7. FILTRE ADAPTIVE BAZATE PE METODA CELOR MAI MICI PĂTRATE

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Funcţia cost: eroare pătratică ponderată

$$J(n)=\sum_{i=1}^{n}\beta(n,i)|e(i)|^2$$

în care

$$e(i)=d(i)-y(i)=d(i)-\mathbf{w}^H(n)\mathbf{x}(i);$$
$$\mathbf{x}(i)=[x(i),x(i-1),\cdots,x(i-N+1)]^T$$

Factorul de pondere

$$\beta(n,i)=\lambda^{n-i},\quad i=1,\cdots,n,\quad 0<\lambda\le 1$$

$$\boxed{J(n)=\sum_{i=1}^{n}\lambda^{n-i}|e(i)|^2}$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

$$J(n)=\sum_{i=1}^{n}\lambda^{n-i}\left(d(i)-\mathbf{w}^H(n)\mathbf{x}(i)\right)\left(d^*(i)-\mathbf{x}^H(i)\mathbf{w}(n)\right)$$

$$J(n)=\sum_{i=1}^{n}\lambda^{n-i}\left|d(i)\right|^2-\mathbf{w}^H(n)\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)d^*(i)-$$

$$-\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}^H(i)d(i)\mathbf{w}(n)+\mathbf{w}^H(n)\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)\mathbf{x}^H(i)\mathbf{w}(n)$$

Notăm

$$\mathbf{\Phi}(n)=\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)\mathbf{x}^H(i)$$

$$\boldsymbol{\theta}(n)=\sum_{i=1}^{n}\lambda^{n-i}\mathbf{x}(i)d^*(i)$$

$$E_d(n)=\sum_{i=1}^{n}\lambda^{n-i}\left|d(i)\right|^2$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

$$\mathbf{\Phi}(n) = \sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}(i) \mathbf{x}^H(i)$$

$$\boldsymbol{\theta}(n) = \sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}(i) d^*(i)$$

$$E_d(n) = \sum_{i=1}^{n} \lambda^{n-i} \left| d(i) \right|^2$$

$\mathbf{\Phi}(n)$, $\boldsymbol{\theta}(n)$, $E_d(n)$ reprezintă estimatori deplasaţi ai matricei de autocorelaţie, vectorului corelaţiei între semnalul dorit şi secvenţa de intrare şi respectiv puterii semnalului dorit.

$$J(n) = E_d - \mathbf{w}^H(n)\boldsymbol{\theta}(n) - \boldsymbol{\theta}^H(n)\mathbf{w}(n) + \mathbf{w}^H(n)\mathbf{\Phi}(n)\mathbf{w}(n)$$

Anulând gradientul funcţiei cost se obţine ecuaţia normală.

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Ecuaţia normală. Minimizarea funcţiei cost se obţine tot pentru

$$\boxed{\mathbf{\Phi}(n)\mathbf{w}(n) = \boldsymbol{\theta}(n)}$$

Presupunând problema rezolvată pentru n-1, deci cunoscând:

$$\mathbf{w}(n-1) = \mathbf{\Phi}^{-1}(n-1)\boldsymbol{\theta}(n-1)$$

ne propunem să găsim o metodă pentru a evalua

$$\mathbf{w}(n) = \mathbf{\Phi}^{-1}(n)\boldsymbol{\theta}(n)$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Relaţii de recurenţă pentru $\mathbf{\Phi}(n)$ ***şi*** $\boldsymbol{\theta}(n)$:

$$\mathbf{\Phi}(n) = \lambda \sum_{i=1}^{n-1} \lambda^{n-1-i} \mathbf{x}(i)\mathbf{x}^H(i) + \mathbf{x}(n)\mathbf{x}^H(n)$$

$$\mathbf{\Phi}(n) = \lambda \mathbf{\Phi}(n-1) + \mathbf{x}(n)\mathbf{x}^H(n)$$

$$\boldsymbol{\theta}(n) = \lambda \sum_{i=1}^{n-1} \lambda^{n-1-i} \mathbf{x}(i) d^*(i) + \mathbf{x}(n) d^*(n)$$

$$\boldsymbol{\theta}(n) = \lambda \boldsymbol{\theta}(n-1) + \mathbf{x}(n) d^*(n)$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Este necesară o metodă de exprimare a inversei matricei $\mathbf{\Phi}(n)$, pornind de la relaţia de recurenţă.

