kanál) do dvou kanálů mp3. Z nich je možné

zrekonstruovat 5.1kanálový zvuk pomocí postranních informací. Když dekodér rozšíření MP3pro anebo mp3 surround nepodporuje, postranní informace ignoruje a soubor přehraje jako standardní soubor mp3.

MPEG-2

Standard MPEG-2 je formálním pokračovatelem MPEG-1. Zahrnuje 2 režimy, jeden zpětně kompatibilní s MPEG-1 (Backward Compatible, MPEG-2 BC) a druhý zpětně nekompatibilní (Non-Backward Compatible, MPEG-2 NBC), který přináší nové metody a techniky kódování.

Jedinými změnami MPEG-2 BC oproti MPEG-1 je podpora pro nižší vzorkovací frekvence (LSF) a vícekanálové kódování podobné rozšíření mp3 surround. Formát MPEG-2 NBC se též označuje jako pokročilé kódování zvuku (Advanced Audio Coding - AAC) a je sestavený jako souprava nástrojů pro efektivní kódování. Čím více nástrojů se použije, tím se dosáhne lepší komprese, přičemž kvalita výstupu zůstane zachována. Cenou je však vyšší výpočetní náročnost a zpoždění. Na rozdíl od MPEG-1 formát MPEG-2 NBC nepoužívá na analýzu signálu hybridní banku filtrů, ale jen MDCT v kombinaci s novými okenními funkcemi. Formát MPEG-2 se stal součástí rodiny standardů MPEG-4.

MPEG-4 AAC

Formát MPEG-4 AAC se pokouší pokořit vládu formátu mp3. Přináší podporu pro vzorkovací frekvence od 8 do 96 kHz, 1 až 48 audio kanálů plus 15 basových a 15 datových kanálů s rozlišením 8, 16, 24 anebo 32 bitů/vzorek. Formát AAC s nízkou složitostí (Low Complexity – LC AAC) představuje původní kodek MPEG-2 AAC a je vhodný na kódování řeči při datovém toku 8-12 kbit/s. Formát AAC s vysokou efektivitou (High Efficiency (HE) AAC) přináší podporu technologie SBR (verze 1) a kanálového režimu parametrické stereo (verze 2), který je založen na profilu stereo standardu MPEG-1 vrstva III.

Ogg Vorbis



Zvukový kodek Vorbis patří mezi nejúspěšnější kodeky s otevřeným zdrojovým kódem. Od roku 2000, kdy byl standardizován, se stal přímým soupeřem formátu mp3 skupiny MPEG. Podporuje vzorkovací frekvence od 8 do 192 kHz, maximálně 255 kanálů a bitový tok jeho výstupu je variabilní.

Proces kódování se odlišuje od standardů MPEG: signál se nejdříve transformuje pomocí MDCT. Ve frekvenční oblasti se určí hrubý odhad spektrální obálky (křivky, která spojuje jednotlivé amplitudové koeficienty frekvenčního spektra) pomocí lomené funkce, který se nazývá floor. Rozdíl mezi floorem a skutečným spektrem se potom zakóduje vícepřechodovou vektorovou kvantizací.

Ogg Vorbis má vyšší nároky na paměť oproti mp3, protože jeho záhlaví obsahuje kódovou tabulku entropického kódování (mp3 má tabulku pevně danou) a nastavení dekodéru. Přesto je formát velmi vhodný na kompresi obecných audio

signálů a poskytuje stejnou anebo vyšší kvalitu zvuku při stejném bitovém toku jako kodek mp3.

Windows Media Audio

Windows Media Audio (WMA) je proprietární kodek vytvořený společností Microsoft jako odpověď na licenční omezení formátu mp3. Existuje více verzí kodeků: WMA 9 je přímým soupeřem mp3 s podporou pro vzorkovací frekvence do 48 kHz se 16 bity/vzorek a výstupním datovým tokem od 64 do 192 kbit/s, podporující CBR i VBR.

Verze WMA 10 Professional rozšiřuje možnosti kodeků v souboji s MPEG-4 AAC přidáním vzorkovací frekvence 96 kHz s 24 bity/vzorek pro 7.1 kanálů. Když zařízení není schopné přehrát 7.1 nahrávku, signál se automaticky degraduje (vzorkovací frekvence, počet bitů na vzorek, snížení počtu kanálů) na úroveň vhodnou pro zařízení.



WMA 10 poskytuje též režim pro kompresi řeči nazývaný WMA 10 Voice, který poskytuje bitový tok od 4 až do 20 kbit/s. Jeho zajímavostí je schopnost dynamicky přepínat mezi řečovou a standardní verzí kodeku, když je zpracováván signál příliš složitý. Kromě toho WMA 10 poskytuje bezztrátový režim, který údajně dokáže zredukovat velikost původního PCM signálu na polovinu až třetinu.



Kodek WMA 10 Professional poskytuje vyšší subjektivní kvalitu při 64 kbit/s v porovnání s MPEG-4 AAC v2 při 70% porovnání.

5.2 Kompresi řeči

Ačkoliv je řeč ve své podstatě audiosignál, má některé specifické vlastnosti, které umožňují použití radikálnějších kompresních technik než u všeobecného zvuku. V první řadě se řečový signál považuje za médium na vyjádření informace. Informace nemusí být po rekonstrukci vyjádřena úplně stejně jako v originále, aby byla srozumitelná. Z toho vyplývá možnost odstranění některých vlastností zvuku. Například standardní telefonní hovor je vzorkován s frekvencí 8 kHz (v porovnání s 44,1 kHz vzorkovací frekvencí při běžném audio), což znamená zachycení šířky pásma jen 4 kHz. Přesto toto pásmo obsahuje většinu energie řeči, a tedy informace.

Za druhé je řečový signál jednoduchý v porovnání například s nahrávkou rockové kapely – obvykle je v něm zachycen jediný řečník bez dalších hudebních nástrojů. Kromě toho na dosažení co možno nejčistší řeči můžeme použít algoritmy na potlačení šumu a ruchů v pozadí.

Metody používané pro kompresi řeči můžeme rozdělit do těchto skupin:

- Kódování tvaru vlny
 - v časové oblasti
 - ve frekvenční oblasti
- Vokodéry
 - lineární predikční kódování
 - formantové kódování

Pro účely hodnocení kvality různých algoritmů na zpracování řeči existuje míra srozumitelnosti. Ta popisuje, jak je řeč pochopitelná a srozumitelná. V úvahu se přitom berou různé vlastnosti, například hlasitost řeči, nelineární zkreslení, úroveň šumu na pozadí, ozvěny i dozvuky a mnohé jiné. Pro porovnávání srozumitelnosti existují dvě hlavní stupnice: index přenosu řeči (Speech Transition Index - STI) a všeobecná stupnice srozumitelnosti (Common Intelligibility Scale - CIS), které mají rozsah od 0 (nejhorší) do 1 (nejlepší), resp. 0% až 100%. Obecně se řeč považuje za srozumitelnou, když algoritmus dosáhne na stupnici skóre alespoň 0,5 (50%).

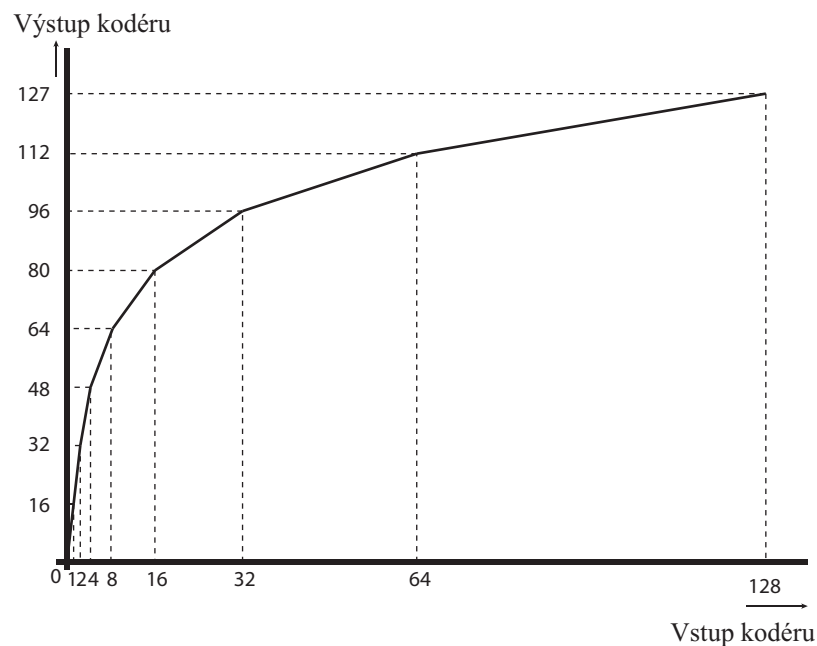
Časová oblast

Kódování tvaru vlny v časové oblasti je reprezentováno technikami PCM. Zatímco lineární PCM používá stejné rozestupy mezi kvantizačními úrovněmi, nelineární PCM využívá nelineární kvantizační stupnici anebo její modifikaci v podobě komprese dynamiky (kompanzi) vstupního signálu na straně vysílače a její expanzi na straně přijímače.

Doporučení G. 711 definuje dvě kompanzní charakteristiky μ -zákon (μ -law - USA a Japonsko) a A-zákon (A-law - Evropa). Na příklad charakteristika A-zákona je dána vztahem:

$$F(x) = \text{sgn}(x) \begin{cases} \frac{A|x|}{1 + \ln(A)}, & |x| < \frac{1}{A} \\ \frac{1 + \ln(A|x|)}{1 + \ln(A)}, & \frac{1}{A} \leq |x| \leq 1 \end{cases}$$

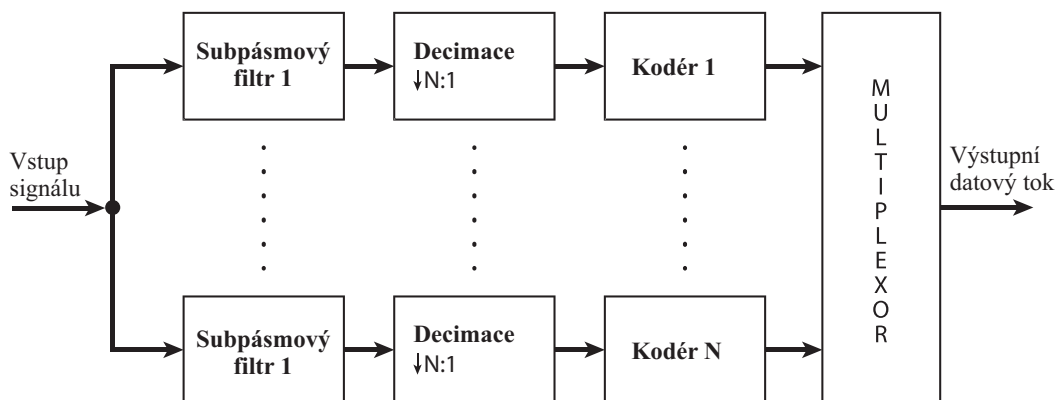
kde $\text{sgn}(x) = \pm 1$ pro kladnou anebo zápornou hodnotu x a A je parametr komprese. Obvykle $A=87,7$.



Příklad kompanzní křivky A-zákona. Vyšší frekvence (na horizontální ose reprezentované vyššími čísly) jsou kódovány menším počtem hodnot než nižší frekvence.

Frekvenční oblast

Ve frekvenční oblasti se využívají metody subpásmového a adaptivního transformačního kódování. Při subpásmovém kódování (Subband Coding - SBC) se řečový signál pomocí sady pásmových propustí (banky filtrů) rozdělí na několik frekvenčních pásem a na snížení počtu vzorků se zdecimuje. Každé pásmo se potom kóduje samostatně, nejčastěji použitím metody ADPCM, která umožňuje flexibilní kvantizaci a přiřazování bitů.

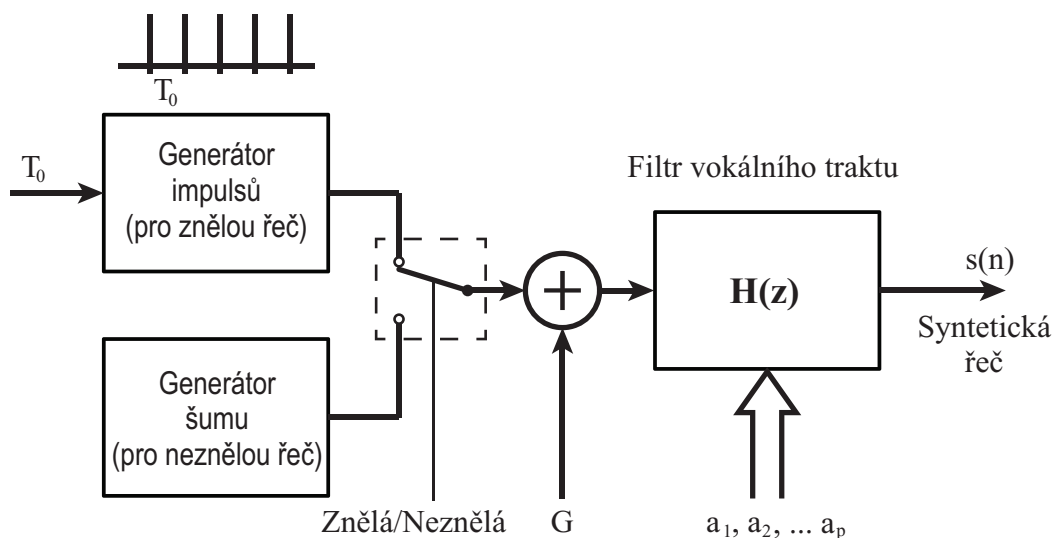


Příklad banky filtrů

Alternativně se místo metody ADPCM mohou použít techniky založené na adaptivním transformačním kódování (*Adaptive Transform Coding - ATC*). Zde se signál transformuje do frekvenční oblasti pomocí FFT anebo jiné transformace a rozdělí se na subpásma. Potom se jednotlivým vzorkům v subpásmech dynamicky přidělí bity podle jejich potřeby.

Lineární predikční kódování

Přirozená lidská řeč se může chápat jako odezva vokálního traktu řečníka na budící signál, v našem případě na vzduch vydechovaný z plic. Výstupní signál se modeluje změnami vlastností vokálního traktu (hlasivek, ústní dutiny, zubů, atd.). Když se na tento proces podíváme z pohledu analýzy signálů, můžeme výstupní signál reprezentovat pomocí budícího signálu a filtru představujícího hlasový trakt s parametry měnícími se v čase. Tyto koeficienty se přepočítávají přibližně každých 10 až 30 ms. I když na popis koeficientů funkce vokálního traktu existuje mnoho metod, nejpoužívanější je metoda založená na lineární predikci, z čehož vychází její název lineární predikční kódování (*Linear Prediction Coding - LPC*).



Všeobecné schéma LPC dekodéru

LPC koeficienty minimalizují kvadratickou odchylku mezi originálními a predikovanými vzorky řeči. Jak je vidět, model LPC generátoru řeči se skládá ze dvou částí vycházejících z výše uvedeného textu:

- Buzení vokálního traktu
- Filtr vokálního traktu

Buzení vokálního traktu reprezentuje generátor impulsů a generátor šumu, mezi kterými je možno přepínat v závislosti na znělosti řeči. Budící signál se dále zesiluje zesilovačem (G) na požadovanou úroveň.



Hlasivky se chvějí, vibrují s určitou základní frekvencí f_0 , ze které umíme odvodit základní hlasivkovou periodu T_0 . Čím vyšší je základní frekvence, tím vyšší je výška řeči.

V závislosti na tom, zda se hlasivky použijí anebo ne, rozdělujeme zvuky řeči na:

- **znělé** – zvuky jsou vyprodukovány pomocí chvění hlasivek, přes které přechází vzduch z plic, např. „a“, „v“, „z“. Všecky samohlásky jsou znělé.
- **neznělé** – zvuky jsou produkovány jen za pomoci toku vzduchu podobnému šumu a hlasivky se nezapojují, například „s“, „c“, „f“.

Filtr vokálního traktu je dán lineárním predikčním filtrem s konečnou odezvou (FIR filtr), jehož přenosovou funkci charakterizuje vztah:

$$H(z) = \frac{G}{1 + \sum_{i=1}^p a_i \cdot z^{-i}} = \frac{S(z)}{E(z)},$$

kde a_i jsou koeficienty filtru a p je řád filtru. Jestliže $S(z)$ představuje výstupní vzorek a $E(z)$ je buzení, následující vzorek $s(n)$ získáme jako lineární kombinaci předcházejících vzorků s buzením $G \cdot e(n)$:

$$s(n) = G \cdot e(n) - \sum_{i=1}^p a_i \cdot z^{-i} = G \cdot e(n) - a_1 \cdot s(n-1) - \dots - a_p \cdot s(n-p)$$

Pro použití LPC generátoru řeči potřebujeme pro každý segment určit tyto parametry:

- znělost segmentu
- základní periodu T_0
- parametry filtru (zesílení G a koeficienty a_i)



Bitový tok řečového signálu zakódovaného pomocí LPC se pohybuje v rozmezí od 1,2 do 2,4 kbit/s a jeho srozumitelnost je okolo 80-85%.

— Rekonstruovaný hovorový signál však zní strojově, což je způsobeno dvěma hlavními důvody:

1. Je těžké segmentovat řeč přesně na znělé a neznělé segmenty, protože v přirozené řeči se znělost prolíná a kombinuje.
2. Základní perioda (která charakterizuje hlas řečníka) se v přirozené řeči mění častěji než je délka rámců a změny nejsou periodické.

Na potlačení nedokonalostí metody LPC existují postupy, které kódují reziduum (rozdíl) mezi originálním signálem a signálem generovaným LPC.

Metoda lineární predikce buzené reziduem (Residually Excited Linear Prediction - RELP) přenáší přímo rozdíl mezi originálním a rekonstruovaným signálem. Na straně přijímače se LPC koeficienty použijí na rekonstrukci signálu a na přesnější rekonstrukci se reziduum připočte.

Nástupcem algoritmu RELP je lineární predikce buzená kódem (Code Excited Linear Prediction - CELP). Algoritmus je založen na principu analýzy syntézou a v uzavřené smyčce percepčně (vjemově) optimalizuje syntetizovaný signál. Potom se prohledá kódová kniha a vybere se nejvhodnější budící funkce. Spolu s LPC koeficienty se přenáší pouze pozice v kódové knize. Budící funkce se může zakódovat pomocí vektorové kvantizace.

