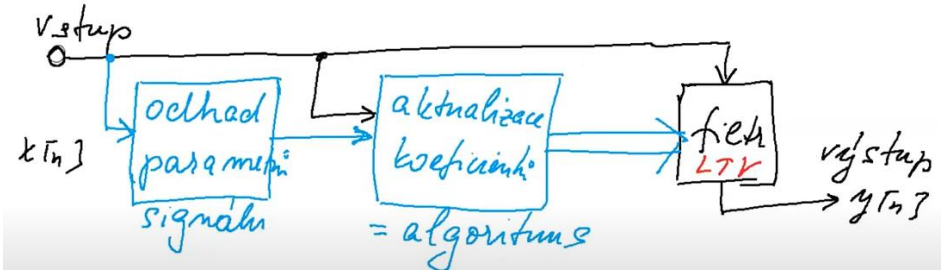
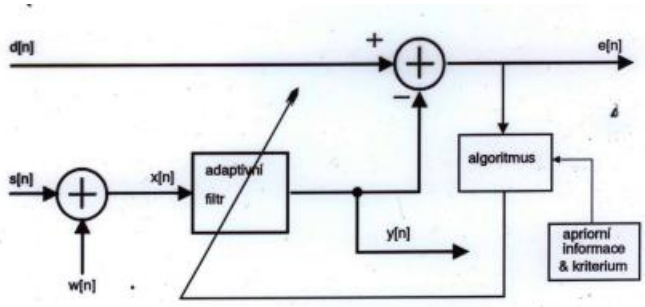
OTÁZKY KE ZKOUŠCE – ADA

1. **Definice adaptivního systému (AS) - základní pojmy – adaptivní systém a jeho části, účelová funkce (kritérium), chyba predikce/estimace, algoritmus, filtr, stabilita, konvergence, … příklady použití AS**

- Adaptivní systém = parametry soustavy se mění v závislosti na parametrech zpracovávaného signálu -> soustava reaguje na změny nebo se učí
- LTV – časově proměnné parametry
- Použití AS: rozpoznávání řeči a obrazů, automatické řídící systémy, telekomunikace, měřící technika
- části AS: filtr, algoritmus
- Základní prvky: Apriorní infromace = „když nic nevím, tak nic nevyřeším“
    - Souhrn předpokladů a údajů nutných k optimalizaci systému
        - Typ vstup. Sig., typ rušení, typ přenosové cesty
    - Kritérium kvality: návrh systému a hodnocení kvality, kompromis mezi požadavky kvality a realizovatelností
        - Často kvadratické: minimalizace energie chybového signálu
        - Kritérium -> účelová funkce = chybový povrch
    - Algorimus a filtr: filtr produkuje výstupní signál, jeho koeficienty nastavuje algoritmus tak, aby bylo dosaženo minima účelové funkce = konvergence
- AS filtr:



- Estimace signálu: vstup = signál+šum > adaptivní filtr, jehož koef. Se nastavují ve zpětné vazbě tak, aby energie signálu e[n] byla minimální



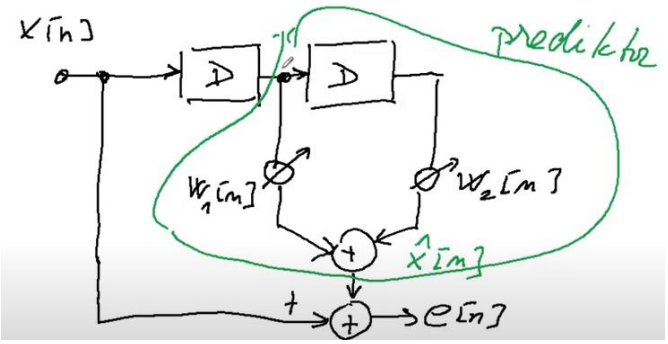
    - Cíl > návrh filtru, který potlačí rušení w[n], signály mají často proměnné parametry

- Chybový povrch (konvergence) (ES): „miska“, tvar a poloha je dána vlastnostmi filtru a signálu
    - Kulička překmitává v minimu, pohyb ve 2d = konvergenční stopa
    - Parametr filtru řídící rychlost konvergence > jak rychle se dostane do minima a jak moc bude překmitávat („tření pro kuličku“)
- Kritéria: MMSE => alg. LMS
    - LS => alg. RLS
- Odhady: rekurentní, průběžný
- Stabilita AS: schopnost systému udržet výkon a správně reagovat na změny v prostředí nebo jeho vlastní vstupní podmínky
    - Stabilita je dána konvergenční konstantou
    - Konvergenční konstanta **μ**: $0 < \mu < 1/Px$ (Px = výkon signálu)

- Tvarování paměti – nekonečná paměť (growing): rekurentní alg. (LTV)
    - exponenciální zapomínání: průběžný alg. (LTI)
    - klouzavé okno
- rovnice aktualizace: $w[n+1] = w[n] + \mu e[n].x[n]$

2. Typické úlohy adaptivní filtrace (AF) – filtrace/predikce, estimace, identifikace – schémata těchto úloh pro FIR filtry, algoritmy aktualizace koeficientů, podmínky činnosti, diskutovat rozdíly mezi úlohami

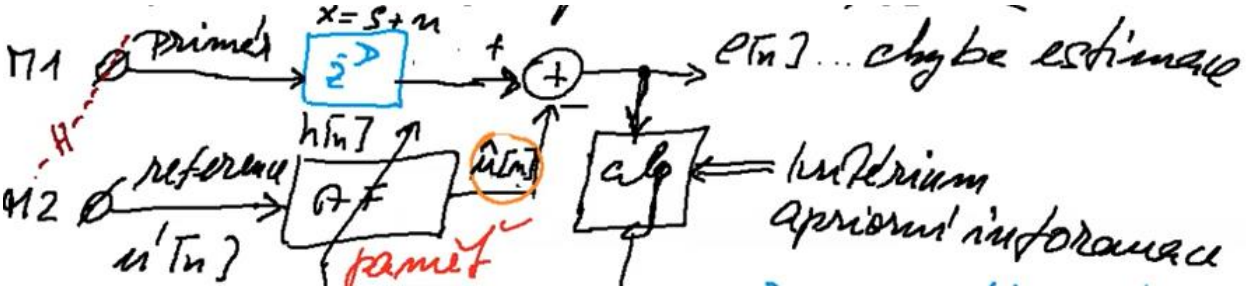
- Filtrace/predikce:
- 1 vstup
- rovnice filtrace: $e[n] = x[n] - \sum_{i=1}^{M} w_i[n]\, x[n-i] = x[n] - W_n^T x_{n-1}$
- kritérium MMSE: $min_{\boldsymbol{w}} J = min_{\boldsymbol{w}} E[e^2[n]]$
- Filtr druhého řádu (2xD), koeficienty $w_{1,2}[n]$ se nastavují (přeškrtnuté kolečko)
- Signál x[n] se porovnává s výstupem filtru (prediktoru) x^[n]