Lema inversării matricei: Fiind date matricele $\mathbf{A}(N \times N)$, $\mathbf{B}(N \times N)$, $\mathbf{C}(N \times L)$, $\mathbf{D}(L \times L)$, în care $\mathbf{A}$, $\mathbf{B}$, $\mathbf{D}$, sunt nesingulare şi satisfac relaţia:

$$\mathbf{A} = \mathbf{B} + \mathbf{CDC}^H$$

inversa matricei $\mathbf{A}$ este dată de

$$\mathbf{A}^{-1} = \mathbf{B}^{-1} - \mathbf{B}^{-1}\mathbf{C}\left(\mathbf{D}^{-1} + \mathbf{C}^H\mathbf{B}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^H\mathbf{B}^{-1}$$

Vom aplica această lemă pentru:

$$\mathbf{A} = \mathbf{\Phi}(n), \quad \mathbf{B} = \lambda\mathbf{\Phi}(n-1), \quad \mathbf{C} = \mathbf{x}(n), \quad \mathbf{D} = \mathbf{I} = 1$$

$$\mathbf{\Phi}^{-1}(n) = \lambda^{-1}\mathbf{\Phi}^{-1}(n-1) -$$

$$-\lambda^{-1}\mathbf{\Phi}^{-1}(n-1)\mathbf{x}(n)\left(1 + \mathbf{x}^H(n)\lambda^{-1}\mathbf{\Phi}^{-1}(n-1)\mathbf{x}(n)\right)^{-1}\mathbf{x}^H(n)\lambda^{-1}\mathbf{\Phi}^{-1}(n-1)$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

$$\mathbf{\Phi}^{-1}(n) = \lambda^{-1}\mathbf{\Phi}^{-1}(n-1) -$$

$$-\lambda^{-1}\mathbf{\Phi}^{-1}(n-1)\mathbf{x}(n)\left(1+\mathbf{x}^{H}(n)\lambda^{-1}\mathbf{\Phi}^{-1}(n-1)\mathbf{x}(n)\right)^{-1}\mathbf{x}^{H}(n)\lambda^{-1}\mathbf{\Phi}^{-1}(n-1)$$

$$\mathbf{\Phi}^{-1}(n) = \frac{1}{\lambda}\mathbf{\Phi}^{-1}(n-1) - \frac{1}{\lambda^{2}}\frac{\mathbf{\Phi}^{-1}(n-1)\mathbf{x}(n)\mathbf{x}^{H}(n)\mathbf{\Phi}^{-1}(n-1)}{1+\frac{1}{\lambda}\mathbf{x}^{H}(n)\mathbf{\Phi}^{-1}(n-1)\mathbf{x}(n)}$$

$$\mathbf{P}(n) \triangleq \mathbf{\Phi}^{-1}(n);$$

$$\mathbf{k}(n) \triangleq \frac{1}{\lambda}\frac{\mathbf{P}(n-1)\mathbf{x}(n)}{1+\frac{1}{\lambda}\mathbf{x}^{H}(n)\mathbf{P}(n-1)\mathbf{x}(n)}$$

$$\mathbf{P}(n) = \frac{1}{\lambda}\mathbf{P}(n-1) - \frac{1}{\lambda}\mathbf{k}(n)\mathbf{x}^{H}(n)\mathbf{P}(n-1)$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

