Adaptivna obdelava signalov v telekomunikacijah

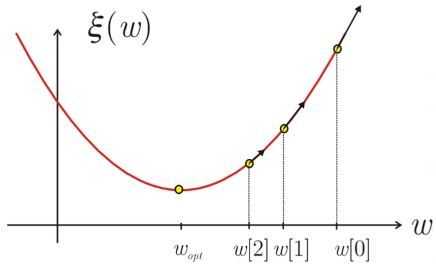
5. del

Pregled tem:

- gradientni algoritem
  - hitrost adaptacije in presežek napake
- LMS algoritem
  - zgledi
  - analiza LMS algoritma
- predznačni algoritem
  - analiza predznačnega algoritma
  - predznačni algoritem na osnovi približka gradienta
- primerjava algoritmov

Gradientni algoritem

- Utež popravljamo v nasprotni smeri gradienta kriterijske funkcije:



$$\boxed{w[m+1] = w[m] - \mu\nabla[m]}$$

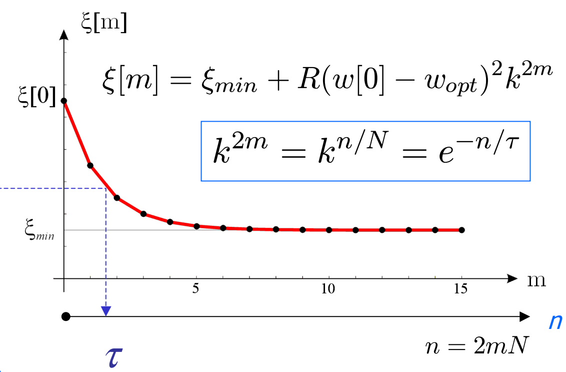
- Presežek napake je posledica perturbacije uteži in napak pri merjenju gradienta:

$$\boxed{M_p = \frac{\delta^2 R}{\xi_{min}}} \qquad \boxed{M_g = \frac{\overline{\xi - \xi_{min}}}{\xi_{min}} = \frac{\mu\xi_{min}}{4N\delta^2(1-\mu R)}}$$

Hitrost adaptacije

- Zanima nas število vzorcev n, pri katerem pade presežek srednje kvadratične napake za faktor e . Odgovor poiščemo iz krivulje učenja:

$$\frac{\xi[0]-\xi_{min}}{e}$$



$$\xi[m] = \xi_{min} + R(w[0]-w_{opt})^2 k^{2m}$$

$$\boxed{k^{2m} = k^{n/N} = e^{-n/\tau}}$$

- časovna konstanta τ :

$$\boxed{\tau = -\frac{N}{\ln k} \approx -\frac{N}{k-1} = \frac{N}{\mu R}}$$

Povečanje ξ zaradi perturbacije in napake gradienta

1. zaradi namerne perturbacije uteži pri meritvi gradienta $$M_p = \frac{\delta^2 R}{\xi_{min}}$$
2. zaradi napake pri meritvi gradienta $$M_g = \frac{\xi_{min}}{4\delta^2 \tau R}$$

$$\boxed{\xi_0 = (1 + M_p + M_g)\, \xi_{min}}$$

- neprilagojenosti sta obratnosorazmerni: $$\boxed{M_g = \frac{1}{4\tau M_p}}$$
- želimo zmanjšati oba vpliva hkrati:

$$\boxed{\xi_{0min} = (1 + \frac{1}{\sqrt{\tau}})\, \xi_{min}}$$

pogoj: $$M_g = M_p = \frac{1}{2\sqrt{\tau}}$$

Povezava minimalne napake s hitrostjo adaptacije

- srednja kvadratična napaka se v ravnotežnem stanju poveča za najmanj:

$$\boxed{\xi_{0min} = (1 + \frac{1}{\sqrt{\tau}})\, \xi_{min}}$$

Ne moremo imeti hkrati majhnega povečanja napake in hitre adaptacije uteži:

- hitra adaptacija pomeni manjhen τ in s tem večji ξ_{0min}
- počasna adaptacija pomeni velik τ in s tem manjši ξ_{0min}

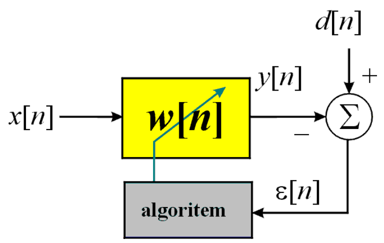
Rešitev problema je v spreminjanju adaptacijske konstante :

- na začetku izberemo večjo vrednost adaptacijske konstante μ
- ko se približujemo optimalni vrednosti uteži, zmanjšujemo μ

$$\boxed{\tau = -\frac{N}{\ln k} \approx -\frac{N}{k-1} = \frac{N}{\mu R}}$$

LMS algoritem

- Gradientni algoritem: $\boxed{w[m+1] = w[m] - \mu\nabla[m]}$
- LMS algoritem: namesto natančnega gradienta kvadrata napake uporabimo preprost približek na osnovi ene same meritve!



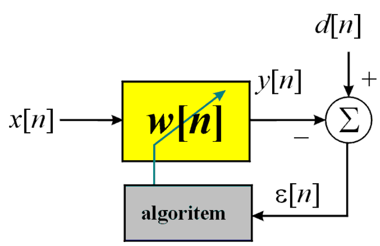
$$\boxed{\hat{\nabla}[n] = \frac{\partial \varepsilon^2[n]}{\partial w[n]}}$$

- LMS (Least Mean Square) algoritem:

$$\boxed{w[n+1] = w[n] + 2\,\mu\,\varepsilon[n]x[n]}$$

LMS algoritem - zgled 1

- Signal na vhodu in željeni signal sta konstantna: x=5, d=10



- LMS algoritem: $\boxed{w[n+1] = w[n] + 2\,\mu\,\varepsilon[n]x[n]}$
- Izberemo utež μ : $\boxed{0 < \mu < \frac{1}{R}}$

μ=1/100

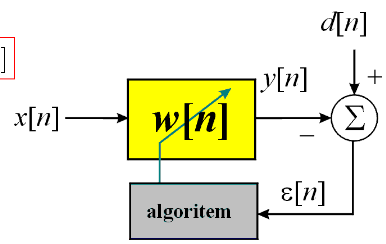
LMS algoritem - zgled 2

- Podana sta signal na vhodu in želeni signal:

$$\boxed{x[n] = A\ \cos[n\Omega_0]}$$

$$\Omega_0 = \frac{\pi}{6}$$

$$A = 1$$



$$\boxed{d[n] = B\ \cos[n\Omega_0 + \Phi]}$$

$$\Phi = \frac{2\pi}{6}$$

$$B = 1$$

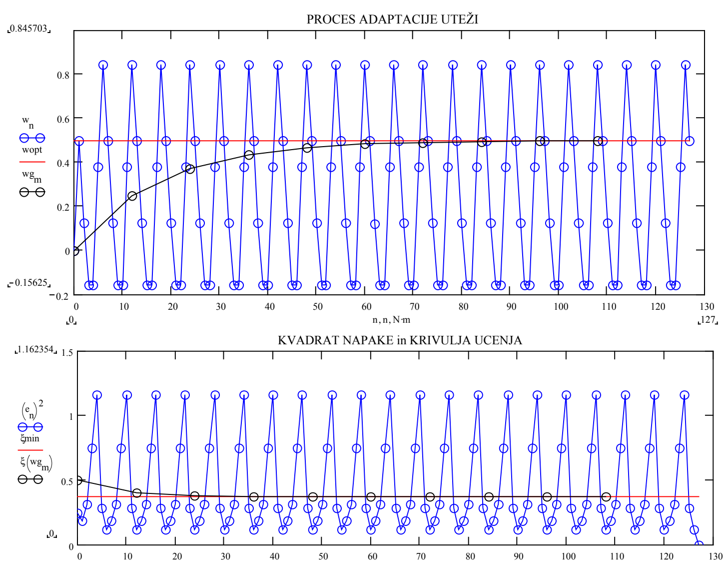
- LMS algoritem: $\boxed{w[n+1] = w[n] + 2\ \mu\ \varepsilon[n]x[n]}$