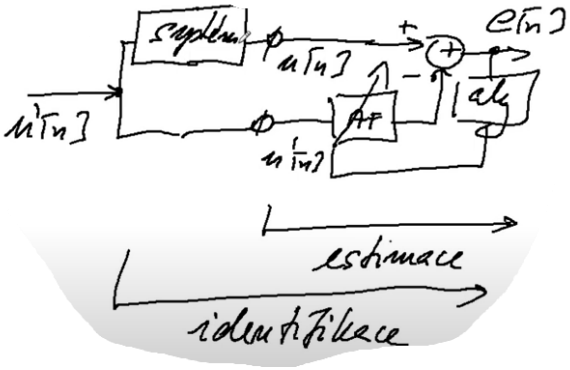
- Algoritmy pro nastavení koeficientů:
    - Blokový/dávkový
        - grad**J** = 0 > Yule-Walkerovy rovnice: $R_x w = r_P \Rightarrow w = R_x^{-1} . r_P$
        - jednou za 100 vzorků spočítáme korelace těch 100 vzorků, sestavíme normální (Y-W) rovnice a vyřešíme
        - Nasbíráme několik vzorků chybové posloupnosti e[n], vybereme vhodné kritérium, např. Min. Stř. Kvadr. Chyby (tu minimalizujeme), výsledkem budou lineární rovnice, jejichž řešení budou hodnoty w1,2
            - Např. Jednou za 100 vzorků se změní hodnoty w1,2
            - Předpokládáme, že v průběhu filtrace se nemění char. signálu

- Průběžný
  - S každým novým vzorkem mění hodnoty vah (w1,2)
  - Metoda největšího spádu:
    - Rychlejší nastavení koeficientů, ale méně přesně
    - Odhad záporného gradientu pomocí klouzavého okna
    - LMS odhad gradientu: $grad\boldsymbol{J}_n = -e[n]x_{n-1}$
  - LMS alg.: nastavení počátečních wah, nastavení počátečních podmínek, v cyklu filtrujeme a aktualizujeme váhy, dodržení podmínky stability

- Estimace: (kompenzační úloha)
- Apliakce LMS: kompenzace šumů, identifikace osustav, potlačování echa
- 2 vstupy
- Předpokládáme, že korelace mezi u[n] a u'[n] je nenulová (pokud nulová, nedokáže odhadnout šum) ... šumv primárním a referenčním vstupu = požadavek korelace mezi šumem v prim. A ref.
- Zpoždění zajistí kauzální systém, pozdrží vzorky než se objeví ve filtru = požadujeme kauzalitu
- Úkolem je odhadnout co nejlépe signál z reference: u^ = odhad u
- Správný odhad = kompenzace v součtovém budě
- Předpoklady: vzájemná energie s,u=0 a zároveň vzájemná energie s,u'=0
- e = s+u-u^ … $e[n] = d[n] - \widehat{W}_n x_n$



- Identifikace
- Bereme výstup systému, který chceme identifikovat
- Chceme širokopásmový signál u'[n] = požadavek širokopásmového buzení
  - Větší šířka pásma signálu, než je frekv. Rozsah identifikovaného sig.
- Požadavek nenulové korelace mezi šumem v prim. A ref. Sig. Je automaticky splněn
- Při identifikaci vyrábíme jeden ze signálu tak, že bereme výstup ze sig, který chceme identifikovat.



3. Rozdělení algoritmů – blokový, rekursivní s exponenciálním zapomínáním; LMS – podmínky, princip, souvislost s metodou největšího spádu a s RLS, struktura FIR-LMS algoritmu; RLS – podmínky, princip, souvislost s metodou největšího spádu a s LMS, struktura FIR-RLS algoritmu

- Dávkový (blokový): vynulujeme gradient účelové funkce -> získáme normální rovnice
  - Normální rce filtrace = Y-W, normální rovnice estimace = Wiener-Hopf
- Průběžné (LMS): nové koeficienty jsou staré koeficienty + oprava
  - Záporný okamžitý gradient je naše oprava... komžitý platí pro LMS

- RLS = Rekurzivní nejmenší čtverce ... nejmenší čtverce = Gauss
- Kritérium: LS
- A) Hledáme min hodnotu, přes koef. Které nám minimalizují tuto hodnotu
- (Alg. S rostoucí pamětí, neumí sledovat změny > čím více vzorků, tím mají menší váhy)
- B) rekurzivní s growing memory
- Alg: Víme estimaci, chceme nové řešení = staré řešení + oprava
  - w[n] = Pn . bn, řešení vhodné pro nestac. Signály, dáno součinem dvou matic
    - Pn = R_x ^-1, bn = r_dx
  - hledáme závislost: Pn na Pn-1 a bn na bn-1
  - rekurentní realizace sumy

$$\left.\begin{aligned} \underline{P}_n^{-1} &= \underline{P}_{n-1}^{-1} + \underline{x}_n \underline{x}_n^T \\ \underline{b}_n &= \underline{b}_{n-1} + d[n]\,\underline{x}_n \end{aligned}\right\} \quad \underline{P}_n^{-1}\underline{b}_n$$

  - Věta o inverzi matice

$$\underline{P}_n = \underline{P}_{n-1} - \frac{\underline{P}_{n-1}\underline{x}_n \underline{x}_n^T \underline{P}_{n-1}}{1 + \underline{x}_n^T \underline{P}_{n-1} \underline{x}_n}$$

nové | staré | oprava | skalár

  - Aktualizace Rx^-1

g) $\underline{w}[n] = \underline{R}_x^{-1}\underline{r}_{dx} = \underline{P}[n]\,\underline{b}[n]$ = dosadíme → upravíme =

=>

h) $\underline{w}[n] = \underline{w}[n-1] + \underline{P}_n\, e[n]\, \underline{x}_n$

- Porovnání RLS a LMS:

RLS: $\underline{w}[n] = \underline{w}[n-1] + \underline{R}_x^{-1} e[n]\,\underline{x}_n = \underline{w}[n-1] + \underline{R}_x^{-1}\frac{1}{2}\,\mathrm{grad}\,J_m$ (oprava)

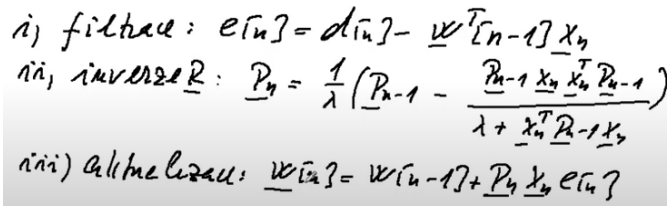
LMS: $\underline{w}[n] = \underline{w}[n-1] + \mu\, e[n]\,\underline{x}_n = \underline{w}[n-1] + \mu\frac{1}{2}\,\mathrm{grad}\,J_m$

