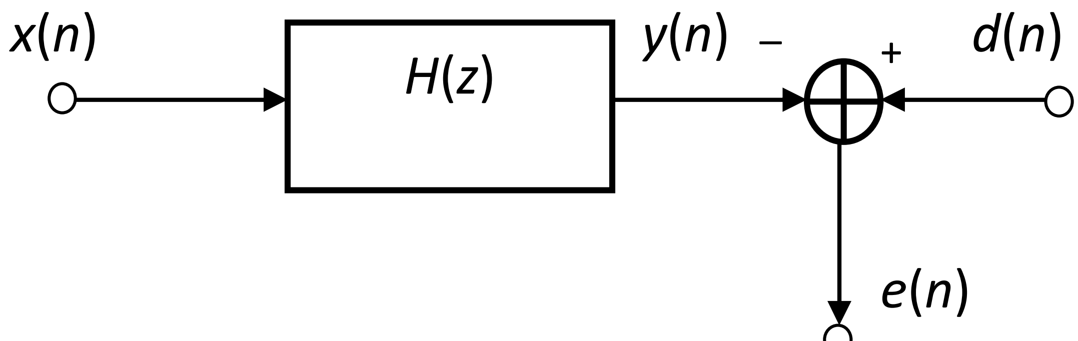
3.TEORIA FILTRĂRII LINIARE OPTIMALE

3.1 CRITERIUL DE OPTIMIZARE



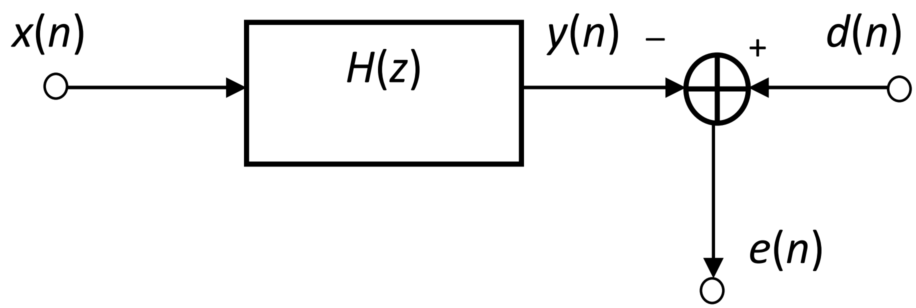
Se cunoaşte semnalul staţionar în sens larg, de medie nulă

$$x(n) = d(n) + v(n),$$

Semnal util   Perturbaţie

Dorim un filtru $H(z)$ care să extragă semnalul util.

3.1 CRITERIUL DE OPTIMIZARE



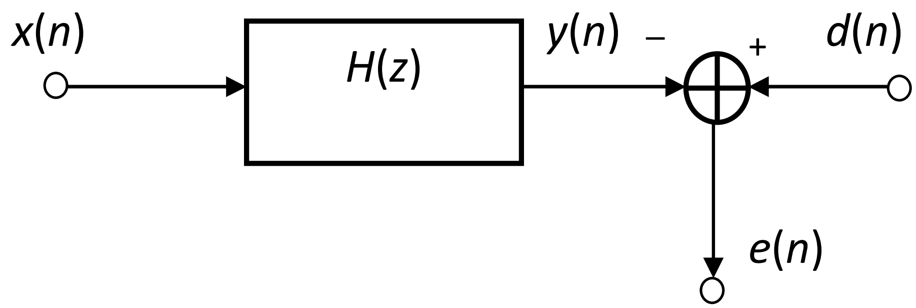
$$h(n) = w_n^*, \quad n = 0,1,\ldots,N-1$$