- Kako izberemo utež μ ? $\boxed{0 < \mu < \frac{1}{R}}$

μ=0.5 μ0 ?

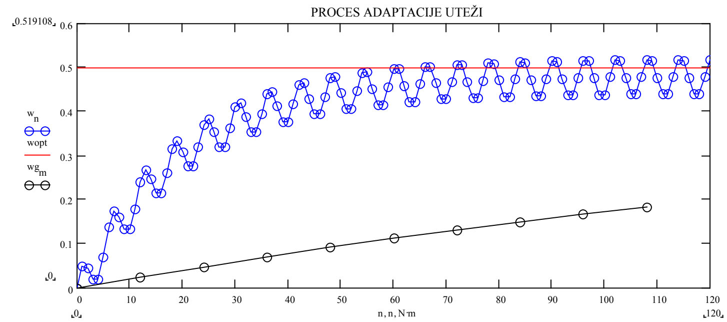
LMS algoritem - zgled 2-a

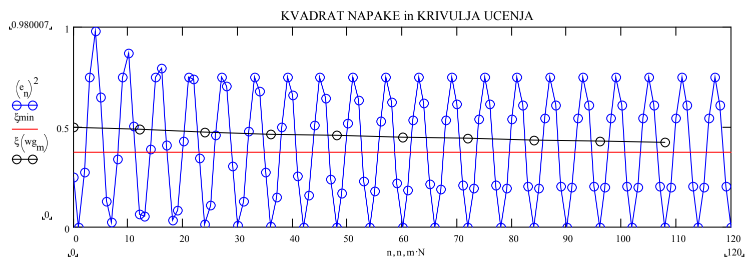
- **μ= 0.5**



LMS algoritem - zgled 2-b

- **μ= 0.05**



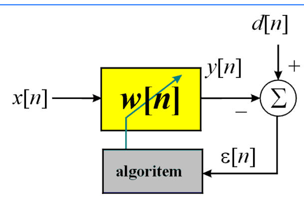


Analiza napak pri LMS algoritmu

- varianca gradienta v ravnotežnem stanju: $\overline{\eta^2[n]} = \overline{\hat{\nabla}^2[n]} = 4\ \overline{\varepsilon^2[n]x^2[n]}$ $\boxed{= 4\ \xi_{min} R}$
- varianca uteži: $\mathrm{var}(w) = \dfrac{\mu\ \xi_{min}}{1-\mu R}$ $\boxed{\mathrm{var}(w) = \mu\ \xi_{min}}$
- presežek napake v ravnotežnem stanju: $\boxed{\overline{\xi - \xi_{min}} = \overline{v^2}\ R = \mu\ \xi_{min}\ R}$
- neprilagoditev: $\boxed{M = \dfrac{\overline{\xi - \xi_{min}}}{\xi_{min}} = \mu R}$
- približek časovne konstante τ : $\boxed{\tau = \dfrac{1}{2\ln(1-2\mu R)} \approx \dfrac{1}{4\mu R}}$
- neprilagoditev in hitrost adaptacije sta v obratnem sorazmerju: $\boxed{M = \dfrac{1}{4\ \tau}}$

Predznačni algoritem

- gradientni algoritem: $\boxed{w[m+1] = w[m] - \mu \nabla [m]}$
- namesto natančnega gradienta povprečja kvadrata napake upoštevamo samo predznak gradienta (**+** ali **-**)!