  - Okamžitý gradient dělá opravu, Opravy se liší pro RLS a LMS
    - Rx^-1 = čtvercová matice, není naše volba, dána signálem x[n] ⇔korelační struktura
    - μ = matice s jedním prvkem, konstanta, naše volba ⇔ výkon
  - RLS využívá mnohem více informace o signálu než LMS, korelace vs. Výkon
  - Geometrický význam násobení vektoru maticí
    - RLS dojde rychleji k minimu

- RLS pro exponenc. Zapomínání
- 1/λ = zapomínání, 0 << λ < 1

$$\underline{P}_n = \frac{1}{\lambda}\left(\underline{P}_{n-1} - \frac{\underline{P}_{n-1}\underline{x}_n\underline{x}_n^T\underline{P}_{n-1}}{\lambda + \underline{x}_n^T\underline{P}_{n-1}\underline{x}_n}\right)$$

- RLS (FIR, řádu M), estimace ... (rekurzivní algoritmus nejmenších čtverců)
  - Počáteční podmínky, w0, P0, λ
  - Cyklus pro n = 1,2,...



4. Konvergenční chování adaptivních algoritmů – vysvětlit pojmy: rychlost konvergence, chyba v ustáleném stavu (misadjustment), porovnání algoritmů: LMS, RLS, NLMS, SLMS

- RLS se učí rychleji
- Chybový povrch (ES) = střední kvadr. Hodnota chyby estimace
  - $\boldsymbol{J} = E[e^2[n]] = R_d[0] + w^T R_x w - 2R_{dx}^T w$
- Minimum ES: $J_{min} = R_d[0] - R_{dx}^R w$
- Alternativní tvar ES: $J = J_{min} + (w - w^*)^T R_x (w - w^*)$ ... w* je vektor poskytující minimum ES
- w* je optimální

- Nová váha LMS je dána jako stará váha + oprava, oprava je daná minus okamžitým gradientem J
- Analýza LMS:
  - Diferenční rovnice, která řídí algoritmus
  - Diferenční rovnice vzhledem k váze:

$$w_0[n+1] = w_0[n] - 2\mu R_x[0](w_0[n] - w_0^*)$$

  - Řešení > postupné dosazování (rekurentní řešení)

$$w_0[n] = w_0^* + (1 - 2\mu R_x[0])^n (w_0[0] - w_0^*)$$

  - Konvergence závisí na hodnotě první závorky a jak daleko odejdeme počáteční podmínkou od optimálního řešení (druhá závorka)

- Analýza řešení Diferenční rovnice:
  - Pro n -> ∞ chceme aby: $w_0[n] \to w_0^*$ ... první závorka musí konvergovat k 1
  - Za této podmínky řešení dif. Rce konverguje: $1/R_x[0] > \mu > 0$
  - Pokud je $1-2\mu Rx[0] > 0$ ... => $0 < \mu < 1/2Rx[0]$
    - = > řešení Rovnice (w0[n]) nemění znaménko = nadkritické tlumení
  - Pokud je $1-2\mu Rx[0] = 0$ ... => $< \mu = 1/2Rx[0]$
    - => Řešení nalezeno v 1. iteraci ... w0[n] = w0* ... = kritické tlumení
  - Pokud je $1-2\mu Rx[0] < 0$ ... => $1/2Rx[0] < \mu < 1/Rx[0]$
    - => řešení osciluje ... = podkritické tlumení
  - $\mu < 0$ nebo $\mu >= 1/Rx[0]$ ... = diverguje

- RLS je úprava alg LMS tak, aby byl kriticky tlumený
- Časová konst. LMS
    - Jak rychle alg konverguje
    - Hledáme takové n=n0, aby w0n = e^-1
    - n_0 = -1/2μRx[0] = -1/λ
    - pro λ->1 ... n0 ≈ 1/1- λ
        - závislé na výkonu signálu Rx[0] ... to nechceme... proto NLMS

- NLMS = normované LMS
    - μ = α/Rx[0] ... n0 ≈ 1/2α
    - volba alfa: 0 < α < ½ … 0 < μ < 1/2μRx[0]
    - rovnice NLMS: $w_k[n+1] = w_k[n] + \frac{\alpha}{\varepsilon + \hat{R}_x[0]} e[n]x[n-k], k = 0,1, \dots, \varepsilon \to 0$
    - ε -> 0 … zabránění divergence, zabránění dělení 0
    - Odhad R^x[0]:
        - blokové, dávkové (nestranný a konzistentní odhad)
        - nebo průběžný odhad (rychlejší reakce, velká lambda), častější
    - optimální hodnota: $\alpha = \frac{1-\lambda_I}{2}$, volíme λ_I

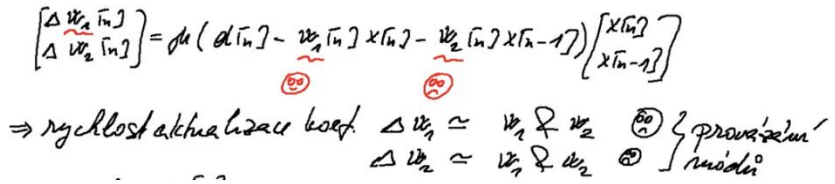
- Misadjustment
    - Chyba v určení koeficientu
    - Značení: „kaligrafické M“
    - LMS: $M = \mu x^T x = (M+1)R_x[0]$ ... M = řád filtru
    - NLMS: $M = (M+1)\alpha$

- SLMS
    - V komunikacích pro odstranění přeslechů v uzlech
    - Chyba estimace e[n] nahrazena sgn(e[n]), popřípadě x[n] nahrazeno sgn(x[n]), nebo nahrazeno obojí zároveň
    - Má větší chybu než LMS
        - Signál v amplitudě < 1, tak e také < 1
    - Velmi nelineární, závisí na vstupní amplitudě
    - Velká chyba v ustáleném stavu -> volíme malé μ -> pomalá konvergence

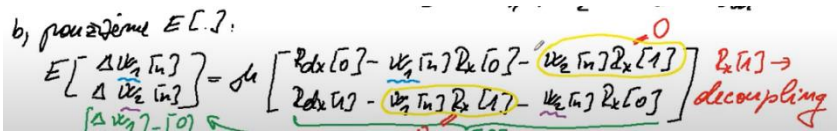
- LMS vhodné pro stacionární signály, s konst. Výkonem
- NLMS – vhodne pro signály s proměnným výkonem
- SLMS – menší výpočetní nároky (jsou celkem 3 typy), větší vhyba v ustáleném stavu