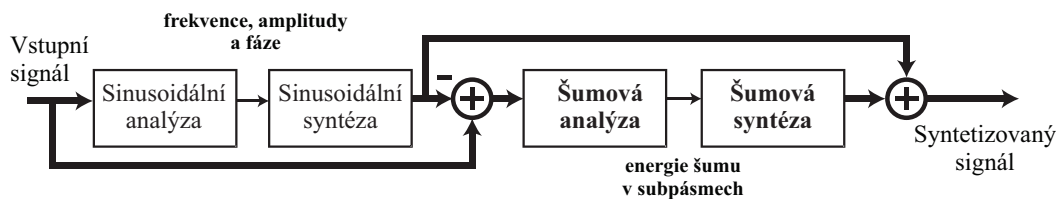
— Metoda CELP dosahuje bitový tok od 4 do 8 kbit/s. Jejími nevýhodami jsou relativní výpočetní náročnost a zpoždění na úrovni 35 ms.



Modifikace CELP-u s malým zpožděním (Low Delay CELP, LD-CELP) snižuje zpoždění na 2 ms při bitovém toku 16 kbit/s a stala se součástí standardu G. 728 organizace ITU-T. Další kodek založený na metodě CELP je Speex, který vytvořila organizace Xiph.Org, autor kodeku Ogg Vorbis, a má otevřený zdrojový kód.

Sinusoidální kódování

Sinusoidální kódování vychází z předpokladu, že každý audio signál je kombinací deterministické a stochastické složky. Deterministickou složku tak můžeme reprezentovat harmonickými funkcemi (sinus, kosinus) a stochastickou složku pomocí šumu anebo jinou parametrizací. Principiální schéma takového kodéru je uvedena níže. Sinusoidy jsou spojeny v čase se měnícími frekvencí, o kterých předpokládáme, že tvoří jeden souvislý tón.



Všeobecné schéma sinusoidálního kodéru

Tento model však nedokáže dostatečně zpracovat rychle se měnící zvuky, proto byla do metody přidána třetí složka, tzv. transienty, které modelují rychlé změny v signálu. To vede k modelu sinusoidy+transienty+šum (STN).

Jiným rozšířením základního SN modelu je model Harmonické + samostatné linie + šum (HILN). V tomto přístupu se sinusoidy rozdělí do dvou skupin, harmonickou a individuální. V harmonické části jsou sinusoidy reprezentovány jako harmonické násobky základních frekvencí a ukládají se pouze násobiče. Potom se kódují individuální sinusoidy a reziduum se zpracovává jako šum.

+ Od sinusoidálního kódování se očekává dobré zpracování jednoduchých signálů, skládajících se obvykle z harmonických zvuků, například řeč. Techniku využíval kodek SVOPC, první kodek společnosti SKYPE, který dosahoval dobrou kvalitu při 20 kbit/s a byl odolný vůči ztrátám paketů.

- Výpočetní náročnost kodeku SVOPC však vedla k vytvoření nového kodeku SILK, založeného na LPC.

Na základě kodeku SILK s přidáním vlastností kodeku **CELT** (*Constrained Energy Lapped Transform*) byl v září 2012 standardizován nový kodek s názvem Opus. Kodek je schopný využít dobrý výkon kodeku SILK při nízkých frekvencích s malým zpožděním kodeku CELT při vyšších frekvencích, mezi kterými dokáže přepínat na požádání. Kodek je vysoce vhodný na kódování řeči i obecného audia, a to i pro online aplikace, jako jsou VoIP a živé přenosy.

5.3 Komprese statických obrázků



Cílem komprese statických obrázků je redukce nepotřebných informací v obraze za účelem snížení šířky potřebného přenosového pásma anebo úložného prostoru. Podobně jako u audiosignálů existují i u obrazu ztrátové i bezztrátové algoritmy, které se používají v závislosti od cílového užití.



Při analýze obrazů automaticky předpokládáme použití 2rozměrných verzí transformací.



Na to, abychom mohli v počítači ukládat obrázky, existují různé barevné prostory. Uvedeme jen ty nejpoužívanější.

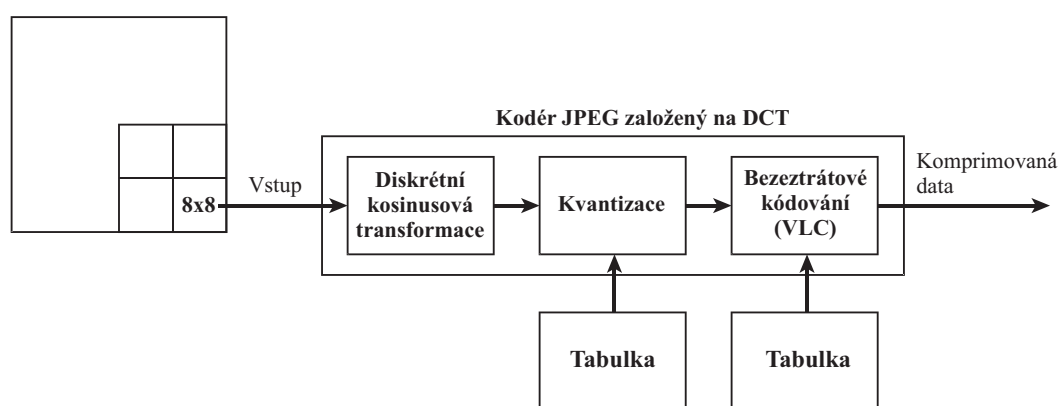
Vybrané barevné prostory

| Zkratka | Význam | Vysvětlení |
|--------------------|--|---|
| RGB | Red, Green, Blue (červená, zelená, modrá) | Každý pixel je dán kombinací 3 barev světla. Kombinace nejvyšších úrovní všech 3 barev dává bílou. Používá se u obrazů vytvářených světlem. |
| RGBA | Red, Green, Blue, Alpha (červená, zelená, modrá, alfa) | Stejný význam jako u RGB. Přidaný alfa kanál popisuje průhlednost. |
| YCbCr anebo YUV | Jas (Y), Modrá barevnost (U), Červená barevnost (V) | Jas se mění od černé do bílé. Modrá a červená barevnost se vypočítají z daného zdroje RGB. I když se YCbCr označuje i jako YUV, existují mezi nimi rozdíly při výpočtu složek barevnosti. |
| CMYK | Cyan, Magenta, Yellow, Black (azurová, purpurová, žlutá, černá) | Každý bod obrazu je dán kombinací čtyřech barev. Kombinace nejvyšších úrovní všech 4 barev dává černou. Používá se v tisku. |

JPEG

Standard JPEG je jedním z nejpoužívanějších ztrátových obrazových formátů na ukládání fotografií. Je pojmenovaný podle spojené expertní skupiny pro obraz (Joint Picture Experts Group), která jej uvedla v roce 1986. Dosahuje kompresní poměr 10:1 s minimálně postřehnutelnou ztrátou kvality.

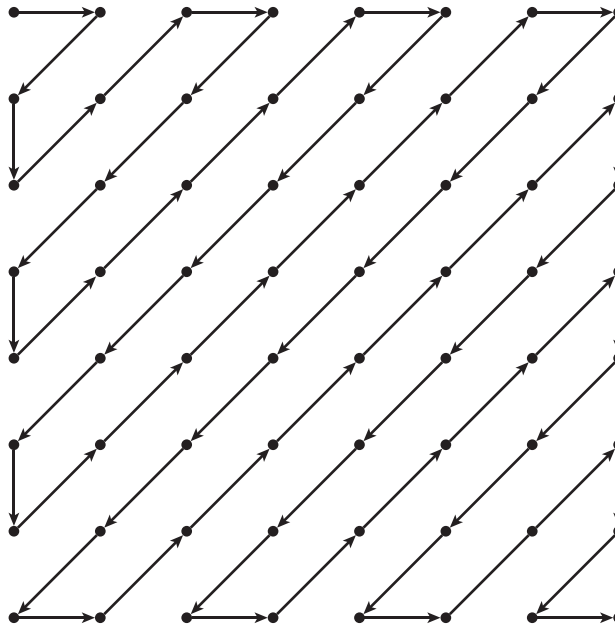
Algoritmus JPEG je založen na 2rozměrné diskrétní kosinusové transformaci (DCT). Vstupní obraz se zkonvertuje do barevného prostoru $Y C_B C_R$, který poskytuje lepší vlastnosti, než řekněme RGB. Obraz se potom rozdělí na nepřekrývající se bloky velikosti 8x8 bodů, které se transformují pomocí DCT. Získané koeficienty se kvantizují a málo významné koeficienty se odstraní (zde dochází k ztrátové kompresi). Koeficienty se potom seřadí do 1rozměrné sekvence a bezztrátově zakódují. Obrazové komponenty (Y , C_B a C_R) se kódují postupně.



Obecné schéma kodéru JPEG. Vstupem je blok 8x8 bodů jasové nebo barevné složky.

Klíčem v kompresi JPEG algoritmem je kvantizace. Kvantizace je nelineární, neboť lidské oko je citlivější na změny v nízkých frekvencích. Pro to, aby byla kvalita, resp. kompresní poměr, měnitelná, byl zaveden faktor kvality q_f , který se mění v rozsahu 1 až 100 a upravuje kvantizační matici.

K seřazení koeficientů slouží tzv. cik-cak čítání, které začíná v levém horním rohu. Jestliže se každý blok 8x8 společně zakóduje, kódování se označuje jako základné (baseline) JPEG. Jiným přístupem je kódování nejprve levých horních rohů všech obrazových bloků a potom následujících koeficientů ze všech bloků. Takovýto postup se označuje jako progresivní JPEG a jeho výhodou je postupná rekonstrukce obrazu už během jeho přebírání. JPEG navíc nabízí tzv. hierarchický režim, kde se obraz kóduje pyramidovým způsobem ve vrstvách. Každý pixel vyšší vrstvy vznikne použitím určité operace na blok 2x2 nacházející se přímo pod ním. Každou vrstvu dokáže dekódér dekódovat samostatně, což umožňuje použít obraz při různých rozlišeních.



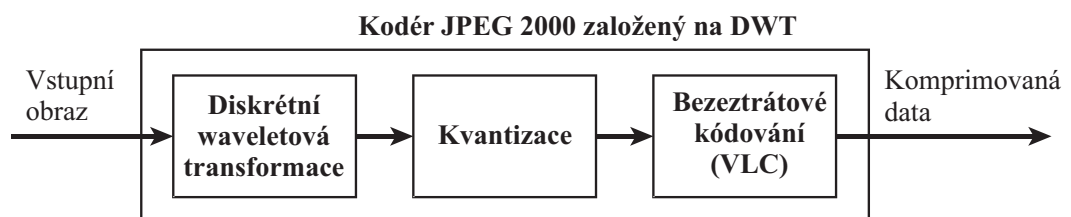
Seřazování koeficientů do 1-rozměrného vektoru pomocí cik-cak čítání



Standard JPEG podporuje též bezztrátové kódování založené na predikčním kódování a bezztrátovém VLC kódování, čímž se vynechává DCT a úpravy spektra. Typický kompresní poměr je přibližně 2:1.

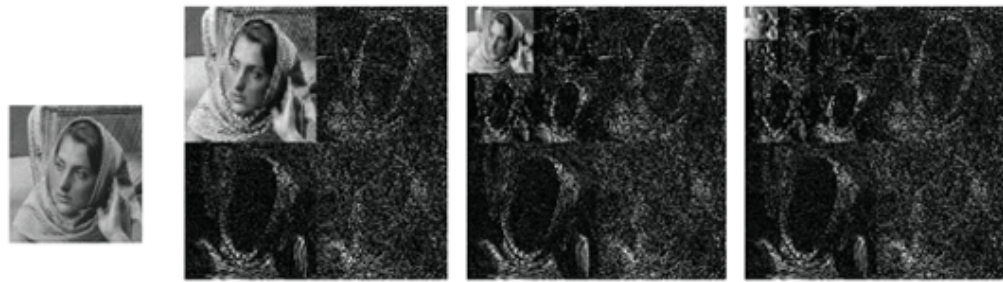
JPEG 2000

Formát JPEG 2000 se snaží nahradit nedokonalý, ale stále výrazně oblíbenější formát JPEG. Rozšiřuje možnosti svého předchůdce, zlepšuje poměr kvalita/komprese, umožňuje škálovatelnou ztrátovou a bezztrátovou kompresi. Mezi další vylepšení patří kódování oblasti zájmu, při kterém se důležité části obrazu kódují přesněji než zbytek.



Obecné schéma kodéru JPEG 2000

Transformační funkce se změnila z DCT na 1D diskrétní waveletovou transformaci (DWT). Původní obraz se transformuje pomocí waveletu, kvantizuje a entropicky zakóduje. Hlavní rozdíl mezi DCT a waveletovou transformací spočívá v tom, že DWT rozděluje bloky obrazu na podbloky, které se rozkládají na podbloky, atd.



Původní obraz 2D DWT, 1. úroveň 2D DWT, 2. úroveň 2D DWT, 3. úroveň

Příklad rozkladu obrazu pomocí 2D DWT. Všimněte si, jak jsou detaily rozdělené do bloků. Každý blok je tvořen DWT transformací v horizontálním, vertikálním a diagonálním směru. Nižší úroveň se rozkládá do bloků vyšší úrovně.



Wavelet je část funkce ve tvaru vlnky, která, na rozdíl od řekněme funkce sinus jdoucí od nekonečna do nekonečna, má stanovený začátek, amplitudu a konec. Může mít různé tvary, které si vybíráme v závislosti na charakteru analyzovaného signálu.

Waveletová transformace v podstatě hledá podobnost waveletu s částí analyzovaného signálu. Jestliže wavelet má svůj začátek a konec, můžeme ho „roztahovat v libovolné míře“. Když použijeme přesně stanovené kroky míry, získáme víceúrovňové waveletové spektrum.

DWT transformace vytváří 2 soupravy vzorků, vzniklé filtrací dolní-, resp. horní propustí. Na úspěšnou rekonstrukci signálu však potřebujeme znát jen vzorky z filtru typu horní propust každé úrovně rozlišení. Ty představují detaily, které musíme přidat k odhadu signálu v nižší úrovni, abychom mohli zrekonstruovat vyšší úroveň.

GIF

Formát GIF (Graphics Interchange Format) je na Internetu stále populárním formátem obrázkových souborů. Byl uveden v roce 1987 a představuje bitmapový formát s podporou 8bitové barevné palety, přehlednost a má dobrý kompresní poměr. V důsledku omezené palety barev (255 barev) má omezené použití pro obrazy s vysokou věrností, jako jsou fotografie. Je však vhodný na obrázky s omezeným počtem barev, jako jsou např. loga, s ostrými hranami a minimálními barevnými přechody. Druhá verze formátu (1989) přináší podporu přehlednosti.

GIF pro kompresi obrazových údajů používá algoritmus Lempel-Ziv-Welch (LZW), který barvám v barevné paletě přiřazuje bytové sekvence ze slovníku.



Přesto, že dnes existují pokročilejší algoritmy, například PNG, formát GIF si stále udržuje svoji popularitu díky svojí podpoře pro animace umístěním více obrázků nad sebou. Tato vlastnost byla využita na vytváření plnobarevných (24bitových) obrázků a animací umístěním třech 8bitových rámců na sebe, kde každý rámeček obsahuje část 24bitové barevné palety.

PNG

Portable Network Graphics (přenosná síťová grafika) je bitmapový obrazový formát určený k náhradě formátu GIF, který je zatížen licenčními a technickými omezeními. Byl představen v roce 1996 a mezinárodním standardem ISO/IEC se stal v roce 2004. PNG podporuje barevné prostory RGB a RGBA s 8 bity/barva (24bitové RGB a 32bitové RGBA).

Formát PNG je velmi flexibilní díky svojí struktuře podobné kontejneru. Obraz se vytvoří „po kouscích“, které umožňují rozložit obrazovou informaci a podporuje se vrstvení a streamování dat.

Bezeztrátová komprese pracuje ve dvou krocích:

- Předkomprese (filtrování)
- Komprese

Během předkomprese se původní množství obrazových dat zredukuje metodou podobnou DPCM: hodnota pixelu se ukládá jako rozdíl mezi ním a pixelem vlevo, nad ním, vlevo nad ním anebo jejich kombinace. Na každý řádek pixelů je možno použít jiný filtr. Potom se hodnoty komprimují algoritmem DEFLATE, který nahrazuje stejné řetězce referencemi a Huffmanovo kódování používá nikoliv na celý obraz, ale na jednotlivé bloky dat.



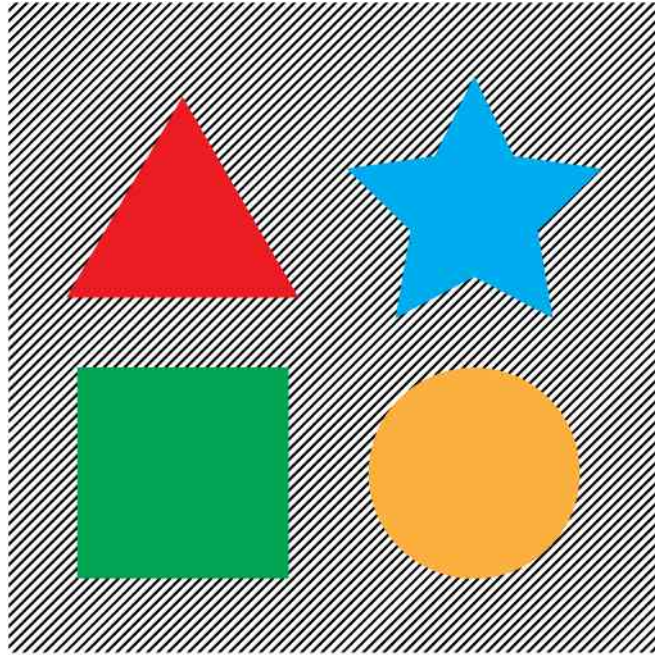
První návrh na název formátu byl PING – „PING is not GIF“ (PING není GIF).