P(n) este o matrice pătrată NxN;
k(n) este un vector Nx1, numit *vectorul câştig* (*Kalman*).
Ecuaţia de mai sus este o ecuaţie de tip *Riccati*.

$$\mathbf{k}(n) = \frac{1}{\lambda}\mathbf{P}(n-1)\mathbf{x}(n) - \frac{1}{\lambda}\mathbf{k}(n)\mathbf{x}^H(n)\mathbf{P}(n-1)\mathbf{x}(n) =$$

$$= \left(\frac{1}{\lambda}\mathbf{P}(n-1) - \frac{1}{\lambda}\mathbf{k}(n)\mathbf{x}^H(n)\mathbf{P}(n-1)\right)\mathbf{x}(n)$$

deci

$$\boxed{\begin{gathered}\mathbf{k}(n) = \mathbf{P}(n)\mathbf{x}(n)\\ \mathbf{\Phi}(n)\mathbf{k}(n) = \mathbf{x}(n)\end{gathered}}$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Avem acum toate elementele pentru exprimarea lui **w**(n).

$$\mathbf{w}(n) = \mathbf{\Phi}^{-1}(n)\boldsymbol{\theta}(n) = \mathbf{P}(n)\boldsymbol{\theta}(n) = \lambda\mathbf{P}(n)\boldsymbol{\theta}(n-1) + \mathbf{P}(n)\mathbf{x}(n)d^*(n)$$

$$\mathbf{w}(n) = \mathbf{P}(n-1)\boldsymbol{\theta}(n-1) - \mathbf{k}(n)\mathbf{x}^H(n)\mathbf{P}(n-1)\boldsymbol{\theta}(n-1) + \mathbf{P}(n)\mathbf{x}(n)d^*(n) =$$

$$= \mathbf{\Phi}^{-1}(n-1)\boldsymbol{\theta}(n-1) - \mathbf{k}(n)\mathbf{x}^H(n)\mathbf{\Phi}^{-1}(n-1)\boldsymbol{\theta}(n-1) + \mathbf{k}(n)d^*(n)$$

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mathbf{k}(n)\left(d^*(n) - \mathbf{x}^H(n)\mathbf{w}(n-1)\right) = \mathbf{w}(n-1) + \mathbf{k}(n)\alpha^*(n)$$

unde

$$\boxed{\begin{aligned} \mathbf{w}(n) &= \mathbf{w}(n-1) + \mathbf{k}(n)\alpha^*(n) \\ \alpha(n) &= d(n) - \mathbf{w}^H(n-1)\mathbf{x}(n) \end{aligned}}$$

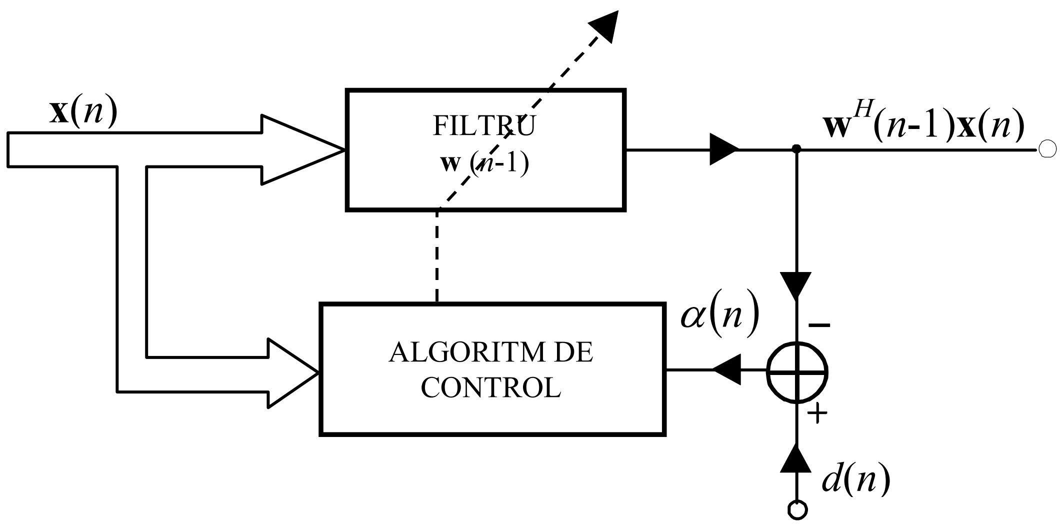
reprezintă eroarea estimării obţinute utilizând vechile valori **w**(n-1) ale coeficienţilor pentru noile date, deci *eroarea de estimare apriori*, numită şi *inovaţie*. Ea nu coincide cu eroarea aposteriori