5. Využití ortogonálních transformací pro adaptivní filtraci pro urychlení konvergence LMS algoritmu: jaké ortogonální transformace lze použít a proč urychlují konvergenci

- LMS s ortogonalizací
- Mód konvergence – jak rychle konverguje váha
- Coupling = provázání konvergenčních módů
    - Změna jednoho koeficientu je vázana na změnu i druhého
    - Rychlost aktualizace koeficientů (konvergenční rychlost) $\Delta w[n+1] = \Delta w[n](\boldsymbol{I}-\mu Rx)$



- Nejpomalejší mód rozhoduje o rychlosti konvergence
- Decoupling = rozvázání módů
    - Pokud požadujeme rozvázání módů, tak Rx[1]=0

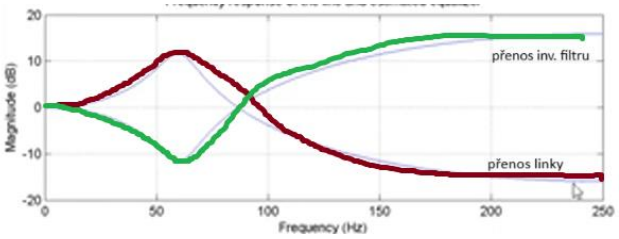


    - Rozvázání módů vyžaduje diagonální matici Rx ⇔ vstup je bílý šum (má diag. Matici)
    - => W-H rovnice ... Rw=rdx
    - Budíme-li RLS a LMS bílým šumem, konvergují stejně rychle
        - Bílý šum rozváže konvergenční módy
    - $\Delta w'[n]=\Delta w'[n](\boldsymbol{I}-\mu D)$
        - D = diagonální matice vlastních čísel daná vztahem $Rx=VDV^T$
        - Matice V reprezentuje ortogonální transformaci (PCA)
            - Matice V vlastních vektorů

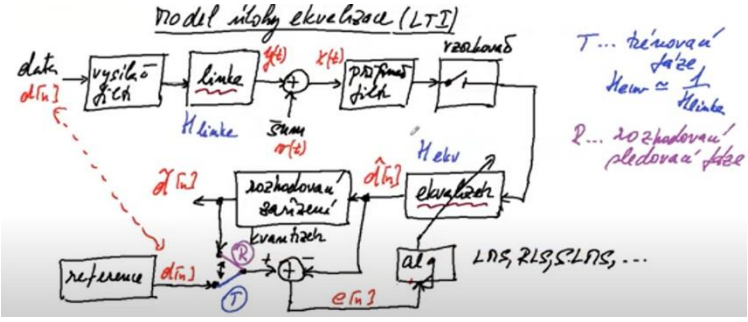
- Ortogonální transformace
    - Barevný šum > bílý šum
    - PCA: optimální ⇔ signálově závislá
    - DFT: signálově nezávislá -> vyžaduje komplexní LMS (výstup DFT jsou komplex. Čísla)
    - DCT: signálově nezávislá -> stačí reálné LMS, lepší kompresní vlastnosti než DFT
    - WHT: sig. Nezáv. -> báze z ortogonálních obdélníků
    - Banka filtrů
    - Vlnková transformace
- Máme provázané módy, kdžž vybělíme signál, tak je rozvážeme, signál umíme vybělit ortogonální transformací
- RLS v podstatě pracuje jako alg. S ortogonalizací, je tam „schované“ PCA

6. **Vysvětlit princip některých aplikací AF – estimace (odhad) frekvence, identifikace LTV systému, ekvalizace přenosové linky, dekorelace a separace signálů, určení směru příchodu signálu (beamforming), jaké jsou podmínky správného použití?**

- Ekvalizace
  - Snažíme se udělat takovou opravu frekvenční charaketristiky přenosové linky, aby to co se přenáší (digi data), aby byla bez jakéhokoliv zkreslení
  - „úloha inverzní filtrace“
  - v podstatě slepá dekonvoluce = cíl je získat vstupní signál ze snalosti výstupu
  - inverzní filtr by měl mít přesně inverzní přenos oproti přenosu linky

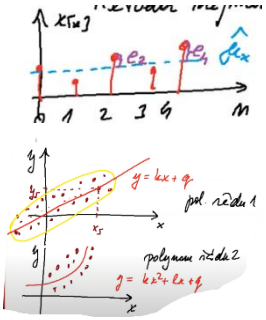


- Problém nalezení inverzního filtru
  - Ekvalizér musí být stabilní, mít póly uvnitř jednotkové kružnice
- Ekvalizér – za úkol najít inverzní funkci
  - pomocí LMS neumíme zajistit úlohu slepé dekonvoluce
  - rozhodovací zařízení (kvantizér): potřebný ke správnému hledání inv fce, často funkce sgn
  - chybový signál pro řízení LMS alg je získám rodílem mezi výstupem LTV FIR a výstupem kvantizéru
  - alg pro sledovací a rozhodovací fázi se nemění, mění se pouze vstup, pro T=referenční signál, pro R=výstup kvantizéru
  - ekvalizér řízen alg.
  - FIR-LMS 2 fáze
  - Trénovací fáze (T)– snaží se nastavit ekvalizér tak, aby získal inverzní přenos k lince
  - Rozhodovací fáze (R) – jakmile naučíme ekvalizér na přenosovou linku, přepnem do R , ekvalizér provede pomocí LMS slepou dekonvoluci



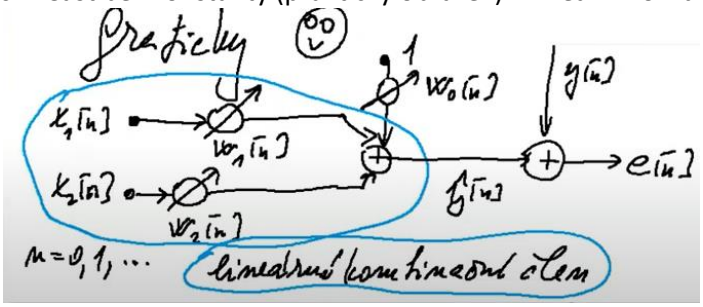
  - LTV linka -> podmínka činnosti: malá chybovost ~ 1%
  - Používá se SLMS, šetří operace
- algoritmy pro slepou ekvalizaci
  - Nepotřebují trénovací fázi
  - Pracují s nelinární funkcí
  - Goddadrdův alg (CMA):

7. Princip metody nejmenších čtverců (LS) – účelová funkce, princip ortogonality, normální rovnice, ilustrovat na příkladu nalezení parametrů přímky pro naměřená data