–

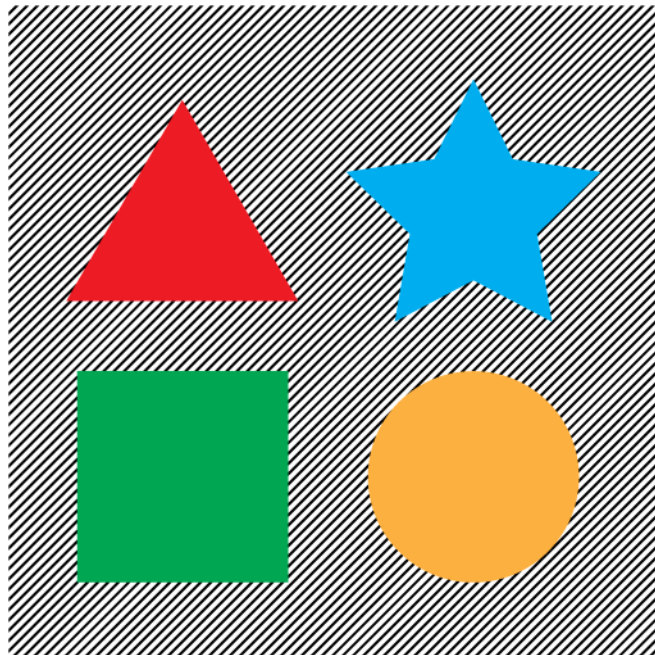
V porovnání s formátem JPEG vytváří formát PNG z fotografií s jemnými přechody větší soubory.

+

Přesto má formát JPEG velké problémy při zpracování ostrých přechodů a hran, jako je text, čáry anebo grafika, a na velkých plochách stejné barvy, kde vytváří artefakty. PNG je schopen lepší komprese, po které nezůstávají žádné artefakty, takže je ideální pro použití na webu.



Porovnání obrázků zakódovaných ve formátu JPEG (vyšší) a PNG (nižší). U JPEG verzi jsou na hranách viditelné nedokonalosti na hranách. Velikost souboru JPEG je 61 kbit, soubor PNG má pouze 11 kbit.



WebP

Nejmladší z obrazových formátů WebP, uvedly laboratoře společnosti Google v roce 2010. Presentují ho jako nový otevřený standard, který chce soutěžit se stále populárním formátem JPEG. WebP v sobě spojuje nejlepší vlastnosti JPEG (dobré zpracování plněbarevné grafiky), JPEG 2000 (ztrátová i bezztrátová komprese), PNG (přehlednost ve ztrátovém i bezztrátovém režimu) a GIF (podpora animací).

Ztrátový algoritmus je založen na postupech ve video formátu VP8. Komprese je založena na predikci obrazových bloků ze třech bloků nad a jednoho nalevo od analyzovaného bloku s použitím jednoho ze čtyř režimů: horizontální, vertikální, DC (jedna barva) a TrueMotion. Špatně predikované a nepredikované bloky se potom komprimují v blocích 4x4 pixelů pomocí CDT anebo Walsh-Hadamartové transformace. Výstup se entropicky zakóduje.

Kromě běžných technik jako jsou slovníkové a Huffmanovo kódování používá bezztrátový algoritmus pokročilé metody jako různé entropické kódy pro různé barevné kanály anebo dočasná paměť nedávno použitých barev.



V porovnání s jinými obrazovými formáty JPEG a PNG, formát WebP překonává svoje konkurenty ve svých zaměřeních nejméně o 20%. WebP je v současnosti podporován na Linux a Windows pomocí pluginů a podporují ho i vyhledavače Firefox, Chrome a Opera.

5.4 Komprese videa

Video, anebo pohyblivý (dynamický) obraz, je v dnešní době nejpoužívanější složkou multimediálního obsahu. Množství video obsahu rapidně stoupá a v porovnání s jinými typy médií zabírá výrazně nejvíce prostoru. S rostoucími možnostmi osobních zařízení ve vytváření a přehrávání videa se množství přenášených dat zvětšuje každý den. Proto je potřebné snížit velikost videa kvůli snížení nákladů na přenos a ukládání dat.

Videosekvence se skládá ze samostatných snímků, anebo obrazů, které je v podstatě možné kódovat stejně jako statické obrazy. Avšak při sekvenci rychle se měnících obrazů se předpokládá výskyt stejného objektu s měnící se pozicí v čase, což znamená, že dva následující snímky jsou si podobné, tj. jen s minimálními rozdíly. Proto když zakódujeme jen tyto rozdíly, můžeme dosáhnout ještě účinnější komprese.

Metody kódování videa můžeme rozdělit na dvě hlavní kategorie:

- Kódování v časové oblasti
- Kódování na bázi modelování

Metody kódování v časové oblasti využívají transformační kódování s mezisnímkovou predikcí a jsou využívány v mnohých formátech. **Metody kódování na bázi modelování** se používají v aplikacích s omezeným přenosovým pásmem, jako je videotelefonie, s datovým tokem do 64 kbit/s. Metody využívají typický vzhled scén při video telefonování s neměním se obsahem, minimálním pohybem a sníženou frekvencí snímkování.

Mezisnímková predikce a kompenzace pohybu

Pro snížení časové nadbytečnosti mezi dvěma snímky se používá mezisnímková predikce s kompenzací pohybu, která pracuje ve dvou krocích:

- Odhad pohybu
- Kompenzace pohybu

Během odhadu pohybu se vytváří vektor pohybu, který popisuje relativní pohyb obrazových bloků z předešlého do aktuálního snímku. Když je pozice předvídatelná, potom stačí přenést jen vektor pohybu. Na odhad pohybu se používají dva algoritmy:

- Iterační algoritmy (pel recursive algorithms)
- Algoritmy s blokovým prohledáváním (block matching algorithms)

Iterační algoritmy se iterativním způsobem snaží minimalizovat chybu predikce. Protože velmi závisí na místních statistických vzdálenostech, nemohou se použít na odhad větších vzdáleností a jsou tak vhodné pro video s malým pohybem, např. videotelefonii. Algoritmy s blokovým prohledáváním předpokládají, že všechny

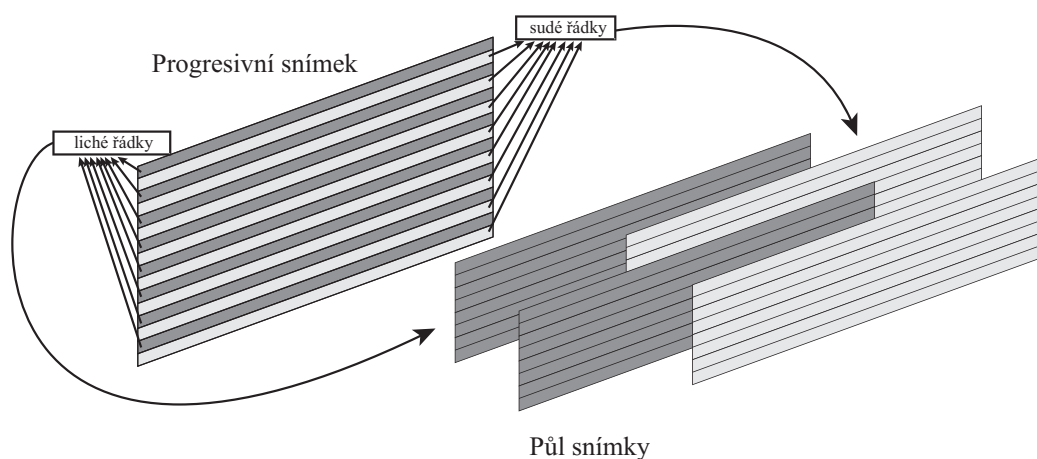
části obrazového bloku se pohybují stejným směrem. Aktuální snímek se rozdělí na bloky, pro které se v předešlém snímku najdou nejpodobnější bloky.

Algoritmy na kompenzaci pohybu využívají vektory pohybu na přesouvání každého bloku z předešlého snímku do nové pozice v aktuálním snímku. Tím vznikne predikční snímek, který se zakóduje a přenáší.

Techniky komprese videa

Překládané video

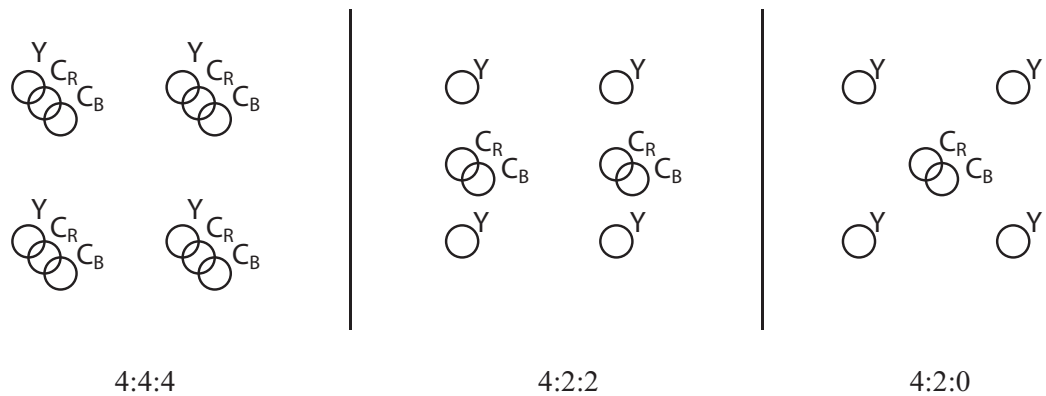
Standardní (progresivní anebo nepřekládané) video je složeno z 25 (Evropa) anebo 29,97 (USA) snímků za sekundu. Překládané video se skládá z dvojnásobného počtu půlsnímků za sekundu (50 v Evropě anebo 59,94 v USA). Každý půlsnímek obsahuje buď sudé, anebo liché řádky celých video snímků, které se dekodují a přehrávají v překládaném pořadí.



Rozdíl mezi progresivním a překládaným videem. Celý snímek se rozdělí na sudé a liché řádky, čímž se vytvoří půlsnímky, které se střídavě uloží do sekvence. Video má dvojnásobnou snímkovou rychlost, avšak pro půlsnímky.

Barevné podvzorkování

Další technikou kódování je barevné podvzorkování. Každý pixel videosnímku je kódován v barevném prostoru YUV skládajícím se ze třech subpixelů: jeden jasový (luminanční) a dva barevnostní (chrominanční). Čtverec čtyř takových pixelů se označuje 4:4:4. Protože lidské oko je citlivější na jasovou složku než na změny barvy, umožňuje to snížit počet chrominančních pixelů na polovinu, či dokonce na čtvrtinu. V prvním případě jsou čtyři luminanční pixely pokryty dvěma a dvěma chrominančními pixely (4:2:2) a v druhém případě jen dvěma, resp. jedním a jedním chrominančním pixelem (4:2:0 anebo 4:1:1).



Příklady barevného podvzorkování. Levý obrázek ukazuje, že každý lumenanční subpixel má svoji dvojici chrominančních subpixelů. Uprostřed dva lumenanční subpixely sdílejí pár chrominančních subpixelů. Napravo jsou pro čtyři lumenanční subpixely dostupné jen dva chrominanční.

Podobně jako u statických obrazů se kompresní techniky u videa opírají o hybridní metody kombinující kódování v časové oblasti s transformačním kódováním (jako například diskretní kosinová (DCT) anebo diskretní waveletová transformace (DWT)). Predikční snímek vytvořený v procesu kompenzace pohybu se odčítá od aktuálního snímku a vytvoří se chybový snímek. Ten se po blocích zakóduje pomocí DCT, transformační koeficienty se kvantizují, seřadí cik-cak čítáním a zakódují pomocí kódu s proměnlivou délkou (VLC). Potom se posílá jen VLC sekvence.

Na straně přijímače se rekonstrukce provádí inverzním postupem, tedy inverzním VLC, inverzní kvantizací a IDCT.

Popsaný postup se s malými odlišnostmi používá v kódovacích standardech MPEG a H26x.

MPEG

Expertní skupinu pro pohyblivý obraz (Motion Picture Experts Group, MPEG) zřídili organizace ISO a IEC, pro vytvoření standardů pro kompresi videa a zvuku. Jako výsledek byly navrženy tyto standardy zaměřující se na různé aplikace:

- MPEG-1
- MPEG-2
- MPEG-4

Standard MPEG-1 se zaměřuje na interaktivní systémy na bázi médií CD-ROM. MPEG-2 rozšiřuje schopnosti MPEG-1 pro digitální televizi a TV s vysokým rozlišením (HDTV). MPEG-4 se zaměřuje na multimediální aplikace s velmi nízkými bitovými toky.

MPEG-1

Standard byl vyvinut pro kódování video signálů s dostatečnou kvalitou při přenosové rychlosti 1,4 Mbit/s. Podporuje rychlé přetáčení, posun vpřed

a pozastavení obrazu. I když typická velikost videosnímku je 352x288 pixelů (CIF), kodek podporuje velikost snímku do 720x576 pixelů při 30 snímcích za sekundu a přenosové rychlosti 1,86 Mbit/s.

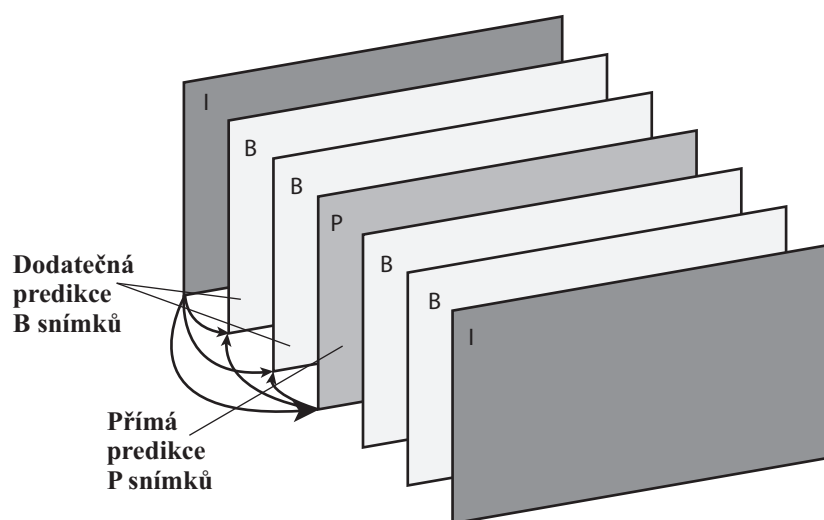
Mezislánkové kódování v MPEG-1 je založeno na mezislánkové predikci a DCT kódování. Jsou definovány 3 typy makrobloků (sad čtyřech lúminančních a dvou chrominančních bloků), ze kterých vyplývají tři druhy snímků:

- **I snímky** – Intra snímky – kódování snímků
- **P snímky** – Predikované snímky – mezislánkové kódování s přímou predikcí
- **B snímky** – Obousměrné snímky – dvojsměrná predikce/interpolace

Snímky I se v blocích 8x8 kódují DCT, kvantizují a po cik-cak seřazení entropicky zakódují. Neprovádí se žádný odhad pohybu, takže snímek se chová jako fotografie, je nezávislý na ostatních snímcích a slouží jako bod zastavení při rychlém přetáčení a posunu vpřed.

Snímky P se kódují s použitím mezislánkové predikce a porovnáním s předešlým snímkem. Tím se vytvoří chybový snímek (reprezentovaný vektorem pohybu), který se rozdělí na makrobloky velikosti 16x16 pixelů. Ty se zakódují pomocí DCT, kvantizují a entropicky zakódují podobně jako u I snímcích. Přesto tyto snímky neobsahují celou obrazovou informaci, protože závisí na předešlých snímcích, slouží jen jako referenční snímky při predikci, ne pro rychlé přehrávání.

Snímky B jsou získány jen pomocí dopředné a zpětné predikce z I anebo P snímků. B snímky obvykle slouží jen jako výplň anebo přidání detailů v rychlých scénách mezi I a P snímky, protože obsahují stejné informace jako P snímky. Z B snímků se neprovádí predikce.



Řazení I, P a B snímků a závislosti mezi nimi

Rámce je možno kombinovat flexibilně podle požadavků aplikací. Sekvence IIIIIIIII nabízí výborný přístup ke snímkům a rychlému přehrávání a posunu vpřed, avšak nízká komprese u I snímků zvyšuje nároky na přenosové prostředky.

Obvykle se proto používá sekvence IBBPBBPBBPBB(I) (nazývaná Group of pictures - GOP), při které se I snímek používá jen přibližně dvakrát za sekundu.

MPEG-2

MPEG-2 je rozšířením předešlého standardu (se zpětnou kompatibilitou) s možností překládaného videa, vylepšeným maximálním rozlišením obrazu, TV kvalitou videa při přenosových rychlostech 4 až 8 Mbit/s a HDTV kvalitou videa při 20 Mbit/s.

Vlastnosti dekodérů jsou dány prostřednictvím profilů a úrovní. Každá úroveň definuje sadu parametrů určujících cílovou aplikaci videa, zatímco profily určují složitost použitých algoritmů. Následující tabulky podrobně popisují úrovně a profily.

Přehled úrovní v MPEG-2

| Úroveň | HIGH (vysoká) | HIGH 1440 (vysoká 1440) | MAIN (hlavní) | LOW (nízká) |
|-----------|--|--|--------------------------------------|-------------------------------------|
| Parametry | 1920x1152 pix. 60 sn/s 80 Mbit/s | 1440x1152 pix. 60 sn/s 60 Mbit/s | 720x576 pix. 30 sn/s 15 Mbit/s | 352x288 pix. 30 sn/s 4 Mbit/s |

Přehled profilů v MPEG-2

| Profil | Algoritmy |
|---|--|
| High (vysoký) | Všechny funkce profilu Spatial Scalable, navíc režimy kódování s 3-vrstvovým prostorovým a SNR škálováním Barevný model YUV 4:2:2 pro náročné úlohy |
| Spatial Scalable (prostorové škálování) | Všechny funkce profilu SNR Scalable, navíc režim kódování s 2-vrstvovým prostorovým škálováním Barevný model YUV 4:2:0 |
| SNR Scalable (SNR škálování) | Všechny funkce profilu Main, navíc režim kódování s 2-vrstvovým SNR (signal-to-noise ratio) škálováním Barevný model YUV 4:2:0 |
| Main (hlavní) | Bez škálování, kódování překládaného videa Náhodný přístup ke snímkům, režim predikce s B snímky Barevný model YUV 4:2:0 |
| Simple (jednoduchý) | Podporuje všechny funkce profilu Main kromě režimu predikce s B snímky Barevný model YUV 4:2:0 |

Škálování videa

Škálování umožňuje dekodérům přehrávání videa s nízkou přenosovou rychlostí, když nejsou schopny přehrát video s vysokou přenosovou rychlostí. Dekodér přijme video s nízkou kvalitou a doplňující informace, které mu umožní zvýšit kvalitu videa. Pomocí SNR škálování se DCT koeficienty škálují nahrubo, čímž vznikne video s nízkou přenosovou rychlostí. Potom se rozdíl mezi hrubou kvantizací a skutečnými hodnotami znovu kvantizuje pomocí jemnější kvantizace a tato informace se přenáší samostatně. To umožňuje zvýšení kvality videa na požádání. Při prostorovém škálování se video kóduje nejprve při nižším rozlišení a vyššího rozlišení se dosáhne doplňkovými daty. Když dekodovací zařízení nepodporuje vyšší rozlišení, vynechá doplňková data a dekóduje jen video s nízkým rozlišením. Časové škálování funguje podobně. Vytvoří se video s nízkým počtem snímků a pomocí doplňkových dat je možná rekonstrukce videa s vyšším počtem snímků. Prostorové a časové škálování se dá kombinovat, což přináší variabilitu při kódování videa, například na podporu HDTV a TV v standardním rozlišení.