$$y(n) = (h * x)(n) = \sum_{k=0}^{N-1} w_k^* x(n-k), \quad n = 0,1,\ldots.$$

şi

$$e(n) = d(n) - \sum_{k=0}^{N-1} w_k^* x(n-k)$$

3.1 CRITERIUL DE OPTIMIZARE



Cu notaţiile

$$\mathbf{x}(n) = [x(n), \ldots, x(n-N+1)]^T;$$

$$\mathbf{w} = [w_0, w_1, \ldots, w_{N-1}]^T;$$

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

3.1 CRITERIUL DE OPTIMIZARE

Funcţie cost: ***eroarea medie pătratică***

$$\boxed{J = \mathrm{E}\left\{\left|e(n)\right|^2\right\} = \mathrm{E}\left\{e(n)e^*(n)\right\}}$$

$$J = E\{[d(n) - \boldsymbol{w}^H \boldsymbol{x}(n)][d^*(n) - \boldsymbol{x}^H(n)\boldsymbol{w}]\}$$
$$J = E\{d(n)d^*(n)\} + \boldsymbol{w}^H E\{\boldsymbol{x}(n)\boldsymbol{x}^H(n)\}\boldsymbol{w} - \boldsymbol{w}^H E\{\boldsymbol{x}(n)d^*(n)\} -$$
$$-E\{d(n)\boldsymbol{x}^H(n)\}\boldsymbol{w}$$

$E\{d(n)d^*(n)\} = \sigma_d^2$ (puterea semnalului dorit)

$E\{\boldsymbol{x}(n)\boldsymbol{x}^H(n)\} = \boldsymbol{R}$ (matricea de autocorelaţie)

$E\{\boldsymbol{x}(n)d^*(n)\} = \boldsymbol{p}$ (vectorul corelaţiei între semnalul de intrare şi semnalul dorit)

$$\mathbf{p} = \left[r_{xd}(0), r_{xd}(-1), \ldots, r_{xd}(-N+1)\right]^T;$$

3.1 CRITERIUL DE OPTIMIZARE

$$J = E\{d(n)d^*(n)\} + \boldsymbol{w}^H E\{\boldsymbol{x}(n)\boldsymbol{x}^H(n)\}\boldsymbol{w} - \boldsymbol{w}^H E\{\boldsymbol{x}(n)d^*(n)\} - E\{d(n)\boldsymbol{x}^H(n)\}\boldsymbol{w}$$

$$\boxed{J = \sigma_d^2 - \mathbf{w}^H\mathbf{p} - \mathbf{p}^H\mathbf{w} + \mathbf{w}^H\mathbf{R}\mathbf{w}}$$

3.2 ECUAŢIILE WIENER-HOPF

J - funcţie de gradul doi de variabilele complexe

$w_k = a_k + \mathrm{j} b_k$, $k = 0,\ldots,N$-1.

$$J_{\min} \;\Rightarrow\; \nabla_{\mathbf{w}^*} J = \mathbf{0}$$

∇J - gradientul complex,

$$\nabla_{\mathbf{w}} J = \left[\frac{\partial J}{\partial w_0}, \frac{\partial J}{\partial w_1}, \ldots, \frac{\partial J}{\partial w_{N-1}} \right]^T = \mathbf{0}$$

$$\frac{\partial J}{\partial w_i} = \frac{1}{2}\left(\frac{\partial J}{\partial a_i} - \mathrm{j}\frac{\partial J}{\partial b_i} \right); \qquad \frac{\partial J}{\partial w_i^*} = \frac{1}{2}\left(\frac{\partial J}{\partial a_i} + \mathrm{j}\frac{\partial J}{\partial b_i} \right)$$

Minimizarea unei funcții reale de un vector complex

Fie o funcție $f(\boldsymbol{z}, \boldsymbol{z}^*): \mathbb{C}^{2N} \to \mathbb{R}$. Condiția necesară pentru ca funcția să aibă un minim în $\boldsymbol{z}$ este ca gradientul complex să se anuleze $\nabla_{\mathrm{z}^*}\{f(\boldsymbol{z}, \boldsymbol{z}^*)\} = \boldsymbol{0}$.