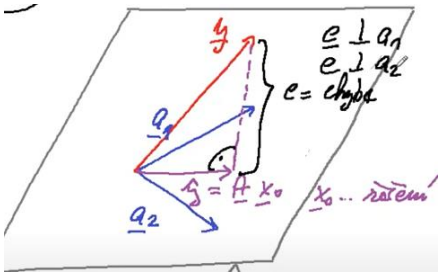


- Proložení data konstantou, tak aby rozdíl mezi daty byl co nejmenší
- aproximace dat polynomem
  - Diskrétní body nahrazujeme spojitou funkcí (aproximující funkce dat)
  - Prokládáme polynomem řádu M
- Úloha aproximace s více proměnnými
  - Body v N-dim prostoru prokládáme nadplochou

- <u>Účelová funkce</u>
  - Měří odchylky mezi pozorovanými daty a hodnotami, které model předpovídá
  - Gauss = L2 norma: $J = \sum_{n=0}^{N-1} e^2[n] = \sum(y[n] + w_0 + w_1 x_1[n] + w_2 x_2[n])^2$
    - $\hat{\mu}_x = \frac{1}{N}\sum_{n=0}^{N-1} x[n]$ ... mean
  - Laplac = L1 norma: $J = \sum_{n=0}^{N-1} |e[n]|$
  - Část bez konstanty (praktický obrázek) – lineární kombinační člen



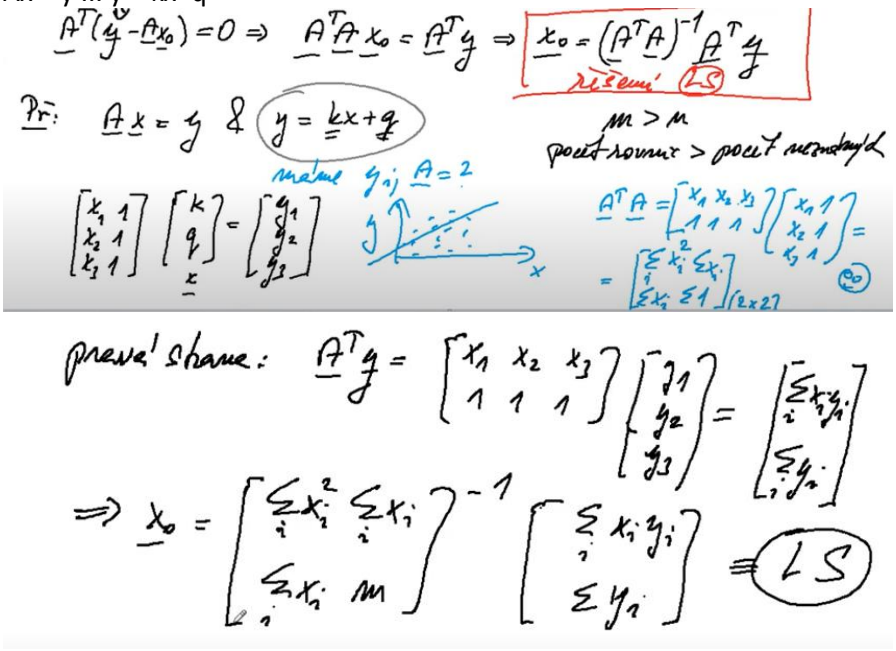
- Soustava lin. Rovnice
  - Ax=y ... data=y_i, A∈R^{mxn} známá, cíl určit x
    - m > n ... preurčená soustava rovnic -> LS ⇔ L2
- y nemusí ležet v rovině generované vektory a1,2 > nemá řešení > promítneme y do roviny generované a1,2 > získáme y^ = Ax0 ... x0 = řešení
- y^ = průmět y do roviny a1,2



- <u>princip ortogonality</u>
  - $A^T e = 0$
  - chyba e je kolmá ke všem vektorům, které leží v rovině (e_|_ a1, e_|_ a2)
  - řešení x0 -> Ax0=y^
  - e = y-y^
  - LS: $\min ||e||^2 = \min ||y - Ax_0||^2 = \min(e^T e) = \min\left((y^T - x_0^T A^T)e\right) =$
    $= y^T e - x_0^T A^T e = y^T e$ ... minimální hodnota ||e||²
  - Díky tomuto principu dokážeme zjistit, jak velké chyby se dopouštíme (její energie)

- <u>Normální rovnice</u>
  - $A^T e = 0 > A^T(y - Ax_0) = 0 > A^T A x_0 = A^T y => x_0 = (A^T A)^{-1} A^T y$: řešení LS
  - Platí za podmínky, že počet rovnice > počet neznámých ... m > n

- Aproximace přímkou
- Ax = y ... y = kx+q



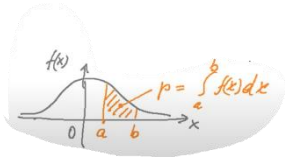
- $\sum x_i y_i$... vzájemná korelace mezi daty x,y

- Vliv šumu
  - Změněná data se šumem: $y_i = y_{i\_opt} + y_i$(=šum)
  - Je A singulární? (det(A) = 0)
  - Šum + špatně podmíněná matice A (je singuláren) -> velké chyby

8. Princip metody Monte Carlo a její použití (např. pro integraci funkce jedné reálné proměnné), princip bayesovské filtrace a skrytých markovovských modelů a jejich vztah ke Kalmánově filtraci

- Historie: simulace fyz. Experimentů, vytvíření demografického modelu USA (1950-60)
- Nyní: ekonomie, řízení tel. Centrál, řízení dopravy, hromadná obsluha, vyhodnocení metod,...
- opakujeme stejný experiment, ale za jiných podmínek (s jiným signálem)

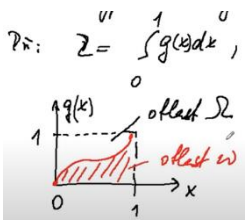
- Matematika:
- ξ = náhodná veličina ϵ Ω
- Rozdělení: P (a < ξ <= b) = p (pravděpodobnost)
- . ξ padn do int (a,b), f(x) = hustota rozdělení pravděpodobnosti ξ
- Předpokládáme, že lze realizovat pokus, jehož výsledek je hodnota x_i
  - Hodnota náhodné veličiny ξ
  - X_i - ϵ (a,b) => zdařilý pokus
  - Počet zdařilých pokusůůů je n, počet pokusů je m, m < n
  - Relativní četnost p‘ jevu, že ξ ϵ (a,b) ... p‘ = m/n



- Bernouliho věta: $lim_{n\to\infty} = P\left(\left|\frac{m}{n} - p\right| < \varepsilon\right) = 1$
  - Při ∞ velkém množtvím pokusů, není rozdíl mezi teoretickou p a relativní četností
  - Při dostatečně velkém počtu nezávislých pokusů se rel četnost jevu liší libovolně málo od pravděpodobnosti p

- <u>Řešení metodou MC</u>
  - Modelování pokusů pomocí náhodných čísel
  - Alg: i = 1,..n ... počet pokusů, m = 0 (poč. Zdařilých pokusů)
    - Vybereme xi o rozdělení f(x) (častá volba rovnoměrné rozdělení, jinak nutné použít transformaci náhodných veličin)
    - Provedeme pokus: h = 1 (zdařilý), = 0 .... m = m+h
    - Konec cyklu, další i
  - P = m/n