Organizace ISO a ITU-T vytvářely standard MPEG-2 společně, přičemž ITU-T ho nazvalo H.262.

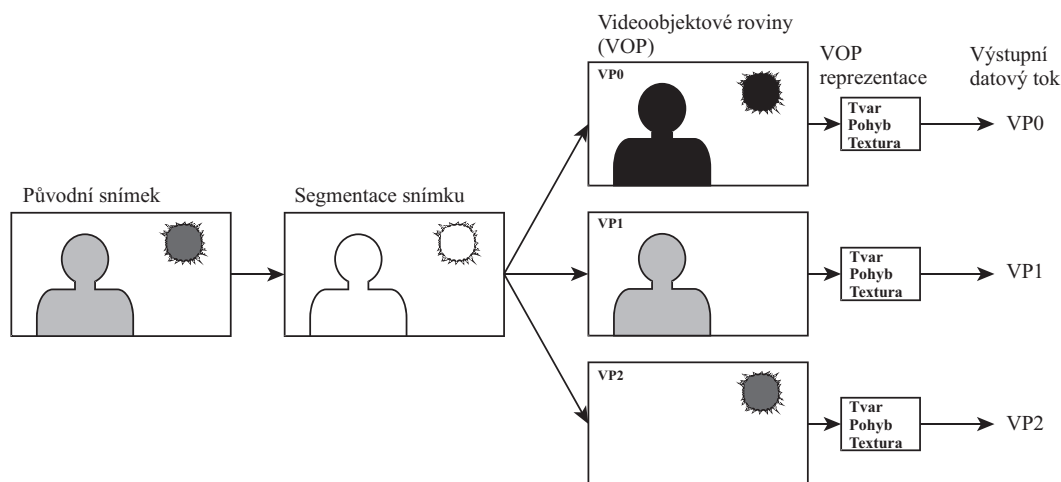
MPEG-4

Standard MPEG-4 byl vyvinut na podporu velmi nízkých přenosových rychlostí do 64 kbit/s. Jeho cílem je podpora videa na Internetu, v mobilních zařízeních a sítích a podpora interaktivity s objekty na scéně. To vyžadovalo vylepšení kompresních metod, které využívají kódování video objektů v přirozeném (standardním) a syntetickém (kódování modelů objektů) videu.



Standard MPEG-4 existuje ve dvou verzích. První z nich se označuje Část 2 a používá jí mnoho kodeků včetně DivX, XviD, Nero Digital a dalších. Druhá verze se označuje Část 10 anebo též MPEG-4 / H.264 Pokročilé kódování videa, a používá se v kodecích x264, Quicktime anebo pro média HD video, jako například Blu-ray Disc.

Kódování přirozeného videa se děje prostřednictvím detekce a kódování videoobjektových rovin (video object planes - VOP). Každá VOP obsahuje informace o tvaru a textuře objektu na scéně. Sekvence VOP-ů představujících ten samý objekt se nazývá video objekt (VO). Každý video objekt je možné kódovat s jiným datovým tokem, což umožňuje flexibilní přiřazování bitového toku a další úkony s objekty (škálování, rotace změn jasu a barvy).



Ukázka použití video objektů

Video objekt je určen svým tvarem daným binární maskou anebo maskou v odstínech šedé. Kódování pohybu je založeno na podobných principech, jaké jsou použity v předešlých standardech MPEG, ale aplikují se na video objektové roviny, čímž se vytvářejí IVOP, PVOP a BVOP snímky. Prostorová nadbytečnost se odstraní pomocí DCT a časová nadbytečnost použitím kompenzace pohybu. Textury video objektů se kódují prostřednictvím Shape Adaptive (SA) DCT, modifikace DCT. Alternativně se může použít kódování pomocí SA DWT (diskrétní waveletové transformace).

Syntetické video umožňuje vytváření umělých objektů a jejich vkládání mezi existující video objekty na scéně. Hlavním důvodem je umožnění animace tváře pro multimediální aplikace.

Standardy H.261 a H.263

Standard H. 261 byl publikován v roce 1990 s použitím ve videotelefónii a videokonferencích s nízkými přenosovými rychlostmi od 64 do 1920 kbit/s a malým zpožděním. Sjednocuje různé televizní standardy s různým počtem řádků a frekvencí pulsů (PAL a SECAM s 625 řádky při 50 Hz, NTSC s 525 řádky při 60 Hz). Kodek používá rozlišení CIF (352x288) a QCIF (176x144), z kterých jedno rozlišení se používá pro videokonference s více účastníky a druhé pro videotelefónii, ve které se obvykle přenáší jen hlava a ramena jediné osoby.

Rozlišení CIF a QCIF se skládají ze skupin bloků (GOB): CIF z GOB 1-12 a QCIF z GOB 1, 3 a 5. Každá GOB se skládá z 33 makrobloků složených z 6 bloků: 4 jasových (luminančních) a dvou barevných (chrominančních – C_R , C_B), každý složený z 8x8 pixelů.

Kodek H. 261 využívá pouze I snímky (označované jako klíčové snímky) a P snímky, které se získají pomocí predikce pohybu z I snímků anebo předešlých P snímků. Standard nevyužívá B snímky.

Kódovací algoritmus využívá hybridní blokové kódování s mezisnímkovou predikcí a kompenzací pohybu a transformační kódování na bázi DCT, které je podobné kódování MPEG-1. Po odstranění prostorové a časové nadbytečnosti se

každý blok transformuje DCT, kvantizuje, pomocí cik-cak čítání seřadí do vektoru a zakóduje Huffmanovým (bezeztrátovým) kódováním. Navíc se používá iterační filtr, který vyhlazuje rozdíly mezi bloky predikovaného obrazu, čímž se vylepšuje mezisnímková predikce.

Iterační filtr pracuje se sekvencí snímků a odstraňuje blokové artefakty vzniklé DCT transformací každého bloku. Jeho úlohou je vyhladit ostré hrany mezi bloky snímků. Vyhlazování probíhá opakovaně ve smyčce, až se dosáhne žádaná hranice. I když zpracování smyčky může zabrat více času než samotné dekódování, ale po procesu odhadu pohybu tak může být potřebné zakódovat menší vektor pohybu.

Standard H. 263 přináší v porovnání s H.261 efektivnější kódování. Díky použití některých technik z MPEG-1 dosáhl kodek snížení bitové rychlosti o 50%, přičemž zachovává stejnou subjektivní kvalitu. V porovnání s H. 261 standard H. 263 přináší širší podporu video formátům (SQCIF, 4CIF, 16CIF), vylepšený odhad vektoru pohybu, modifikované VLC kódování a zavádí PB snímky.

Odhad pohybu v H. 263 pracuje s půl pixelovou (half-pel) predikcí. Zatímco vektory pohybu v H. 261 pracují s celými čísly, vektory v H. 263 jsou reprezentovány s přesností 0,5. Kromě toho se vektor pohybu makrobloku odhaduje porovnáním s vektory pohybu okolních makrobloků (výpočet mediánu) a přenáší se jen rozdíl mezi odhadovaným a skutečným vektorem pohybu (nazývá se mediánová predikce).

Režim PB snímků pracuje podobným způsobem jako v kodeku MPEG-1. P snímek se získá z I snímku anebo P snímku. B snímek se získá obousměrnou predikcí z okolních snímků. Rozdíl mezi MPEG-1 a H. 263 je v tom, že B snímek v H.263 je uložen přímo v P snímku, čímž se vytváří PB snímek. Je to výhodné zvláště pro video s nízkým datovým tokem.



Rozšíření standardu H. 263, standard H.263+, přináší odolnost vůči chybám přenosu, dynamické rozlišení scény a škálování snímků.

MPEG-4 AVC/H.264

Další (a novější) verze standardu MPEG-4, označovaná jako pokročilé kódování videa (Advanced Video Coding, AVC), je dnes nejpoužívanějším standardem. Spravován je společně organizacemi ISO a ITU-T a je vhodný zejména pro kompresi videa ve vysokém rozlišení.

Standard přináší mnoho vylepšení předcházejících standardů obou organizací, jako je vyšší rozlišení barevné informace, škálovatelné kódování videa a více pohledové kódování videa, které umožňuje kódování několika úhlů videa, a tedy podporuje stereoskopické (3D) video.

Variabilní velikost bloků umožňuje precizní segmentaci pohyblivých regionů s velikostí od 16x16 do 4x4 pixelů. Z jednoho makrobloku může být odvozených více vektorů pohybu směřujících k různým referenčním obrazům. Algoritmus kompenzace pohybu pracuje se čtvrtpixelovou přesností (v porovnání s půlpixelovou přesností H.263), což umožňuje vyšší přesnost vektorů pohybu.

DCT transformace byla vylepšena a upravena tak, aby poskytovala jednoznačné dekódování. Navíc se na hladké oblasti může použít sekundární Hadamardova transformace, která ještě více zlepšuje kompresní poměr.

Kromě toho bylo zavedeno bezeztrátové kódování makrobloků, které umožňuje dokonalou reprezentaci vybraných oblastí obrazu. Pracuje ve dvou režimech: PCM makroblok anebo vylepšený bezeztrátový makroblok (efektivnější). Entropické kódování používá nové algoritmy na kódování prvků syntaxe a kvantizovaných hodnot transformačních koeficientů: Kontextově adaptivní binární aritmetické kódování a Kontextově adaptivní kódování s proměnlivou délkou, které kódují efektivněji než v předešlých standardech.

Množství dalších vylepšení zachovává stejnou subjektivní kvalitu videa jako předešlé standardy, ale při polovičním, či dokonce menším bitovém toku, což je zřejmé zejména při vysokém datovém toku a videu s vysokým rozlišením.

Podobně jako u MPEG-2, standard MPEG-4 AVC/H.264 podporuje kódovací profily pro použití na různé cílové aplikace a úrovně definující požadovaný výkon dekodéru.

WebM

WebM je audio a video kodek s otevřeným zdrojovým kódem od společnosti Google, určený na použití s HTML5 videem. Je to multimediální kontejner, založený na kontejneru Matroska, který obsahuje zvuk kódovaný standardem Ogg Vorbis a video kódované kodekem VP8.

Kodek VP8 vyvinula společnost On2 Technologies a po tom, jak jí v roce 2010 odkoupila společnost Google, byl uvolněn pod licenci s otevřeným zdrojovým kódem. I když VP8 využívá mnohé techniky zavedené standardy MPEG a H.26x, přináší další vylepšení, které umožňují zachování vysoké subjektivní kvality při snížení výpočetní náročnosti. Některé z nich nyní uvedeme.

Kodek využívá tzv. konstruovaný referenční snímek, který slouží jako referenční snímek pro kompenzaci pohybu několika snímků. Vzhled konstruovaného referenčního snímku není určen, takže je ponechán na volbě návrhářů. Proces iteračního filtrování, který odstraňuje blokové artefakty po snížení prostorové nadbytečnosti (DCT transformace) může u každého bloku použít různý počet bloků v sekvenci. Entropické kódování využívá zejména binární aritmetické kódování, které se přizpůsobuje každému snímku zvlášť.



Kromě už zmíněných kódovacích standardů existuje množství dalších video formátů, například Windows Media Video (WMV) od společnosti Microsoft anebo Ogg Theora, založený na starším kodeku VP3 od On2 Technologies, a mnohé další standardy, které nepokrývají organizace ISO ani ITU-T.

6 Zpracování multimédií

6.1 Syntéza řeči



Pod syntézou řeči rozumíme vytváření řeči, která je co nejpodobnější lidské, za pomoci stroje, který je označován též jako syntetizátor.

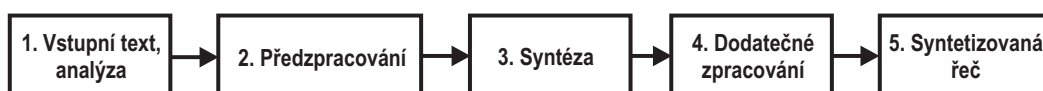
Existuje více typů syntetizátorů, avšak cílem každého z nich je v zásadě to samé: reprodukovat zadaný text v co nejsrozumitelnější a nejpřirozenější formě.

Existují čtyři základní přístupy:

- syntéza pomocí segmentů (například difónová syntéza, kde jsou malé, dopředu nahrané vzorky lidské řeči spojené tak, aby tvořili smysluplná slova, viz níže),
- formantová syntéza (celá řeč je generovaná počítačem a ne lidskými vzorky, je jednoduchá na implementaci, avšak zní uměle),
- artikulační syntéza (přístup založený na modelu lidského vokálního traktu a reálných artikulačních procesech, není příliš rozšířená),
- HMM syntéza (matematický model, generující řeč na základě kritérií maximální pravděpodobnosti).

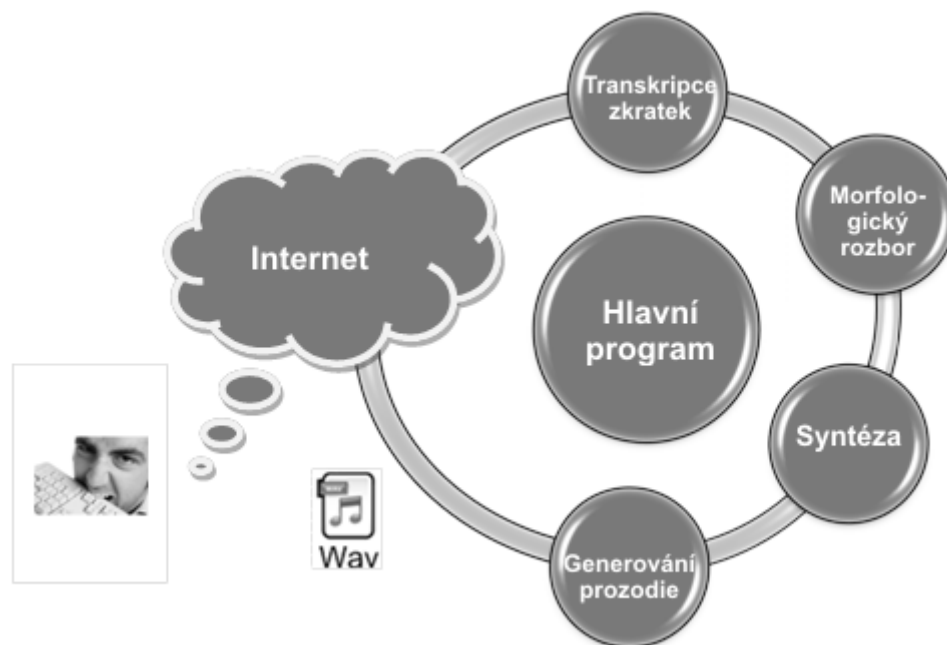
Na obrázku je blokový diagram všeobecného syntetizátoru. Samozřejmě, tento diagram je značně zjednodušený pro naše potřeby a některé elementy (jako například zpětné vazby nacházející se v některých učících se syntetizátorů) jsou vynechány. Avšak v podstatě každý syntetizátor se skládá z těchto částí:

- vstup textu
- předzpracování – slova anebo věty jsou „přeložené“ do speciální formy, resp. jazyka, kterému rozumí počítač: čísla a všechny zkratky jsou přepsány do slov ve správné formě, všechny písmena jsou napsána ve speciální abecedě – SAMPA abeceda. SAMPA (z angl. Speech Assessment Methods Phonetic Alphabet) je fonetický překlad používající jen tisknutelné ASCII znaky,
- samotná syntéza,
- post procesing – finální syntézu je ještě potřebné udělat více přirozenou, především modifikací prozodie (tempa řeči),
- výstupný blok poskytující syntetizovanou řeč.



Blokové schéma všeobecného syntetizátoru

Pro dosažení co nejpřirozenější syntézy řeči musí syntetizátory plnit komplexní úlohy jako předzpracování a následné zpracování (postprocessing). Pro co nejdokonalší podání řeči by tyto systémy měli být adaptivní a schopné se učit. Takový systém by se skládal ze čtyř základních modulů: fonetický překlad slov, určení slovního druhu (týká se především germánských a slovanských jazyků používajících skloňování), fonetického překladu zkratk a modulu modifikace prozodie.



Modulární architektura syntetizátoru

V následujícím příkladu se zaměříme na difonovou syntézu.

Příklad difonové syntézy řeči

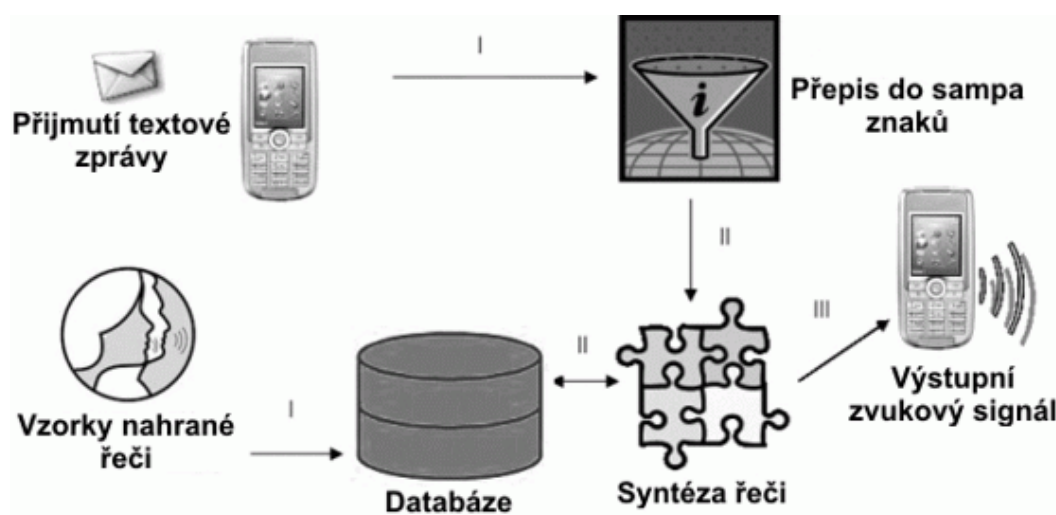
Následující příklad popisuje využití difonového syntetizátoru. Hlavní výhodou tohoto řešení je přirozeně znějící hlas a malá databáze. Slovenský jazyk má jen 1550 difon, a právě to dělá následující řešení velmi praktickým (zvláště v porovnání s jinými druhy syntéz, kde je velikost databáze mnohem větší, jako např. při korpusové syntéze).