$$e(n) = d(n) - \mathbf{w}^H(n)\mathbf{x}(n)$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

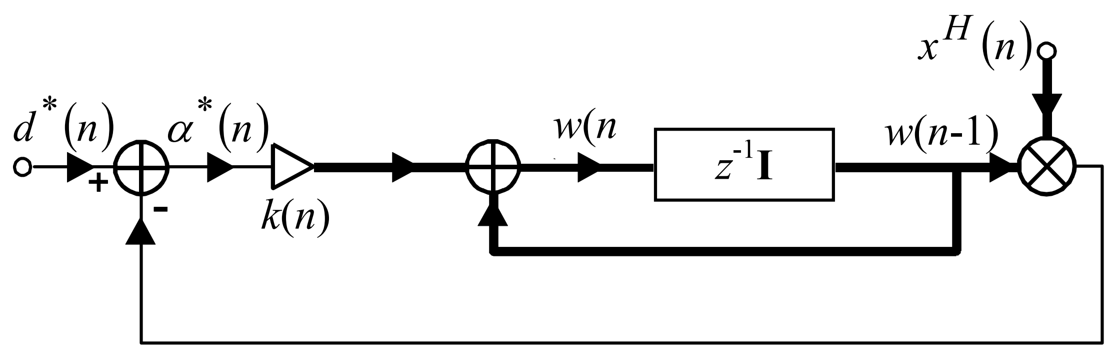
$$\boxed{\begin{aligned}\mathbf{w}(n) &= \mathbf{w}(n-1)+\mathbf{k}(n)\alpha^*(n)\\ \alpha(n) &= d(n)-\mathbf{w}^H(n-1)\mathbf{x}(n)\end{aligned}}$$



7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

$$\mathbf{w}(n) = \mathbf{w}(n-1) + \mathbf{k}(n)\alpha^*(n)$$
$$\alpha(n) = d(n) - \mathbf{w}^H(n-1)\mathbf{x}(n)$$



7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Relaţiile de ortogonalitate

$$\sum_{i=1}^{n} \lambda^{n-i} \mathbf{x}(i) e_{\min}^{*}(i) = \mathbf{0}$$

$$\sum_{i=1}^{n} \lambda^{n-i} y_0(i) e_{\min}^{*}(i) = 0$$

Consecinţă

$$E_{\min}(n) = J_{\min}(n) = E_d(n) - E_y(n)$$

În cazul de faţă:

$$E_y(n) = \sum_{i=1}^{n} \lambda^{n-i} y_o(i) y_o^{*}(i) = \sum_{i=1}^{n} \lambda^{n-i} \mathbf{w}^H(n) \mathbf{x}(i) \mathbf{x}^H(i) \mathbf{w}(n) =$$

$$= \mathbf{w}^H(n) \mathbf{\Phi}(n) \mathbf{w}(n) = \boldsymbol{\theta}^H(n) \mathbf{w}(n)$$

aşa încât rezultă o primă expresie utilă:

$$J_{\min}(n) = E_d(n) - \boldsymbol{\theta}^H(n) \mathbf{w}(n)$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

$$E_d(n) = \sum_{i=1}^{n} \lambda^{n-i} |d(i)|^2 = \lambda \sum_{i=1}^{n-1} \lambda^{n-1-i} |d(i)|^2 + |d(n)|^2 = \lambda E_d(n-1) + |d(n)|^2$$