- <u>Použití MC pro integraci funkce</u>
  - podmínka/omezení: 0 <= g(x) <= 1
  - Můžeme použít pouze tehdy, pokud Integrand bude splňovat podmínku, že se jedná o pravděpodobnost
    - g(x) >= 0
    - když integrujeme přes definiční obor (+-∞) získáme =1 (úplná p)
  - snažíme se vyčíslit plochu, kterou uzavírá křivka proti ose x
  - generace náhodných čísel, budeme je umisťovat do roviny a zjišťovat, která čísla padnou do červené oblasti
  - <u>implementace MC:</u>
    - generujeme xi a yi (rand) … rouřadnice bodu v Ω
    - [xi,yi] ϵ ω
      - yi < g(xi) -> h -> m = m+h
    - konec cyklu pro i
    - integrál je =~ m/n ... pro dostatečně velké n



- přesnost výpočtu – chcili zjistit chybu -> musím výpočet opakovat
  - lze určit var, med, kvantily, histogram,... výsledků integrálů

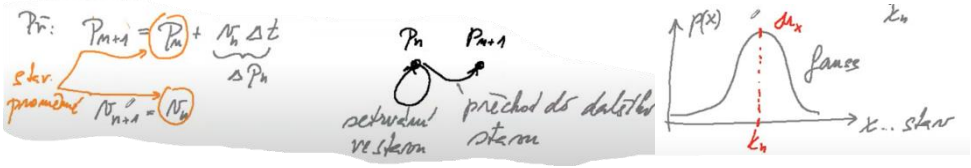
- řešení lineárních rovnic: Markovovy řetězce
- baysovská fitlrace u KF

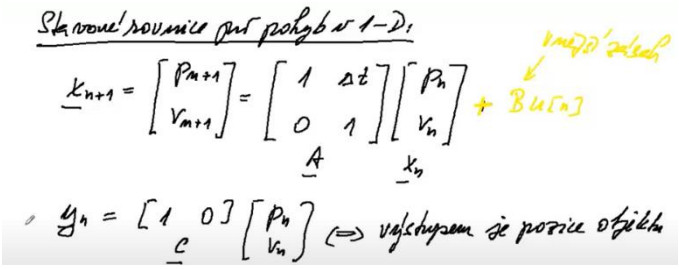
9. Kalmánova filtrace (KF) – předpoklady pro její použití, účelová funkce (kritérium), princip KF – bloková struktura algoritmu, jednoduché příklady použití; (případně porovnání KF s Wienerovou filtrací)

- Wienerův filtr použití pro stac. Náhodné procesy (kritérium MMSE)
- KF pro nestacionární náhodné signály s LTV systémem
- Použití: odhad stavu systémů
- Kritérium: iterační řešení, MMSE (J = E[||e[n]||^2])
- Výstup KF: parametry systému, signály, odhad stavu
    - Stav. Systém: napětí, proud uvnitř obvodu (stav veličiny), pozice systému (kyvadlo,..)
    - KF odhaduje hodnoty stavového vektoru

- KF je iterativní výpočet kombinující predikci a korekci
    - Predikce je založena na stavovém popisu a přechodu mezi stavy (stav. Popis)
    - Model Stavu: $x_{n+1} = x_n + w[n]$ ... w[n] = šum (nepřesnost), stav je skrytý (KF odhaduje)
        - Ví kde jsme, ale neví kde budeme, to odhaduje pomocí modelu
        - Predikce stavu je opravena (korigována) získaným měřením
    - Stav x_n -> predikce -> stav $x^{\wedge}_{n+1}$ -> měření -> korekce -> zpět do stavu x_n
        - Predikce bez měření, korekce na základě měření
    - Model měření: $y_n = x_{n+1} + v_n$ ... wn a vn jsou bílé šumy (modely nepřesností)

- Příklad stavového popisu pro pohyb systému v 1D
    - diskrétní popis: $x_{n+1} = Ax_n + Bu_n$, x_n = stav, u_n = vstup
      $y_n = Cx_n + Du_n$, y_n = výstup, A,B,C = stavové rovnice
    - buď setrváváme ve stavu (Pn -> Pn), nebo přejdeme do dalšího (Pn -> P_{n+1})

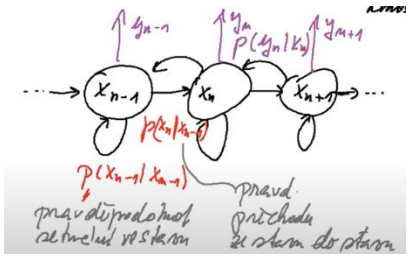




    - stav (např. Pozice Pn) budeme chápat jako pravděpodobnost

- snažíme se najít max, nejvíce pravděpodobný jev
- nepřesnosti odhadu stavu a nepřesnost měření modelujeme pomocí šumů

Markovův řetězec



- Staový automat popsaný pravděpodobnostmi
- Podmínka: P(xn|xn-1) = P(xn|x0,x1,..,xn-1)
  - Projde celou minulost

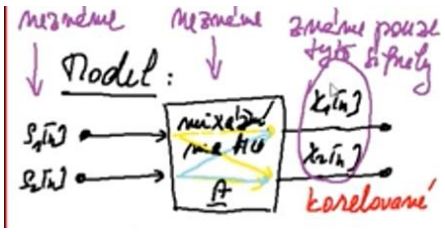
- Bayeovský vztah s Markovem
  - Bayesovská filtrace:

$$p(x_n|y_n) = \int_{x_{n-1}} \frac{p(x_n|x_{n-1})p(y_n|x_n)}{p(y_n|y_{n-1})} p(x_{n-1}|y_{n-1})$$

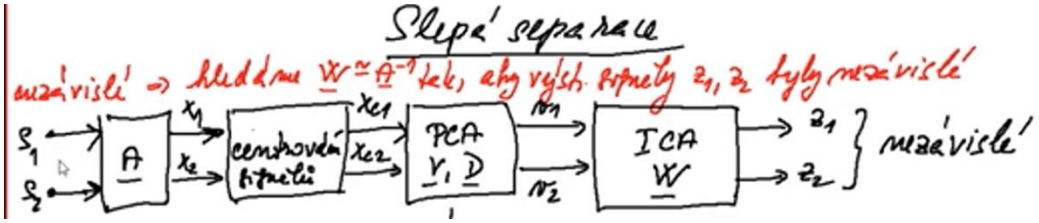
  - P = pravděpodobnost nebo pdf
  - Cílem je určit vztah na základě měření, určit posterior
  - Řešení 2 kroky:
    - Predikce$(x_n) = \int p(x_n|x_{n-1})p(x_{n-1}|y_{n-1})$
    - Korekce$(x_n) = p(x_n|y_n) = \eta p(y_n|x_n)pred(x_n)$
  - KF: pdf = Gauss, šumy = Gauss, systém je lineární
    - Za těchto předpokladů lze místo pdf použít parametry normálního rozdělení