Difóna je hned po fonémě jedna ze základních jednotek v řeči. Fonéma je nejmenší komplexní jednotka zvukového systému jazyka. Jednu difónu tvoří dvě po sobě následující fonémy. Hranice difóny jsou v polovině těchto foném, takže její délka v časové oblasti není dvojnásobná oproti fonémám, ale přibližně stejná. Výhodou difón oproti fonémám je to, že lepší sledují přechodovou změnu zvuku mezi jednotlivými hláskami a mají hranice v jejich středu, kde jsou časové průběhy nejstabilnější.

Obecně je počet difón v řeči rovný počtu druhé mocniny foném. Tím dostaneme všechny kombinace. Ve skutečnosti je ale počet difón menší, neboť ne všechny kombinace dokážeme vyslovit, resp. se v jazyce nenacházejí. Reálný počet difón

zjistíme jen důkladným studiem jazyka. Difónová databáze obsahuje reálné nahrávky řeči, které jsou rozděleny na menší části – difóny. Je více možností, jak vytvořit a nahrát databázi, například výběr takových slov, aby v nich byly obsaženy všechny difóny ze slovníku. Slova ve slovníku nemusí mít žádný význam anebo souvislost, důraz je kladen na co nejmenší počet nahrávek.

Návrh řečového syntetizátoru je na obrázku. Princip syntézy je tam znázorněn velmi zjednodušeně.



Návrh řečového syntetizátoru

Vstupní text se musí zesyntetizovat na řeč. Nejprve se musí text přepsat do SAMPA abecedy. Proto se v prvním kroku syntézy všechny znaky přepíše do SAMPY. V druhém kroku je přepis v SAMPE přepsán podle pravidel výslovnosti pro konkrétní jazyk (v našem případě slovenský jazyk). Potom se pro odpovídající difóny vyberou nahrávky z databáze a pospojují se. Výstupem je zesyntetizovaný text.



Uvádíme několik příkladů využití syntézy v praxi: osobní řečovní asistenti, asistenti v mobilních telefonech pro nevidomé, dopravní grafikon a navigační systém, internetové multimodální služby, aplikace pro dokumentaci nehod, služby manažování času anebo inventář. V posledních letech se staly velmi populární syntézy ve slovnících přímo na internetu (online), anebo čtečky knih s implementovanou syntézou textu, zvláště pro anglický jazyk.

6.2 Rozpoznávání obrazu

Typický problém s obrazem je detekce toho, zda se v obrázku nachází specifický objekt, anebo jestli má specifickou vlastnost. Tento problém je v současnosti vyřešen pouze pro specifické objekty (například detekce obličeje), ale ne pro obecné objekty (například seznam všech věcí, co se nacházejí na daném obrázku). Z tohoto důvodu rozeznáváme více oblastí rozpoznávání a detekce, jako například:



- Čárové kódy – čárový kód se skládá z černých čar různé tloušťky, které reprezentují číslice. Šířky jsou samozřejmě relativní, aby mohly být kódy skenovány z různých vzdáleností a různé velikosti.
- 2D kódy – též známé jako QR kódy, anebo maticové kódy jsou v zásadě rozšířením čárových kódů o další dimenzi. 2D kódy dokáží uložit mnohem větší množství informací než jednoduché čárové kódy, včetně URL adres, textu anebo čísel. Mohou mít různou velikost i hustotu a mají implementovány RS kódy, které mají schopnost detekce a opravy chyb.
- Optické rozpoznávání znaků (Optical Character Recognition OCR) – rozpoznávání znaků (písmen a číslic) z obrázků tištěného, anebo psaného textu. Tento proces je v zásadě specifickým případem rozpoznávání vzorů.
- Identifikace otisků prstů – druh algoritmu rozpoznávajícího vzory, který porovnává základní charakteristiky otisků (a pattern based algorithm compares basic fingerprint patterns (oblouky a smyčky) na zkoumaném obrázku s uloženým originálem.
- Specifické detekce – obraz je prohledáván za účelem identifikace specifických podmínek, například detekce pohybu v bezpečnostních kamerách.
- Rozpoznávání obličeje – obrázek je prohledáván za účelem identifikace specifických znaků obličeje.
- Rozpoznávání objektů – jeden, anebo více předem specifikovaných, anebo naučených druhů objektů je automaticky rozpoznáváno. Příkladem mohou být programy jako například Nokia Lens, anebo Google Goggles automaticky zobrazující popis různých objektů na obrazovce v reálném čase.

Každý typ rozpoznávání používá vlastní specifický algoritmus, který může být jednoduchý, ale i složitý – například použití statistických metod anebo neuronových sítí. Jako příklad algoritmu rozpoznávání uvádíme jednoduchý algoritmus detekce obličeje s body pro čelo a bradu.

Detekce bodů obličeje

Detekce bodů obličeje je založena na barevnosti lidské kůže a morfologické charakteristice lidské hlavy. Výstup algoritmu pro detekci kůže je důležitý pro oddělení významných bodů zkoumaného obličeje. Prvním bodem pro správnou detekci je odstranění šumu pomocí dolnoproputného filtru. Za takovýto filtrem nám nebudou vznikat nechtěné (a nesprávné) detekce. Dalším důležitým krokem je zjišťování hran. K tomu se používá tzv. Sobelův filtr. Po těchto nutných úpravách je obrázek předzpracován pro detekci. Následuje samotná detekce jednotlivých částí obličeje (jako jsou oči, nos a ústa, brada,...) založená na známých charakteristikách těchto bodů.

Detekce brady probíhá v oblasti pod dolním rtem a postupujeme směrem dolů. Horizontální prohledávání končí asi ve třetině obrázku, protože předpokládáme, že budou splněny podmínky pro vstupní fotografii (tj. že brada se nebude nacházet dále). Hledáme oblast, která je charakteristická relativně klidným průběhem s následným prudkým stoupáním, které označuje konec brady.

Detekce vychází z porovnávání horizontální vzdálenosti dvou bodů. Oba body leží na hraně – to znamená na hranici profilu. Ypsilonová souřadnice druhého bodu je však o 10 bodů (10 pixelů – je to hodnota určená konkrétně v algoritmu a vychází z požadavků na vstupní fotku) větší. Když horizontální vzdálenost takovýchto dvou bodů (tedy rozdíl x – ových souřadnic) je větší než je povolená vzdálenost, tak bod je označen jako bod brady.

Oblast vyhledávání bodu na čele jsme definovali v první třetině obrázku v horizontálním směru i vertikálním směru. Detekce probíhá na obrázku se zvýrazněnými hranami (ze Sobelova filtru) a na obrázku, na kterém je výsledek detekce kůže (Skin detect). Hledaný bod se nachází přibližně na místě, kde končí pokožka, a začínají vlasy.

Při detekci bodu definujícího čelo vycházíme nejdříve ze Sobelova filtru. Najdeme bod, který leží na hraně. Potom začneme prohledávat obrázek „Skin detect“ v horizontálním směru, přičemž postupujeme od detekovaného bodu směrem vpravo. Barevné složky bodů pokožky na „Skin detect“ obrázku mají hodnotu 0 (černá barva), barevné složky ostatních částí obličeje mají hodnotu 255 (bílá barva). Když najdeme první bod s hodnotou 255 (v naší oblasti prohledávání to znamená, že jsme našli část hlavy, kde už začínají vlasy), zjistíme rozdíl x – ových souřadnic. Když je menší než stanovená hranice, tak bod je právě náš hledaný bod – čelo.



Detekce brady



Detekce čela

Na základě výsledků detekce bodů obličeje se může deformovat obecný 3D model hlavy a může být personalizován.

6.3 Animace obličejů

Existuje mnoho metod pro animaci a modelování obličejů. Před tím, než se vybere správný způsob, je potřebné definovat požadavky na konkrétní animaci. Příklad takových požadavků na modelování a animaci je tento:

- Musí být animace nepřetržitá, kontinuální?
- Výkon musí být optimalizován pro PC anebo mobilní telefon?
- Musí se animace synchronizovat s řečí?
- Kvalita animace musí být co nejlepší, výsledná animace by měla být co nejvíce podobná skutečnosti.



Existují dva základní přístupy pro animaci obličejů: dvourozměrná anebo trojrozměrná a animace v reálném čase anebo zpětně vypočítaná animace.

Animace v reálném čase dovoluje uživateli interaktivně zasahovat do animace a redukuje čas potřebný pro přípravu animace. Nevýhodou této metody je kvalita obrazu, neboť čas a tím pádem i výkon na přípravu obrazu je omezen. Kalkulace potřebná pro výpočet každého obrázku by neměla být delší než přibližně 0.05s, protože je třeba dosáhnout alespoň 20 obrázků za sekundu. (20 FPS – obrazy (frames) za sekundu). Kvalita trojrozměrné animace je samozřejmě subjektivně lepší než kvalita dvourozměrné animace, protože výsledná animace je více přirozená.

V oblasti animace obličejů jsou nejznámější metody jako interpolace, parametrizace, simulace svalů atd.

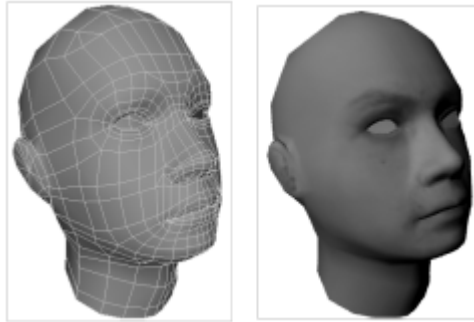


Je mnoho různých technik pro modelování lidského obličejů v prostoru, například polygonální modelování, modelování pomocí parametrických ploch, modelování menších částí.

Hovořící hlava v mobilním telefonu

Tato část představuje projekt aplikace syntézy řeči společně s animací obličejů na mobilním telefonu. Cílem tohoto projektu bylo vytvoření multimediálního komunikačního systému nepřenášejícího video a hovor, ale pouze text a pomocné informace pro animaci obličejů. Výsledkem byla mobilní Java aplikace fungující na mobilních telefonech, která dokázala číst krátké textové zprávy (SMS). Po přijetí SMS se na obrazovce objevil animovaný obraz hovořící hlavy vytvořený z fotografie odesílatele (fotografie byla předem uložena v databázi telefonu). Tato hlava následně četla přijatou zprávu.

V našem příkladu se model i vizuální prvky potřebné pro animaci nacházely přímo ve formátu objekt (OBJ), což znamená, že hlava neměla jen čáry a kontury, ale i texturu.



Neutrální model a textura obličeje

Vizéma je specificky deformovaný model obličeje. Deformace samozřejmě není libovolná, ale obličej je deformován tak, aby to vypadalo, že obličej vyslovuje příslušnou fonému. Model vizémy má ty samé čáry a body a schéma jako nedeformovaný obličej. Mění se pouze pozice uzlů modelu. Na základě této změny je model schopný poměrně jednoduše a s malým nárokem na výkon provádět interpolace uzlů jakož i změnu orientace povrchů definovaných příslušnými uzly.

Samotná animace je realizována zmíněnými interpolacemi vizém. Neutrální model je přečtený ze souboru (spolu s ostatními modely a vizémami). Interpolace se provádí mezi modely. Animace je prováděna mezi jednotlivými modely (model reprezentuje různě se tvářícího člověka, resp. různou mimiku obličeje). Animace je vytvářena v reálném čase. Model je deformován na základě výsledků detekce bodů obličeje (popsané v předešlé kapitole).



Vizémy

Animace se přehrává současně se syntetizovanou řečí. Řečový syntetizátor použitý v tomto projektu je identický s difónovým syntetizátorem, popsáním v kapitole Syntéza řeči. Velmi důležité je zesynchronizovat animaci a syntetizovanou řeč.



Ukázka aplikace na mobilním telefonu: personalizovaný model (první)
a obecný model (druhý a třetí)

6.4 Rozpoznávání řeči

Řečový signál vytvářejí lidské orgány řeči a původně je reprezentován vlněním vzduchu. Kromě jiného v sobě obsahuje lexikální složku, která je pro rozpoznávání řeči klíčová. Lexikální informace je zakódována v akustickém signálu jako sekvence akusticky odlišných zvuků. Každý jazyk obsahuje vlastní soustavu základních zvuků, které jsou přiřazeny k fonémám. Například ve slovenštině jich je 51. Naneštěstí, běžně vyslovené fonémy se vzájemně ovlivňují v časové i frekvenční oblasti, takže vytvářejí akusticky odlišné zvuky. Kromě toho se každá fonéma liší s každým jejím vyslovením a u každého řečníka (obsahuje charakteristické informace o řečníkovi). V každém reálném prostředí ještě navíc existuje aditivní a konvolutorní šum, který situaci dále stěžuje. A aby toho nebylo málo, každý jazyk obsahuje obrovskou slovní zásobu, obvykle několik stovek tisíc slov, ze kterých každé může nabývat více tvarů (pádů, časů, atd.). Je jasné, že úloha je pak dosti složitá a výpočetně náročná.

Po mnoho let se velké úsilí vynakládalo na stavbu a využití systémů ASR (Automated Speech Recognition) v oblastech jako jsou systémy pro přístup k informacím, dialogové systémy, atd., avšak až s rostoucími technologickými možnostmi se začínají vytvářet náročnější systémy: diktační systémy, či dokonce automatický přepis mluvené řeči. Tyto pokročilé systémy musí být schopny provozu v reálném čase, být nezávislé na řečníkovi, dosahovat vysoké přesnosti a podporovat slovníky skládající se ze stovek tisíc slov. V současnosti existují systémy na rozpoznávání spojitě nebo diktované řeči v reálném čase, přičemž velikost slovníku se pohybuje v řádech stovek tisíc slov.

Metody extrakce vlastností řeči

Jak už bylo zmíněno v úvodu, je řeč komplexní signál tvořený hlasovými orgány člověka. Takový signál v sobě zahrnuje různé typy informací jako: co se mluví, kdo mluví, jakou má náladu, odkud pochází (dialekt), jaké má sociální pozadí, je nemocný, má vadu řeči a mnoho dalších. Všechny tyto informace jsou „nepřímo zakódované“ pomocí mozku a hlasových orgánů do výstupního hovorového signálu.

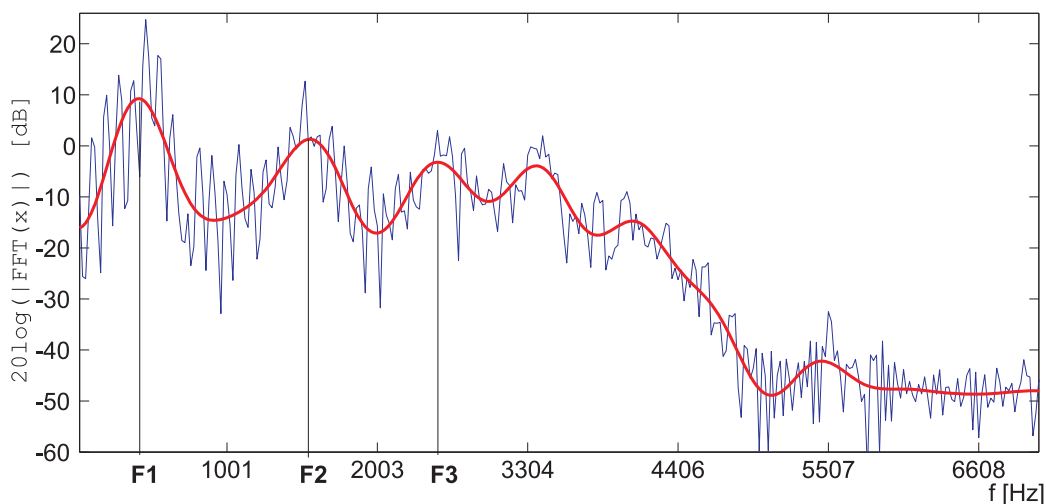


Podle šířky efektivního spektra řeči a počtu potřebných kvantizačních úrovní se dá odhadnout informační tok hovorového signálu, který se pohybuje okolo 100 kbit/s. Když se ale bude zkoumat informační hodnota lexikální složky obsažená v hovorovém signálu, pak je tento tok mnohem pomalejší, a sice cca 10 bit/s.

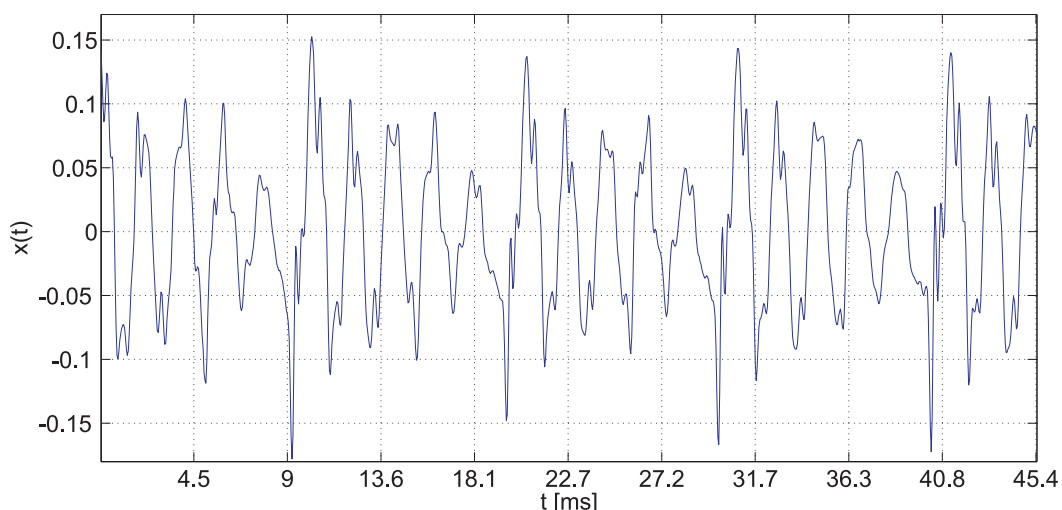
Je tedy zřejmé, že z řeči je potřeba extrahovat jen velmi malé množství informace, které bude odrážet právě jen lexikální složku. Tento proces je však velmi komplikovaný, neboť proces „kódování“ všech složek do výsledného hovorového signálu je složitý a ne zcela reverzibilní. Proto v prvním kroku v návrhu ASR systému se rozhoduje, kterou techniku výběru „lexikálních“ vlastností se má použít. Existuje množství metod, které buď simulují tvorbu hovorového signálu (lineární model tvorby řeči) anebo častěji se snaží simulovat lidský sluchový systém, matematicky ho popsat, zjednodušit pro praktické zpracování a případně

ho přizpůsobit pro správné a jednoduché použití s vybranými typy klasifikačních metod. To má vliv na efektivní potlačení aditivních a konvolučních šumů, které jsou přítomné v běžných podmínkách. Předpokládá se totiž, že evoluce selektivním výběrem „nastavila“ sluchové orgány spolu s příslušnými částmi mozku tak, aby tyto byly co nejvíce citlivé právě na řeč a lexikální informaci, čímž vlastně potlačují ostatní rušivé vlivy.