Ţinând seama şi de relaţiile de recurenţă pentru $\boldsymbol{\theta}(n)$ şi $\mathbf{w}(n)$, rezultă

$$J_{\min}(n) = \lambda E_d(n-1) + |d(n)|^2 - \left(\lambda \boldsymbol{\theta}^H(n-1) + d(n)\mathbf{x}^H(n)\right)\left(\mathbf{w}(n-1) + \mathbf{k}(n)\alpha^*(n)\right) =$$

$$= \lambda\left(E_d(n-1) - \boldsymbol{\theta}^H(n-1)\mathbf{w}(n-1)\right) +$$

$$+ d(n)\left(d^*(n) - \mathbf{x}^H(n)\mathbf{w}(n-1)\right) - \left(\lambda \boldsymbol{\theta}^H(n-1) + d(n)\mathbf{x}^H(n)\right)\mathbf{k}(n)\alpha^*(n) =$$

$$= \lambda J_{\min}(n-1) + d(n)\alpha^*(n) - \boldsymbol{\theta}^H(n)\mathbf{k}(n)\alpha^*(n)$$

Ultimul termen al sumei se mai scrie

$$\boldsymbol{\theta}^H(n)\mathbf{k}(n)\alpha^*(n) = \boldsymbol{\theta}^H(n)\boldsymbol{\Phi}^{-1}(n)\mathbf{x}(n)\alpha^*(n) = \mathbf{w}^H(n)\mathbf{x}(n)\alpha^*(n)$$

Revenind la $J_{\min}(n)$ se obţine:

$$J_{\min}(n) = \lambda J_{\min}(n-1) + \alpha^*(n)\left(d(n) - \mathbf{w}^H(n)\mathbf{x}(n)\right)$$

$$\boxed{J_{\min}(n) = \lambda J_{\min}(n-1) + \alpha^*(n)e(n)}$$

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Iniţializare

$$\mathbf{P}(0)=\delta^{-1}\mathbf{I},\quad \mathbf{w}(n)=\mathbf{0}$$

for n=1,2,…

$$\mathbf{z}(n)=\mathbf{x}^{H}(n)\mathbf{P}(n-1)$$

(N^2 înmulţiri şi N (N-1) adunări)

$$\mathbf{k}(n)=\frac{1}{\lambda+\mathbf{z}(n)\mathbf{x}(n)}\mathbf{z}^{H}(n)$$

(2N înmulţiri şi N adunări)

$$\alpha(n)=d(n)-\mathbf{w}^{H}(n-1)\mathbf{x}(n)$$

(N înmulţiri şi N adunări)

$$\mathbf{w}(n)=\mathbf{w}(n-1)+\mathbf{k}(n)\alpha^{*}(n)$$

(N înmulţiri şi N adunări)

$$\mathbf{P}(n)=\frac{1}{\lambda}\left(\mathbf{P}(n-1)-\mathbf{k}(n)\mathbf{z}(n)\right)$$

(2N^2 înmulţiri şi N^2 adunări)

end

7.1 Metoda recursivă a celor mai mici pătrate (RLS)

7.1.1 Algoritmul RLS

Complexitatea aritmetică a algoritmului

Iniţializarea algoritmului constă în impunerea valorilor **P**[0] şi **w**[0]. Uzual se ia **w**[0]=**0** şi

$$\mathbf{P}(0) = \delta^{-1}\mathbf{I}$$

unde δ este o valoare constantă mică, astfel încât matricea de autocorelaţie să nu rezulte singulară.

Complexitatea aritmetică este de $3N^2 + 4N$ înmulţiri şi $2N^2 + 2N$ adunări, deci numărul de operaţii este de forma O(N^2), (creşte cu N^2).