- Diskrétní úloha Bayesovské filtrace:
  - Predikce: pred(xn) = sum(„přechod“ . kor(xn-1)
    - „přechod“ = Markov (řetězec)
  - Korekce: kor(xn) = „likelyhood“ . pred(xn)
    - „likelyhood“ měření
- Diskrétní KF
  - stavový popis: xn = An-1xn-1 + wn
  - měření: yn = Cnxn + vn
  - inicializace, počáteční podmínky, stav v čase 0
  - kovariační matice chyb měření
  - cyklus:
    - predikce stavu s korekcí
    - odhad chyby
    - Výpočet Kalmánova zisku
    - Odhad stavu
    - Odhad chyby
- Použití: GPS
- Předpoklad: LTI/LTV a Gaussovské šumy

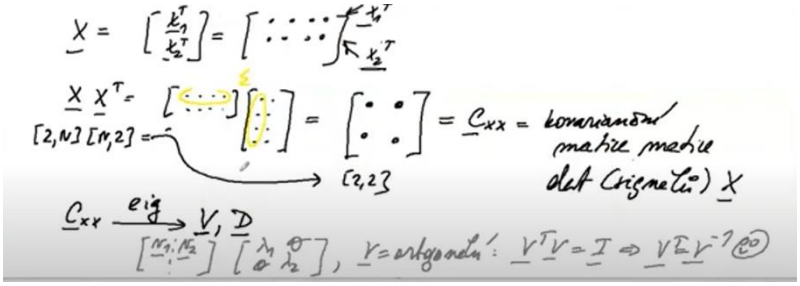
10. Dekorelace vícerozměrných signálů (PCA) – model vzniku signálu, podmínky použití metody, kdy lze pomocí PCA dosáhnout úplné separace signálů?



- PCA: rozklad na hlavní komponenty, rozklad kovarianční matice signálů
  - Máme dva nezávislé signály, ty smícháme (mixážní matice), známe pouze výstupní signály
  - Rotací souřadné soustavy provedeme dekorelaci
  - PCA pracuje s ortogonalní transformací
  - Musíme centrovat signály



- PCA diagonalizuje kovarianční matici dat $Cx = XX^T = VDV^T$ ... (X = matice dat)
  - D diagonální matice vlastních čísel, $D = V^TCxV$
  - V ortogonální matice vlastních vektorů

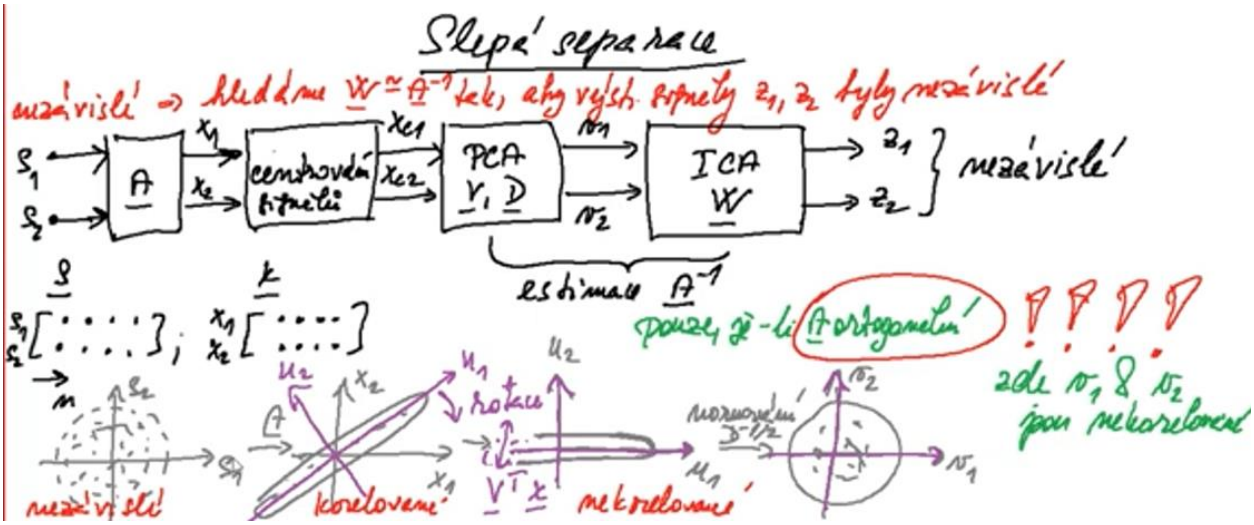


  - Dekorelaci dat provádí transformace $Y = V^TX$ (odrotování souřadné soustavy)
    - Y je matice nekorelovaných dat
  - Po dosazení za Y a úpravě získáme: $C_Y = V^TC_XV = D$
  - C_Y je diagonální -> Y obsahuje dekorelovaná data (Y je matice v1,v2 v obrázku)
- Normujeme (vlastní čísla matice D představují rozptyl)
  - Přenásobíme V^TX s D^{-1}, zbavíme se tím rozptylu, který tam původně nebyl
  - Pokud bude A ortogonální, samotné PCA stačí k dekorelaci
    - Pokud nebude ortogonální, nedotočíme elipsu do „nulových“ os

- Předpoklady: lineární kombinace $x_i(t)$ (skalární mixáž
  - Nezávislých signálů $s_i(t)$ ... mixážní matice $x(t) = As(t)$
- Řešení může mít jiné pořadí původních signálů
- Řešení může mít opačnou polaritu
- Řešení nezachovává měřítko

- PCA provádí dekorelaci Cxx (kovarianční matice = centrovaná)
- Cxx je odhad s chybou -> další zpracování může zesílit chyby
- PCA hledá směry největšího rozptylu dat (směry jsou určeny vlastními vektory)

11. Separace vícerozměrných signálů (ICA) – model vzniku signálu, podmínky použití, význam použití charakteristik vyšších řádů (např. špičatost); nakreslit kompletní blokové schéma metody separace signálů a vysvětlit význam dílčích bloků.