Požadovanými vlastnostmi metod extrahujících lexikální informaci jsou, kromě redukce bitové náročnosti, též vysoká citlivost na změny zvuku, které jsou lidmi vnímány různě. Naopak by citlivost měla být „hluchá“ na ty, které jsou nedůležité pro náš sluchový systém. Ukázalo se, že významnými indikátory rozdílu jsou tzv. formantové frekvence. Na následujících obrázcích je nakresleno frekvenční spektrum signálu a jeho časový průběh. Ve spektru jsou označeny polohy formantových frekvencí (z čehož je též jasný jejich význam), jakož i obálka spektra, která souvisí s postavením hlasových orgánů při vyslovení samohlásky „e“. Na obrázku časového průběhu hlásky „e“ je dobře pozorovatelná hlasivková frekvence.



Polohy formantových frekvencí a obálka spektra pro hlásku „e“



Časový průběh samohlásky „e“.

Aby bylo zřejmé, jak formanty souvisí s vyslovováním různých samohlásek, je v tabulce naznačeno jaké jsou průměrné polohy prvních třech formantových frekvencí pro muže a ženy – z nich je zřejmý kladný posun jejich polohy pro ženy.

Polohy formantových frekvencí pro samohlásky odděleně pro muže a ženy

| Hláška | Muži | | | Ženy | | |
|--------|---------|---------|---------|---------|---------|---------|
| | F1 [Hz] | F2 [Hz] | F3 [Hz] | F1 [Hz] | F2 [Hz] | F3 [Hz] |
| a | 730 | 1100 | 2450 | 850 | 1200 | 2800 |
| e | 530 | 1850 | 2500 | 600 | 2350 | 3000 |
| i | 400 | 2000 | 2550 | 430 | 2500 | 3100 |
| o | 570 | 850 | 2400 | 590 | 900 | 2700 |
| u | 440 | 1000 | 2250 | 470 | 1150 | 2700 |

Na základě mnoha experimentů bylo zjištěno, že následující fenomény jsou posuzovány jako výrazně vnímatelné:

Rozdíly v počtu formantových frekvencí

Výrazné rozdíly v poloze formantových frekvencí

Výrazné rozdíly v šířce pásem formantových frekvencí

Rozdíl energií signálu (a frekvenčních složek) je vnímán spíše poměrově a nikoliv absolutně. Člověk tedy vnímá intenzitu zvuku nelineárně, tj. hlasitější části potlačuje a slabší zesiluje. Matematicky se tato vlastnost dá aproximovat logaritmem.

Naopak obecně existují následující spektrální rozdíly, které nepřispívají k výrazně odlišnému vjemu zvuků:

- Sklon spektra ve tvaru $X(\omega)=S(\omega)*\omega\alpha$, kde α je faktor sklonu
- Ořezání nízkofrekvenčních složek ležících před první formantovou frekvencí F1

Ořezání vysokofrekvenčních složek ležících nad třetí formantovou frekvencí F3

Potlačení úzkého frekvenčního pásma, a to i v pásmu potřebném pro dosažení základní srozumitelnosti

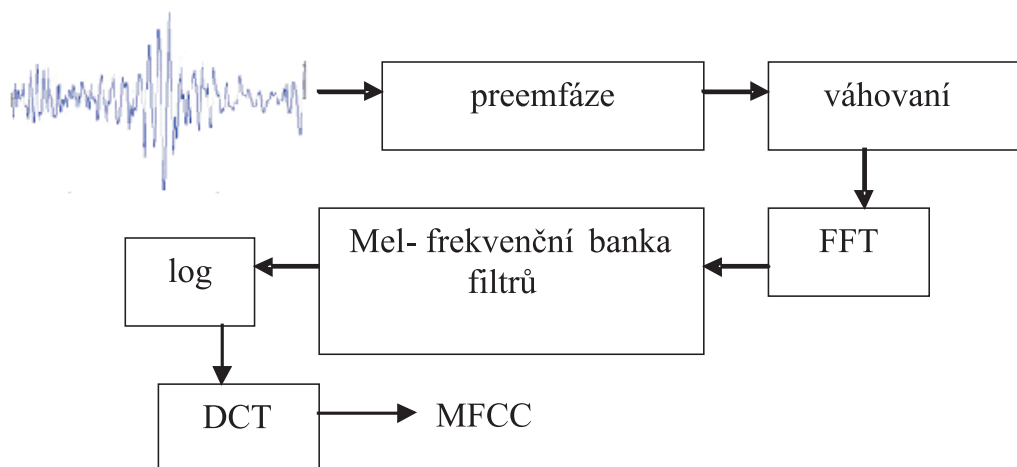
Navíc vlastnosti extrahovaných příznaků by neměly být citlivé na aditivní (šum pozadí) a konvolutorní šum (vliv nahrávacího zařízení a místnosti – ozvěny). Tyto podmínky však je těžké současně splnit. Proto je vhodné, aby alespoň řečové příznaky byly prezentovány tak, že tato rušení budou v prostoru vlastností lehce lokalizovatelné a potlačitelné. Navíc dobré příznaky musí být lehce extrahovatelné současnými matematickými postupy a technickým vybavením v reálném čase.



Současně nejpreferovanější akustické příznaky jsou MFCC a PLP, které jsou navrženy na zachycování pozic a šířek formantů, které jsou akusticky vnímatelné. Samozřejmě existuje celá řada dalších metod, které mohou dosahovat ještě lepší vlastnosti, ale tyto jsou většinou specializovány na užší oblast praktického použití, např. jsou určeny pro konkrétní prostředí.

Výpočet MFCC koeficientů

Zkratka MFCC znamená Melovy frekvenční cepstrální koeficienty. Tyto koeficienty tedy popisují modifikované cepstrální koeficienty, které jsou počítané nikoliv ve frekvenční, ale Melové škále, která lépe odráží psychoakustické vlastnosti lidského vjemu. Postup výpočtu MFCC koeficientů je graficky znázorněn na obrázku.



Postup výpočtu MFCC

Výpočet PLP koeficientů

Zkratka PLP znamená perceptuální lineární predikční koeficienty. Tyto koeficienty též popisují modifikované cepstrální koeficienty, ale jejich postup extrakce je komplexnější než v případě MFCC, protože zahrnuje více psychoakustických jevů. Jejich výpočet postupně obsahuje segmentaci signálu a Hammingovo váhování, výpočet spektra (fft), transformaci na Barkovu frekvenční osu, vyhlazení (DP filtrace) Barkova spektra (simuluje funkci kritická pásma), vzorkování spektra, které odpovídá šířce kritického pásma, - tedy 1 bark, váhování kopírující křivku stejné hlasitosti, transformace energie do hlasitosti, výpočet modelu produkce řeči (LPC) a přepočtení modelu do cepstra. I přes složitější výpočet poskytují MFCC a PLP podobné výsledky.

Pomocné koeficienty – dynamika a energie

Řeč je úzce spjata s plynutím času, a proto je velmi důležité obsáhnout v základních akustických parametrech také jejich vývoj v čase. Pro tento účel se používají tzv. dynamické koeficienty, které aproximují první a druhou derivaci příznaků v čase. Dalším parametrem je energie signálu. Samotná energie však samozřejmě nenesou dostatek akusticky diskriminační informace, a proto se užívá jen zřídka, což však neplatí o jejich dynamických vlastnostech v čase, které v hrubém nárysu popisují plynutí řeči.



Na závěr by se však mělo poznamenat, že proces extrakce příznaků není, pro svoji komplexnost, doposud úplně vyřešen a stále se pro tuto oblast vynakládá velké úsilí. Existuje mnoho příznaků a metod jejich výpočtu, ale zatím žádná nesplňuje všechny požadované vlastnosti.

Techniky sloužící k rozpoznávání řeči

Po extrahování vhodných příznaků řeči je každý hovor popsán jako časová posloupnost vektorů příznaků. V dalším kroku je proto potřeba porovnat neznámou posloupnost příznaků, která může odpovídat buď slovu, frázi anebo i celé větě se vzorem, o kterém víme, jakému hovoru patřil. Tyto promluvy však musely existovat ve fázi tréninku. Podle toho, zda rozpoznávání je založeno na rozpoznávání menších akustických jednotek jako jsou fonémy, či slabiky anebo větších jako jsou slova či celé fráze, rozlišujeme systémy založené na rozpoznávání menších akustických jednotek menších než slova (např. hlásky) U celých slov nebo frází je možno postihnout tzv. koartikulační jev, kdy výslovnost jednotlivých hlásek je ovlivněna výslovností konkrétního slova. To však naráží na problém, že všechna slova nebo fráze musí být v tréninkové databázi obsaženy víckrát, což je u rozsáhlých slovníků nerealizovatelné. Lepší volbou je použít jednotky menší než slovo, kterých je menší počet, ty ale nemusí postihnout fenomén koartikulace.

Dále se systémy dělí podle toho, zda rozpoznávají izolovaná slova anebo spojitou řeč, což je složitější problém, protože není zřejmé, kde jednotlivá slova začínají a končí. Specifický problém porovnávání vzorů mezi referenčním (známým) a neznámým vzorkem řeči spočívá v tom, že i stejné promluvy mohou, a skoro

vždy mají různé délky (počty vektorů příznaků). Situace se komplikuje tím, že dochází k nerovnoměrnému prodlužování anebo zkracování jednotlivých částí (hlásek) v rámci slov. Proto není vhodné použít jednoduché principy lineárního zkracování (decimace) nebo prodlužování (lineární interpolace) posloupnosti vektorů.

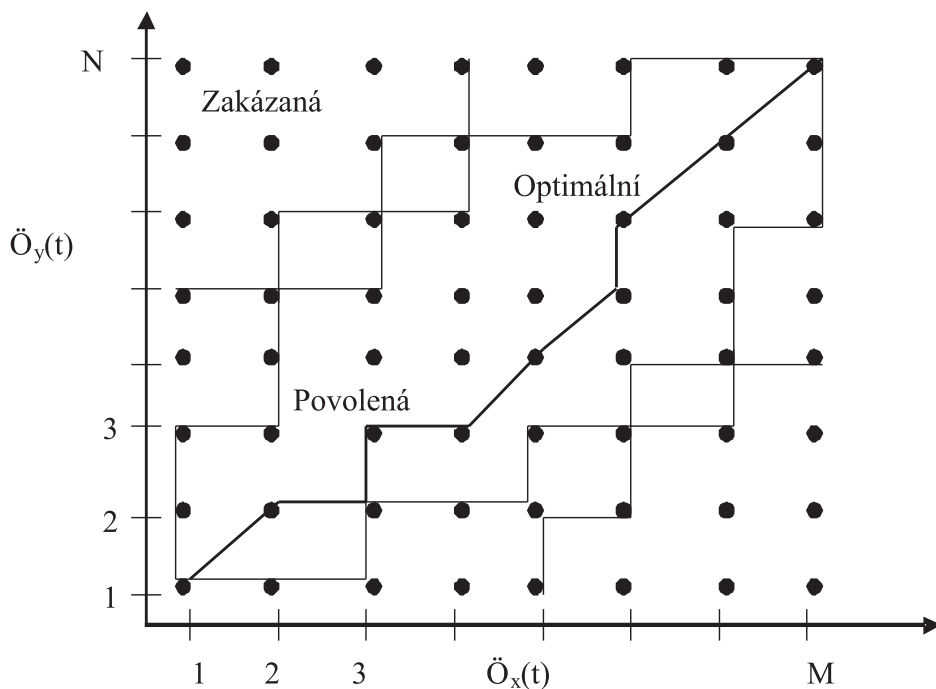
Existuje více metod, které se pokoušejí řešit tento problém, ale nejúspěšnější z nich se jeví DTW a HMM. Proto budou tyto dvě v dalším textu popsány.

DTW

DTW (Dynamic time Warping) je metoda, která slouží pro akustické porovnávání dvou vzorů (příznaků řeči) referenčního a vstupního slova, přičemž jsou zohledněny specifické vlastnosti řeči. Jde hlavně o nelineární časovou míru při porovnávání příznaků příslušných ke vzorovému a testovanému slovu. Tato vlastnost je nutná pro eliminaci vlivu různých délek vyslovování stejného slova. Dále v sobě zahrnuje řešení častého případu variací různých délek jednotlivých hlásek uvnitř slova, což se nedá řešit jednoduchým lineárním přechíslováním (roztáhnutím resp. stlačením) časové osy referenčního nebo testovaného hovoru.

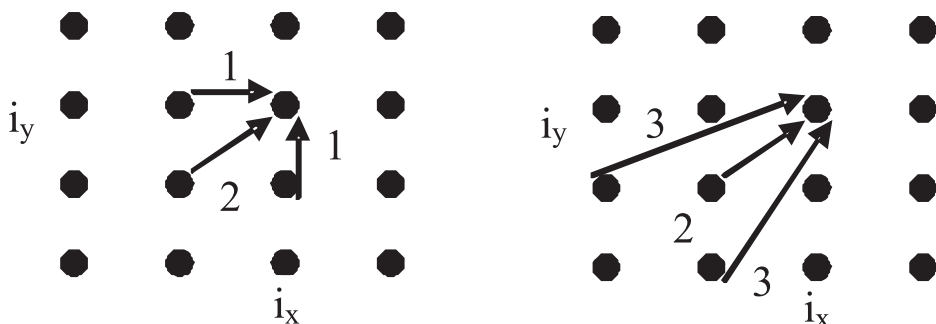
— V základním použití této metody se předpokládá poznání přesných hranic obou porovnávaných hovorů. To si v praxi vyžaduje dodatečné použití algoritmu detekce řeči.

Jádro metody vychází z principu dynamického programování řešícího problém nalezení minimální cesty mezi dvěma body na stanovený počet kroků. Tedy hledá se nelineární mapování (ohýbání času) indexů (vektorů) neznámého slova na referenční, aby byla dosažena nejmenší akustická vzdálenost mezi nimi. Toto mapování musí ale splňovat jisté přirozené požadavky jako jsou začáteční akonečné vektory slov se musí mapovat na sebe, funkce času pro indexy referenčních i testovacích vektorů musí být rostoucí a musí existovat nějaký rozumný práh maximální časové vzdálenosti mezi mapovaným indexem referenčním a testovacím. Při samotném výpočtu se používají dvě matice, jedna tzv. lokální, kde jsou vypočteny akustické vzdálenosti jednotlivých vektorů příznaků řeči mezi sebou a druhá globální, ve které se kumulují tyto chyby při hledání možných minimálních cest s přihlédnutím k různým lokálním omezením na pohyb vpřed. Ukázka globální matice vzdáleností je na obrázku.



Globální matice vzdáleností při DTW s optimální cestou, povolenou a zakázanou oblastí

Na dalším obrázku jsou zase pro úplnost zobrazeny některé z používaných lokálních cest i s váhováním.



Dve možné lokálne cesty pre pohyb v globálnej matici aj s váhovaním smerov



Metoda měla významné postavení při počátku budování jednoduchých systémů ASŘ, ale vzhledem k potřebám kontinuálního rozpoznávání řeči, kde je třeba slova řadit jednoduchým způsobem za sebou, přičemž se ještě vyžaduje i nezávislost na hovořícím, ztratila svůj význam a byla pro větší aplikace úspěšně nahrazena statistickou metodou modelování řeči pomocí HMM .

Skryté Markovové Modely (HMM)

Metoda skrytých Markovových modelů je založena na statistickém modelování hovorového signálu, respektive pro jeho části, které se návrhář systému rozhodne použít, tj. slova, slabiky, fonémy, subfonémy, atd. Její velkou výhodou je její

nezávislost na hovořícím, neboť model řeči je možno trénovat pro různé řečníky současně, jakož i lehké spojování jednotlivých modelů za sebou, což vede až ke spojitému rozpoznávání řeči. Metoda vychází z konceptu Markovových řetězců prvního řádu, které matematicky jednoduchým způsobem popisují časový sled událostí pomocí stavů a jejich pravděpodobností. Díky tomu je výpočetně velmi efektivní, i když úplně neodráží přirozený časový vývoj běžné řeči.



Markovův řetězec je dán stavy S_1 až S_n , přechodovou maticí $P_{N \times N}$ pravděpodobností přechodu mezi stavy a vektorem počátečních pravděpodobností π , následovně:

$$P = \begin{bmatrix} p_{11} & p_{12} & \dots & p_{1N} \\ p_{21} & p_{22} & \dots & p_{2N} \\ \dots & \dots & \dots & \dots \\ p_{N1} & p_{N2} & \dots & p_{NN} \end{bmatrix}, \pi = [P(S_0 = 1), P(S_0 = 2), \dots, P(S_0 = N)]$$



Potom pravděpodobnost výskytu nějaké posloupnosti stavů na nějakém modelu je dána

$$P(S_1, S_2, \dots, S_{N-1}, S_N) = \pi(s_1) p_{12} \dots p_{(N-1)N}$$

Takovýto model však neobsahuje žádné propojení stavu s vektorem příznaků. Když se předpokládá, že stav bude korespondovat s nějakou stacionární částí řeči, např. nějakou samohláskou, potom je zřejmé, že minimálně pro více hovořících bude existovat i více realizací vektoru příznaků. Proto musí stav v sobě zahrnovat dodatečnou pravděpodobnost s jakou generuje takovéto vektory v daném stavu, tj. $P(X/S_i)$. Kombinací těchto pravděpodobností potom vznikne výsledný model nazývaný skrytý Markovův model. Existují tři základní možnosti jak modelovat pravděpodobnost pozorování ve stavu. Podle toho rozlišujeme tři typy HMM:

- Diskrétní HMM
- Spojité HMM
- Polospojité HMM

Diskrétní HMM modely předpokládají, že vstupní posloupnost vektorů příznaků je konečná, což se realizuje pomocí vektorové kvantizace (VQ). Potom každý vstupní vektor je nahrazen některým z konečné množiny vektorů z kódové knihy. Jestliže VQ má L vektorů, pak v každém stavu je potřeba znát pravděpodobnost těchto vektorů, tedy jen L čísel.