$$\boxed{w[m+1] = w[m] - \mu \text{sign}(\hat{\nabla}[m])}$$

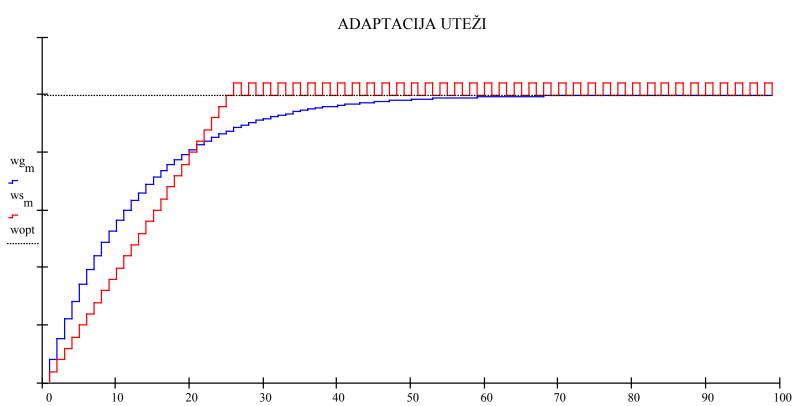
- predznak gradienta je odvisen samo od odmika od optimalne uteži:

$$\boxed{v[m+1] = v[m] - \mu \text{sign}(v[m])}$$

Proces adaptacije uteži

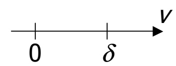
Zgled: x[n]=2 cos(n π/6), d[n]=2 cos(n π/6+π/3), μ=0.02

- gradientni algoritem
- predznačni algoritem



Analiza predznačnega algoritma

- predznačni algoritem: $\boxed{v[m+1] = v[m] - \mu \text{sign}(v[m])}$
- v najslabšem primeru je presežek napake: $\boxed{\overline{\xi - \xi_{min}} = \frac{\mu^2 R}{2}}$



- neprilagojenost v okolici optimalne uteži: $\boxed{M = \frac{\overline{\xi - \xi_{min}}}{\xi_{min}} = \frac{\mu^2 R}{2\xi_{min}}}$

Poenostavljeni SIGN algoritem

- predznačni algoritem: $\boxed{w[m+1] = w[m] - \mu \text{sign}(\hat{\nabla}[m])}$
- uporabimo približek gradienta: $\boxed{\hat{\nabla}[n] = -2\ \varepsilon[n] x[n]}$
- poenostavljeni SIGN algoritem:

$$\boxed{w[n+1] = w[n] + \mu\ \text{sign}(\varepsilon[n]) \text{sign}(x[n])}$$

Analiza SIGN algoritma:

- poenostavljeni SIGN algoritem: $$\boxed{w[n+1] = w[n] + \mu \ \mathrm{sign}(\varepsilon[n])\mathrm{sign}(x[n])}$$
- odstopanje od optimalne uteži: $$v[n+1] = v[n] + \mu \ \mathrm{sign}(\varepsilon[n]x[n])$$

$$\varepsilon[n] = \varepsilon_{min}[n] - v[n]x[n]$$

$$\boxed{v[n+1] = v[n] + \mu \ \mathrm{sign}(\varepsilon_{min}[n]x[n] - v[n]x^2[n])}$$

Napaka v okolici optimalne uteži je odvisna od minimalne napake:

- če je minimalna napaka enaka 0, se utež spreminja v okolici ***w_{opt}+/- μ***
- zaradi residualnega šuma nastopajo tudi popravki v napačno smer, število popravkov v pravi smeri pa narašča z odmikom od optimalne uteži !!

Primerjava algoritmov

Zgled: x[n]=2 cos(n π/6), d[n]=2 cos(n π/6+π/3), μ=0.02

- gradientni algoritem, SIGN algoritem,
- LMS algoritem, poenostavljeni SIGN algoritem

w_{opt}=0.5