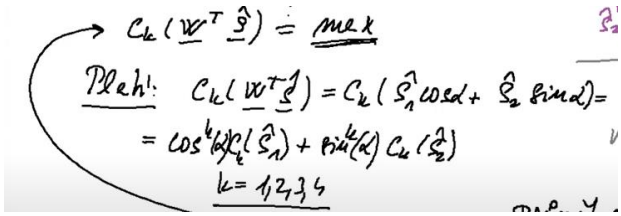
- FastICA využívá špičatost – signály budou negaussovské
- Iterční algoritmus
- Logaritmus charakteristické funkce F(ω) -> rozvoj do řady -> koeficienty této řady = kumulanty
- FastICA využívá kumulanty: (součet kumulantů náhodných veličin)

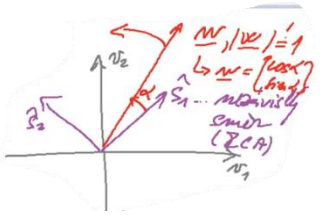
$$x + y \triangleq f(x) * g(y) \triangleq c_{fk} + c_{gk}, \quad k = 0,1,2,3,4,\dots$$

součet kumulantů něk. gram. x a y

- Kumulant je vyjádření nějakého momentu (1. stř hodnota, 2. rozptyl,..)
- 4. kumulant je špičatost: $c_4 = m_4 - 3m_3^2$
- Špičatost pomáhá rozlišit typy rozdělení

- Aplikace kumulantů:
  - Sledujeme modul vektoru w = 1

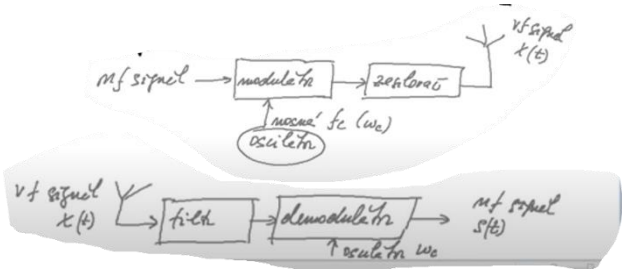




  - Nalézt směr w tak, aby $c_4(w.v)$ byl maximální (maximum špičatosti)
    - Podmínka ||w|| = 1

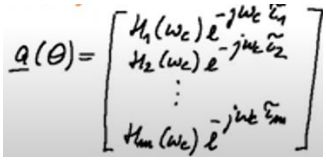
- PCA normuje obě osy > stejné rozptyly obou os (c2 = 1... ve všech směrech stejná hodnota)
- 4. mocnina > vidíme kytičku > pokud je pootočená (měli jsme neortogonální A), je zde nějaká špičatost > ICA vidí pootočení > sesouhlasí směry do os ve 4.mocnině

12. Tvarování přijímací charakteristiky pole senzorů (beamforming) - pojmy: pole senzorů – načrtnout a popsat zvláštní případ lineárního pole (ULA); prostorový vzorkovací teorém. Principiální rozdíl mezi konvenčním (neadaptivním) systémem pro směrový příjem - např. DAS (Delay and Sum Beamformer) a adaptivním systémem pro směrový příjem - např. Caponova metoda (MVDR - Minimum Variance Distortion-less Response Beamformer).



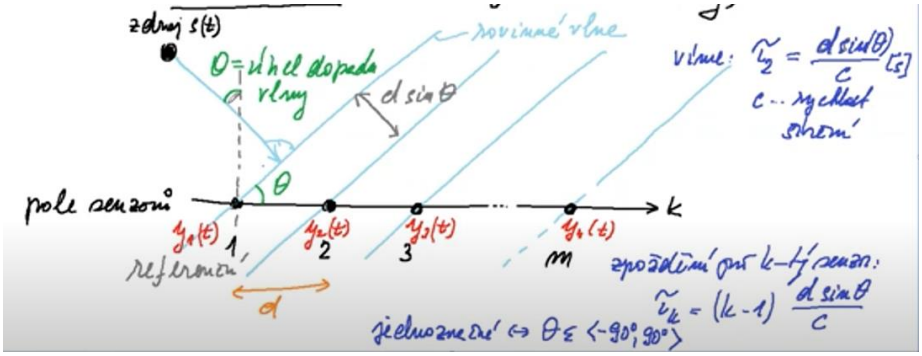
- Určení směru dopadu vlny
- Motivace: příjem EM vlnění a akustika
- Potlačení rušení, které přichází z jiných směrů
- Přenosový řetězec: vysílač > přijímač
- Cíl: získat odhad směru příchodu signálu

- <u>model signálu:</u>
- **a(θ)** (vektor)... θ = úhel dopadu vlnoplochy na senzor
  - ➢ vlastnosti pole senzorů



- 1-n senzorů s přenosy H a různými zpožděními, τ_i = zpoždění signálu x = s na senzorech
- Signál na výstupu demodulátoru je ovlivněn zpožděním a přenosem senzorů, signál který dopadá na senzory a šum
  - Diskrétní model: y[n] = a(θ)s[n] + e[n] … (y = data, s = signal, e = šum)
- Model Maticově pro 1 zdroj: Y = a(θ)s + e ... známe Y, a ... neznáme s,e
- Model pro Více zdrojů: Y = As + e ... (A co sloupec to jeden zdroj $a(\theta)_{1,2,..}$)

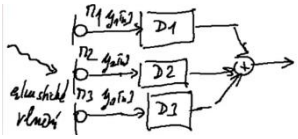
- <u>Pole senzorů ULA</u>
- předpoklady
  - Senzory mají stejné vlastnosti – všesměrové stejné senzory: jejich zisk je konstantní
  - Na senzory dopadá rovinná vlna
  - První senzor je referenční
  - Senzory jsou rozmístěny ekvidistantně na přímce ve vzdálenosti *d*



- ULA => a(θ) = [1 (referenční), $e^{-j\omega_c \tau_2}$, …) … předp: $|H_k(\omega_c)| = 1$
  - > Báze DTFT

- <u>Prostorový vzorkovací teorém</u>
  - Vzdálenost mezi senzory musí být menší než polovina vlnové délky: $d < \lambda/2$

- DAS
    - Zpoždění, parametry D1-Dn <-> odhad zpoždění -> např korelace nebo PSD
        - Max Korelace/PSD odpovídá zpoždění
    - V závislosti na zpoždění, preferujeme nějaký směr pohledu
    - Nevýhoda: frekvenčně závislý prostorový přenos
    - Na obrázku širokopásmový DAS
    - Nevyužívá vlastnosti signálu
    - Prostorové rozlišení je závislé na konfiguraci pole senzorů
    - Používá FT pro zjištění odkud jde signál
    - Vlastnosi jsou nezávislé na datech – je neadaptivní



$$\max \underline{a}^H \underline{R}_y \underline{a} \rightarrow \sum_n |DFT(y(n))|^2 = \sum_n |\underline{a}^H \underline{y}(n)|^2$$

- Uděláme Fourierku datové rovnice a hledáme maximum

- MVDR (Capon)
- Adaptivní
- Umí využít vlastnosti signálu
- Dovoluje rozlišit signály z velmi blízkých směrů
- Počítáme skalární součin vektorů *a* vážených inverzní autokorelační maticí