Tento postup je poměrně jednoduchý a rychlý. Jeho další výhodou je jistá robustnost vůči malým zkreslením způsobených šumem.

- Při procesu VQ se však dopouštíme trvalé kvantizační chyby. Navíc, pro malý počet trénovacích vzorků a malý rozsah kódové knihy dochází k poměrně velké chybě.

Spojité HMM představují v současnosti ověřený standard, který poskytuje dobré výsledky pro široké spektrum použití.

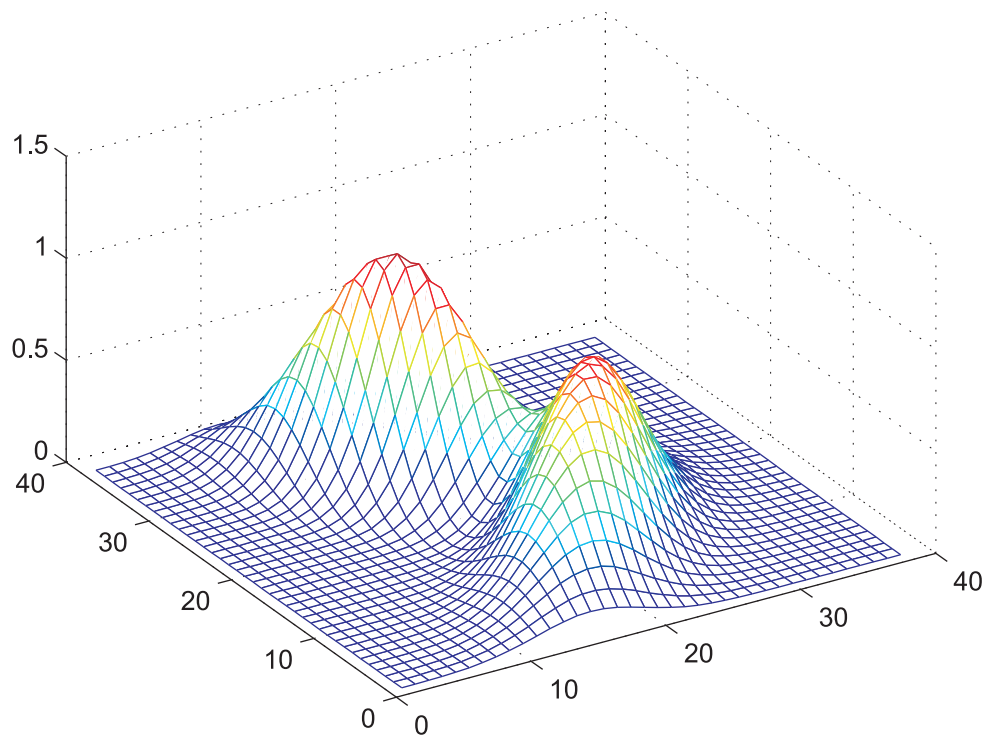


Pravděpodobnost pozorování vektoru příznaků v daném stavu $P(X/S_i)$ je dána směsí Gaussových rozdělání takto:

$$f(x/S_i) = \sum_{i=1}^P c_i \frac{1}{\sqrt{(2\pi)^d |U_i|}} e^{-\frac{1}{2}(x-\mu_i)' U_i^{-1} (x-\mu_i)}$$

Z uvedeného plyne, že takovýto model je dán vektory středních hodnot (μ), kovariančními maticemi (Σ) a váhami (c_i).

Směs dvou Gaussových dvourozměrných rozdělání je zobrazena na obrázku.



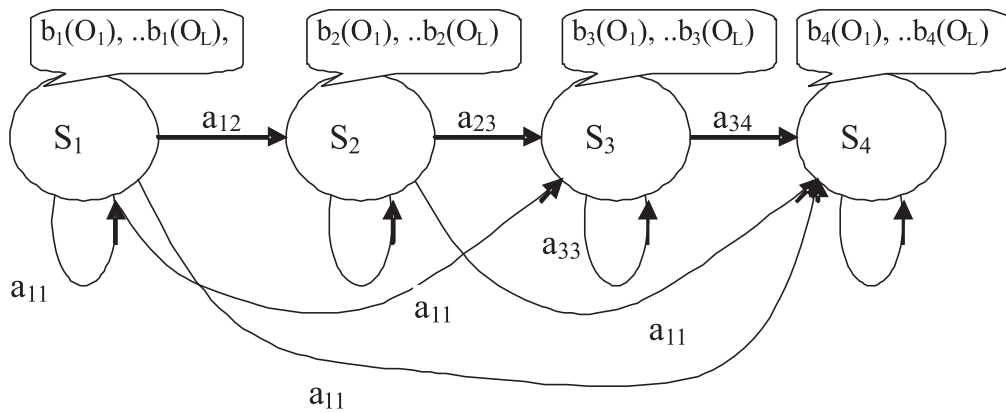
Opis 2D priestoru pomocou dvoch Gaussových zmesí.

- + Takovýto model je schopen dobře popsat prostor parametrů i při malém počtu vzorků.

Jeho nevýhodou je ale vyšší výpočetní i paměťová náročnost.

Polospojité HMM se snaží kombinovat výhody obou předchozích. Postup je takový, že se vytvoří jednotný popis vícerozměrného prostoru pomocí množiny Gaussových funkcí (může jich být až několik set) na rozdíl od VQ, kde jsou oblasti dány jen centroidami. Tyto směsi jsou následně využívány všemi stavy společně, tj. nebude pro každý stav vytvořena vlastní množina Gaussových funkcí, čímž se zredukuje množství dat a ulehčí se i samotný výpočet. Potom se v jednotlivých stavech už jen pamatuje pravděpodobnost pozorování vektorů na dané směsi pro daný stav.

Příklad levopravého čtyřstavového diskretního HMM modelu je znázorněn na obrázku.



Čtyřstavový levopravý diskretní HMM model



Potom pravděpodobnost pozorování řetězce vektorů příznaků o délce T na modelu λ s N stavy se počítá rekursivně pomocí pomocné proměnné α .

$$P(\mathbf{x}_1 \dots \mathbf{x}_T | \lambda) = \sum_{i=1}^N \alpha_T(i), \text{ kde } \alpha_t(j) = \left[\sum_{i=1}^N \alpha_{t-1}(i) * a_{ij} \right] * P(\mathbf{x}_t / S_j) \quad j=1, \dots, N$$

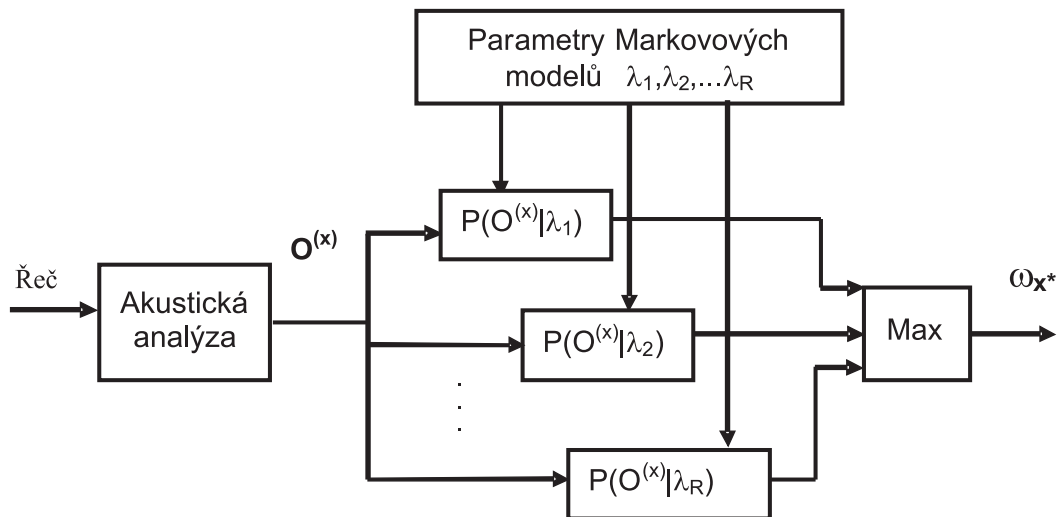
$$\text{a } \alpha_1(j) = \pi_j P(\mathbf{x}_1 / S_j)$$

$$\text{kde } j=1, \dots, N, t=2, \dots, T \quad a_{ij}(j) = \pi(j) * pP(O_1 / S_j)$$

Výsledná pravděpodobnost je dána:

$$P(O / \lambda) = \sum_{i=1}^N a_T(i)$$

Potom se rozpoznávání provede zvolením HMM modelu λ s nejvyšší pravděpodobností, jak je to schematicky znázorněno na obrázku.



Princip rozpoznávání pomocí HMM

Je založena na konceptu Markovovy řady, díky čemuž je výpočetně velmi efektivní i přesto, že neodráží časový vývoj přirozené řeči. Tedy každý model se musí odhadnout použitím obvykle velmi velkého souboru trénovacích příkladů, které obsahují více nahrávek stejného slova (jeho různé vyslovení). HMM je definován apriorními pravděpodobnostmi (π) začátku v určitých stavech, přechodovou maticí (přechodové pravděpodobnosti mezi stavy, a_{ij}) a pravděpodobnostním rozdělením (likelihoods) generování pozorovacích vektorů v daném stavu, $P(x/s_i)$. Tato rozdělení nejsou předem známa, ale většina těch široce používaných je směsí vícerozměrných normálních rozdělení (GMM). 4stavový model zleva doprava je znázorněn na obrázku níže.



Velkou výhodou HMM modelů je možnost zřetězení několika modelů do řetězce, tedy vyjádření libovolné délky může být vybudováno použitím souboru základních modelů.

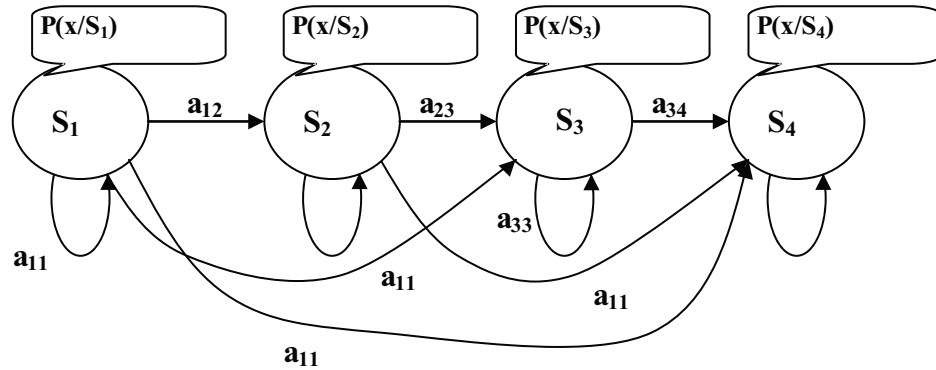


V praktické situaci se kvůli rychlosti a řetězení modelů počítá pouze cesta s maximální pravděpodobností, např. $P(x_1, S_{t1}, x_2, S_{t2}, \dots, x_T, S_{tT})$ a zpětně je určena posloupnost skrytých stavů, resp. modelů použitím Viterbiho algoritmu.

Složitou otázkou je trénování takovýchto modelů, tj. najít neznámé parametry, aby pro množinu pozorování daly maximální pravděpodobnost. Neexistuje analytické řešení tohoto procesu, a proto se provádí iteračně, přičemž je zaručeno nalezení pouze lokálního maxima. Existuje více způsobů výpočtu, ale nejznámějším je tzv. Baum-Welche algoritmus. Kromě trénovacích strategií je též důležitá struktura HMM modelu. Obvykle se užívá model zleva doprava pro řečové jednotky. Je všeobecně akceptováno, že každý akustický stav je modelován třemi stavy HMM za účelem zachycení jeho začátku, středu a končící části. Na základě teorie klasifikace jsou modely HMM náchylné na „nadměrné přeučení“. Proto se na detekci a prevenci tohoto fenoménu musí použít validační soubor.



Nejmodernější postupy dnes využívají diskriminativní trénování (MMI, MCE), anebo modelování pozorování ve stavech pomocí tzv. Support Vector Machines a Neuronových sítí (Deep belief networks).



Čtyřstavový model HMM zleva doprava

Potom pravděpodobnost pozorování řetězce vektorů vlastností v modelu λ je dána:

$$P(\mathbf{x}_1 \dots \mathbf{x}_T | \lambda) = \sum_{i=1}^N \alpha_T(i), \text{ kde } \alpha_t(j) = \left[\sum_{i=1}^N \alpha_{t-1}(i) * a_{ij} \right] * P(\mathbf{x}_t / S_j) \quad j = 1, \dots, N$$

$$\text{a } \alpha_1(j) = \pi_j P(\mathbf{x}_1 / S_j)$$

Potom rozpoznávání se provede volbou HMM modelu (λ) s nejvyšší pravděpodobností. Velkou výhodou HMM modelů je možnost zřetězení více modelů do řetězce, tedy vyjádření jakékoliv délky může být vybudováno použitím souboru základních modelů. V praktické situaci se počítá jen cesta s maximální pravděpodobností, např. $P(x_1, S_{t1}, x_2, S_{t2}, \dots, x_T, S_{tT})$, a zpětně je určena posloupnost skrytých stavů (stavy se mohou vztahovat k posloupnosti slov) použitím Viterbiho algoritmu.

Kromě trénování strategií je též důležitá struktura HMM modelu. Obvyčejně je použit model zleva doprava pro řečové jednotky a nejvšeobecnější je pravděpodobnost Bakisovy struktury. Je obecně akceptováno, že každý akustický stav je modelován třemi stavy HMM za účelem zachycení jeho začátku, středu a končící části. Základní HMM model může modelovat fonémy, slabiky, slova anebo i celé věty. Je to kompromis mezi přesností modelování a výpočetní flexibilitou a dostupností trénovacích dat. Některé častá, obvykle krátká slova mohou být modelována samostatným HMM modelem. Dobrou vyváženost možno dosáhnout tzv. navázaným trifónovým modelem, který používá většina existujících systémů. Necht' je použito jakékoliv trénovací kritérium, neexistují přesné vzorce pro optimální parametry, musí se tedy použít mnoho iteračních cyklů. Tím se však zabezpečí nalezení pouze lokálního maxima. Na základě teorie klasifikace jsou modely HMM náchylné na nadměrné přeučení, proto se na detekci a prevenci tohoto fenoménu musí použít validační soubor.

6.5 Multimodální rozhraní

Multimodální rozhraní jsou dnes velmi populární. Každý o nich mluví – o tom, že jsou velmi přirozené a oblíbené mezi uživateli. Multimodální rozhraní nabízejí řešení mnohých problémů s uživatelským rozhraním a taktéž umožňují aplikovat nové služby.

Multimodální rozhraní představuje kombinaci více modalit, čili způsobů interakce s počítačovými systémy. Multimodální rozhraní slouží k identifikaci hovořícího rozpoznáním obličeje a hlasu, tedy využívá rozpoznání a autentifikaci uživatele pomocí forem jako hlas, obličej, atd. Kromě toho se multimodální rozhraní používá i na ovládání *set top boxu* (**STB**) pomocí hlasu anebo gest.

Jedním z příkladů, kde je integrováno multimodální rozhraní, je projekt HBB-Next. Úlohou projektu je napomoci slučování vysílání a internetu vyvinutím technologie s ohledem na koncové uživatele. Projekt má obohatit klasickou TV o sociální aspekt – implementovat sociální síť, možnost přístupu k filmům z více zařízení (TV, PC, mobil) a odkazy k filmům. Taktéž má napomoci mixování vysílaných filmů, doplňkových videí na internetu a filmů vytvořených uživatelem.

HBB-Next je založen na modulární architektuře. Moduly v **HBB** (*Hybrid Broadcast Broadband* – hybridní širokopásmová televize) jsou navrženy tak, aby navzájem spolupracovaly. Například jak uživatel vstoupí do místnosti, systém ho rozpozná a automaticky se nastaví podle jeho požadavků. Když potom uživatel otevře aplikaci AppStore, systém mu dovolí otevřít, koupit a instalovat zvolenou aplikaci bez další autentifikace. Každá aktivita anebo operace uživatele může vyžadovat několikanásobnou autentifikaci určitým bezpečnostním stupněm. Zde je například zřejmá spolupráce modulu rozpoznání obličeje uživatele a modulu, který zabezpečuje autentifikaci.

Identifikace více hovořících vychází z nahraného audiosignálu, který může obsahovat hovory více osob. Tato úloha se může rozdělit do několika kategorií podle více detailů. Jednou kategorií jsou případy, kdy hovořící je známý a chceme zjistit, zda se jeho hlas nachází v nahraném signálu – to je možné tehdy, když byl součástí trénovací fáze detektoru. Potom se úloha mění na identifikaci jednoho konkrétního hovořícího. V případě, že žádný hovořící není znám, musí se použít techniky klastrování a segmentace. Cílem většiny aplikací tohoto druhu je to, aby nepřetržitě fungovaly: „poslouchaly“ přicházející zvuk zakódovaný v PCM vzorcích, detekovaly přítomnost hlasu (*voice activity detection*, **VAD**), dokázaly rozpoznat ticho a šum v pozadí a rozpoznaly překrývání řečníků. Když aplikace zaznamenají delší hlasový záznam, musí identifikovat hovořícího s určitou přesností. Při identifikaci jednoho hovořícího je přitom cílem identifikovat ho bez ohledu na významové hledisko věty.

Každý systém na identifikaci hovořícího se skládá ze dvou hlavních částí. Prvá je rozpoznání řečových parametrů z nahraného audio signálu, druhá je klasifikační metoda, která na základě už vybraných parametrů s určitou pravděpodobností určí hovořícího. Tyto systémy jsou většinou navrhovány pro konkrétní účel a zatím nejsou univerzální - navrhovatel aplikace musí na základě typu úlohy vybrat správnou kombinaci metod a jejich modifikací.

V případě rozpoznání řečových příkazů se implementuje systém na rozpoznání izolovaných slov. Nejúspěšnější a nejpoužívanější jsou ty, které pracují na principu HMM, statistického modelování řeči, obzvláště ty, které používají jako základní jednotku pro modelování kontextově závislé fonémy. Když se používá pevně určená skupina příkazů, potom se mohou na modelování použít celá slova, a tak dosáhnout potenciálně lepší přesnosti (lépe zachycený koartikulační efekt).

Obecně jsou dvě kategorie postupů sledování gest uživatele: vzhled a postup založený na 3D modelu. Postup založený na 3D modelu porovnává vstupní parametry končetiny v 2D projekci 3D modelu končetiny. Postup založený na vzhledu používá obrazové vlastnosti pro modelování vizuálního vzhledu končetiny a porovnává s vybranými obrazovými vlastnostmi z video vstupu. Při druhém postupu výsledek závisí na schopnostech výpočetního zařízení. Když se používá RGB kamera, metody se zaměřují na sledování barvy kůže anebo tvaru gestikulující části těla. Avšak postup hlavně závisí na světelných podmínkách stejně jako na stabilitě popředí a pozadí sledovaného subjektu. Taktéž se ve vyhodnocované oblasti nemohou objevit jiné objekty barvy kůže anebo objekty ve tvaru končetiny, protože mohou zmást algoritmus. Infračervená hloubková kamera používá svůj vlastní zdroj IR světla, a je proto mnohem odolnější vůči světelným podmínkám scény. Navíc je kamera schopna poskytnout hloubkovou mapu, pseudo 3D obraz scény, což může být velmi užitečné při sledování gestikulujících částí těla, např. ruky.

V současnosti je několik metod, jak se dá implementovat ovládání a kontrola očima. Nejdůležitějším bodem je správná detekce oční zorničky. V HBB-Next projektu byla použita nejjednodušší a nejpřirozenější metoda, která využívá RGB kameru zařízení Kinect. Princip je následující: osoba sedící před monitorem se bude bez pohybu dívat na vyznačené body na obrazovce. Aplikace odměří vzdálenost hlavy od monitoru. Pro potřeby dalších výpočtů je třeba zadat i velikost monitoru a následně se na základě Pythagorovy věty dají vypočítat parametry pohybu zorniček – jejich úhel a pohyb. Aplikace určí maximální rozsah pohybu zorničky tak, že nasměruje uživatele, aby se podíval na okraje monitoru, a vypočítají se i úhly odkloněné zorničky od základní pozice. Při známé vzdálenosti hlavy od monitoru umíme tedy přepočítat rozptyl a úhly při změně polohy zorničky tak, aby bylo stále zabezpečeno ovládání.

Techniky pro rozpoznání obličeje byly podrobně popsány v kapitole Rozpoznávání obrazu. V reálném systému je definován seznam požadavků podle toho, co všechno systém musí/měl by/může:

- Systém musí identifikovat uživatele v místnosti na základě jeho obličeje, když patří do skupiny lokálních uživatelů.
- Systém musí rozpoznat uživatele a porovnat jeho profil s uloženými profily, bez přístupu na internet.
- Systém může identifikovat uživatele v místnosti na základě jeho obličeje, přičemž uživatel je neznámý (nepatří mezi lokální uživatele).
- Systém může rozpoznat uživatele i v tmavém prostředí.

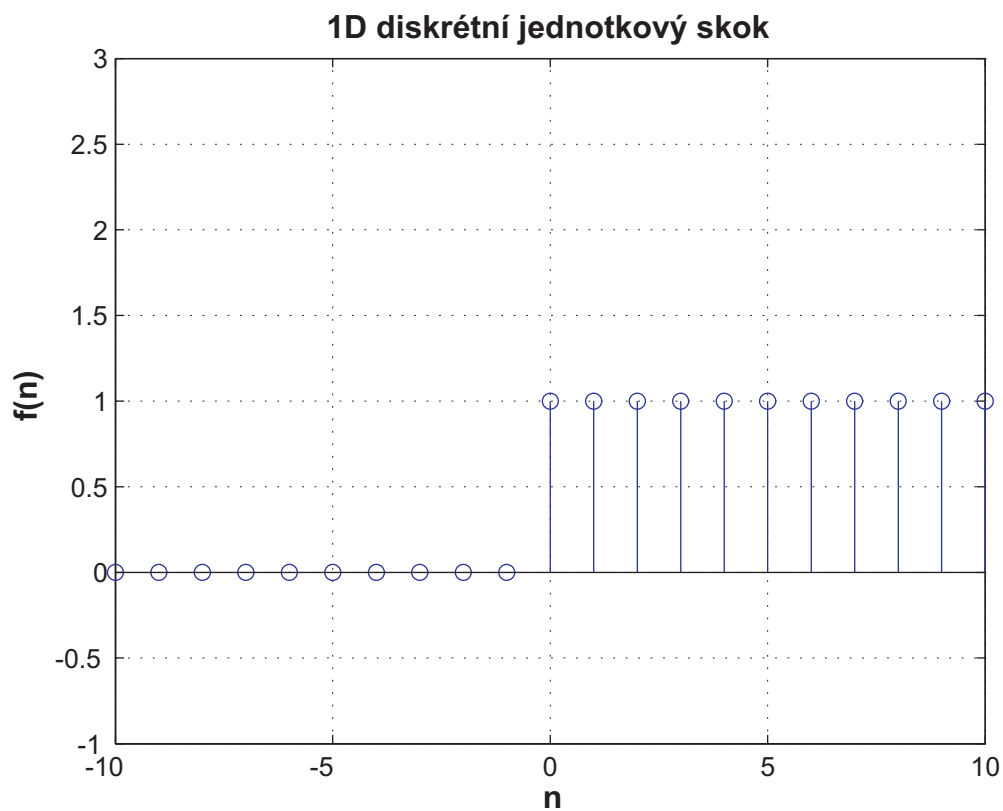
7 Příklady v Matlabu

7.1 Příklady v Matlabu



Příklad 1

Napište kód v Matlabu, který vykreslí jednorozměrný, diskrétní jednotkový skok.



Výsledek: Diskrétní jednotkový skok

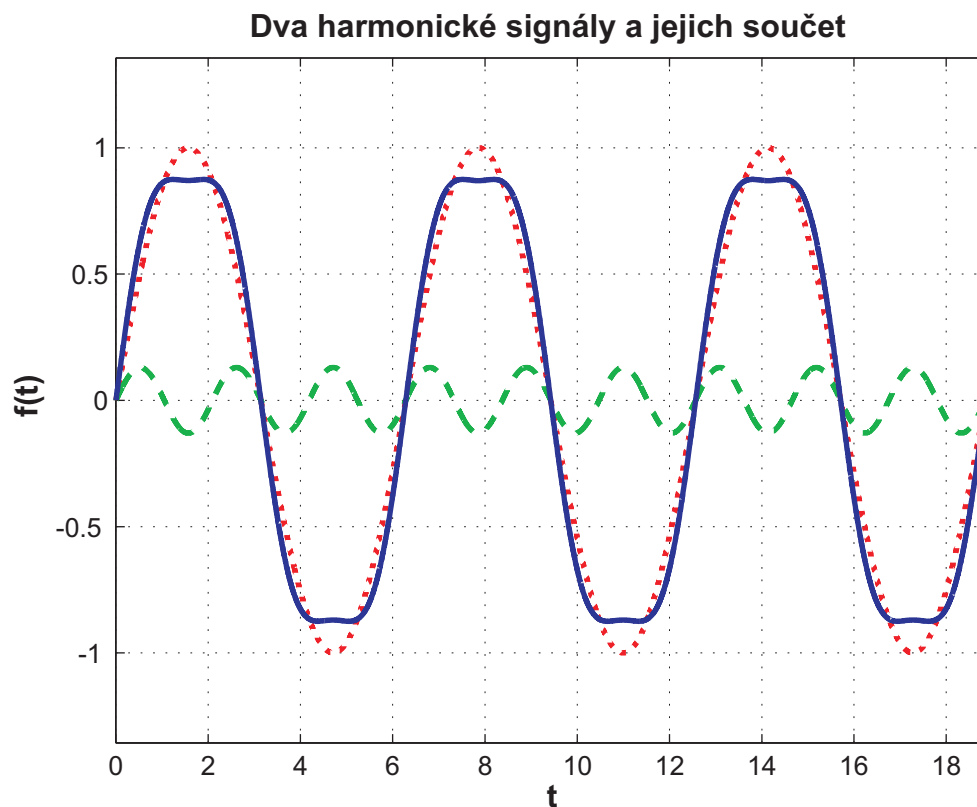
```
t=-10:20; %definice časové periody signálu
step=heaviside(t); %heaviside(x) je
                    funkce v Matlabu,
                    jejíž hodnoty jsou 0
                    pro x < 0, 1 pro x > 0,
                    a 0.5 pro x = 0.

step(t==0)=1;
figure; %příkaz na vykreslení
        obrázku
stem(t,step);
grid on;
xlabel('n');
ylabel('f(n)');
title('1D diskrétní jednotkový skok');
axis([-10 10 -1 3]);
```



Příklad 2

Napište kód v Matlabu, který generuje sinusový signál.



Výsledek: Dva harmonické signály a jejich součet

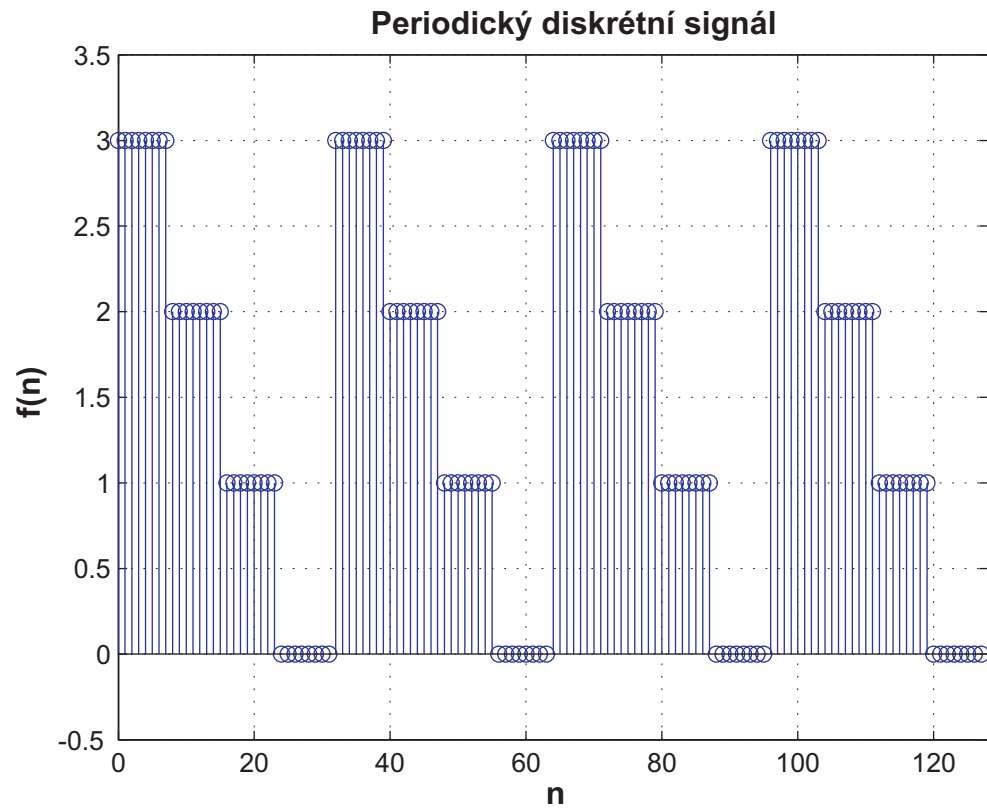
ŘEŠENÍ

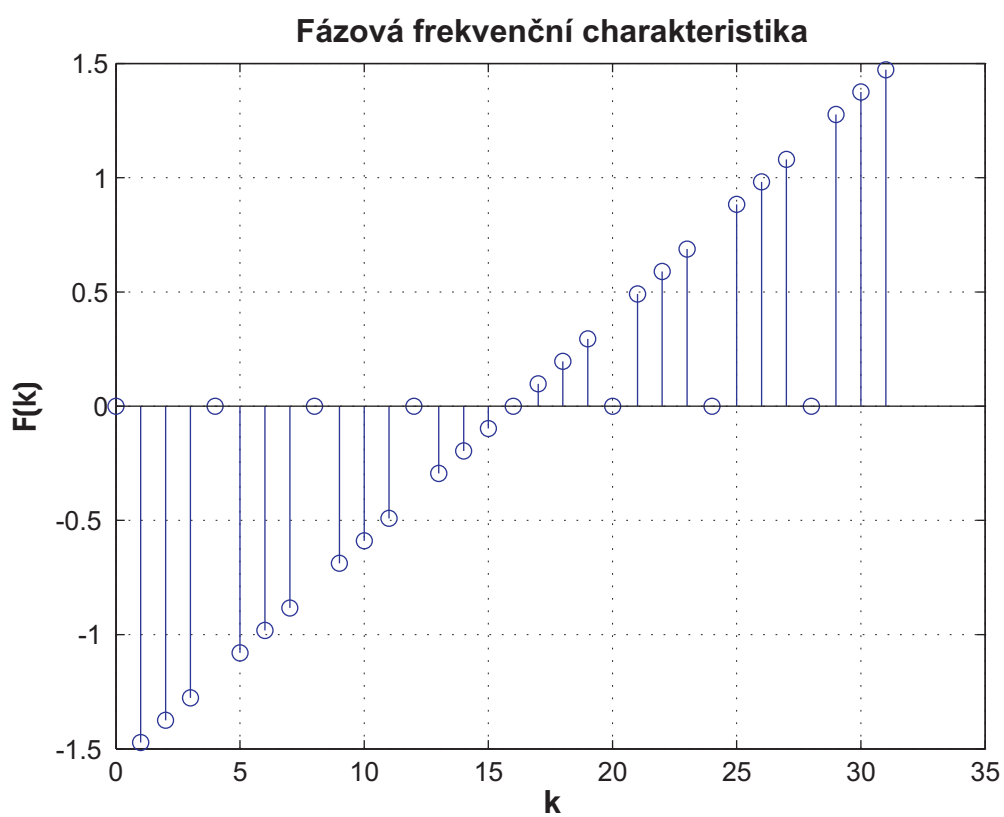
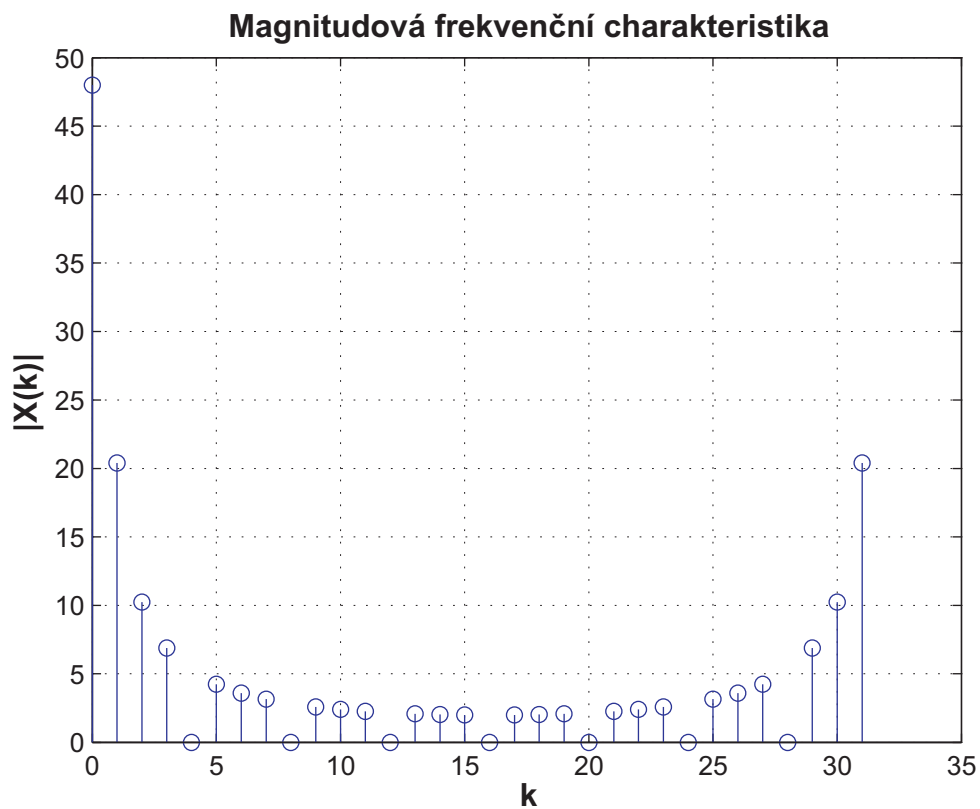
```
range=6*pi;          %časový rozsah signálu
t=0:0.001:range;    %časové body
A=[1 0.13];         %vektor amplitud
w=[1 3];           %vektor frekvencí [Hz]
phi=[0 0];         %vektor fází
sig1=A(1)*sin(w(1)*t+phi(1)); % definování
                                jednotlivých
                                signálů
sig2=A(2)*sin(w(2)*t+phi(2));
signal=sig1+sig2;
figure;
plot(t,sig1,':r','LineWidth',2);
hold on;
plot(t,sig2,'--g','LineWidth',2);
hold on;
plot(t,signal,'LineWidth',2);
grid on;
axis([0 rozsah -1.2*sum(A) 1.2*sum(A)]);
xlabel('t [s]');
ylabel('f(t)');
title(Harmonické signály a jejich součet);
```



Příklad 3

Napište kód v Matlabu pro určení diskrétní Fourierovy transformace a frekvenčních charakteristik.





Příklad přenosové funkce LDKI systému, magnitudová a fázová frekvenční charakteristika

```
count=32;
Ts=4/count; %vzorkovací frekvence
per=4;      %počet period
syms k;    %symbolické proměnné
syms n;
signal=[3.*ones(1,count/4) 2.*ones(1, count /4)
        ones(1, count /4) zeros(1, count /4)];
fn=[];
for n=1:per
    fn=[fn signal];
end
n=0:count*per-1;

figure;
stem(n,fn);
title('Diskrétny signál');
axis([0 length(fn) min(abs(fn))-0.5
      max(abs(fn))+0.5]);
grid on;

figure;
Xk=fft(signal); %diskrétní Fourierova
                transformace
os=0:length(Xk)-1;
stem(os,abs(Xk)); %magnitudová frekvenční
                  charakteristika
title('Magnitudová frekvenční charakteristika');
grid on;

figure;
stem(os,angle(Xk)); %fázová frekvenční
                   charakteristika
title('Fázová frekvenční charakteristika');
grid on;
```