Prin definiție

$$\nabla_{\mathbf{z}} f = \left[\frac{\partial f}{\partial z_0}, \frac{\partial f}{\partial z_1}, \ldots, \frac{\partial f}{\partial z_{N-1}}\right]^T$$

$$\nabla_{\mathbf{z}^*} f = \left[\frac{\partial f}{\partial z_0^*}, \frac{\partial f}{\partial z_1^*}, \ldots, \frac{\partial f}{\partial z_{N-1}^*}\right]^T$$

$$\frac{\partial f}{\partial z_i} = \frac{1}{2}\left(\frac{\partial f}{\partial a_i} - \mathrm{j}\frac{\partial f}{\partial b_i}\right), \quad \frac{\partial f}{\partial z_i^*} = \frac{1}{2}\left(\frac{\partial f}{\partial a_i} + \mathrm{j}\frac{\partial f}{\partial b_i}\right), z_i = a_i + jb_i$$

Exemple

$$\frac{\partial z_i}{\partial z_i} = \frac{1}{2}\left(\frac{\partial}{\partial a_i} - \mathrm{j}\frac{\partial}{\partial b_i}\right)\left(a_i + jb_i\right) = 1$$

$$\frac{\partial z_i}{\partial z^*_i} = \frac{1}{2}\left(\frac{\partial}{\partial a_i} + \mathrm{j}\frac{\partial}{\partial b_i}\right)\left(a_i + jb_i\right) = 0$$

$$\frac{\partial z^*_i}{\partial z_i} = \frac{1}{2}\left(\frac{\partial}{\partial a_i} - \mathrm{j}\frac{\partial}{\partial b_i}\right)\left(a_i - jb_i\right) = 0$$

$$\nabla_{\mathbf{z}}\mathbf{z}^* = \mathbf{0}; \quad \nabla_{\mathbf{z}^*}\mathbf{z} = \mathbf{0}$$

Observație – $\mathbf{z}$ și $\mathbf{z}^*$ sunt tratate ca două variabile distincte.

Gradientul în raport $\mathbf{z}$ al unei cantități ce nu depinde de $\mathbf{z}$ este nul.

$\nabla_{\mathrm{z}}\{\mathbf{p}^{\mathrm{H}}\mathbf{z}\} = \mathbf{p}^*$ $\qquad\qquad$ $\nabla_{\mathrm{z}^*}\{\mathbf{p}^{\mathrm{H}}\mathbf{z}\} = \mathbf{0}$

$$\frac{\partial}{\partial z_i}\sum_{l=0}^{N-1} p_l^*\left(a_l + jb_l\right) = \frac{1}{2}\left(\frac{\partial}{\partial a_i}\text{-j}\frac{\partial}{\partial b_i}\right)\left\{\sum_{l=0}^{N-1} p_l^*\left(a_l + jb_l\right)\right\} = p_i^*$$

$$\frac{\partial}{\partial z_i}\sum_{l=0}^{N-1} p_l^*\left(a_l + jb_l\right) = \frac{1}{2}\left(\frac{\partial}{\partial a_i}+\text{j}\frac{\partial}{\partial b_i}\right)\left\{\sum_{l=0}^{N-1} p_l^*\left(a_l + jb_l\right)\right\} = 0$$

$\nabla_{\mathrm{z}}\{\mathbf{z}^{\mathrm{H}}\mathbf{p}\} = \mathbf{0}$ $\qquad\qquad$ $\nabla_{\mathrm{z}^*}\{\mathbf{z}^{\mathrm{H}}\mathbf{p}\} = \mathbf{p}$

$\nabla_{\mathrm{z}}\{\mathbf{z}^{\mathrm{H}}\mathbf{A}\mathbf{z}\} = (\mathbf{A}\mathbf{z})^*$ $\qquad\qquad$ $\nabla_{\mathrm{z}^*}\{\mathbf{z}^{\mathrm{H}}\mathbf{A}\mathbf{z}\} = (\mathbf{A}\mathbf{z})$

Pentru ca punctual staționar respective să fie punct de extrem, hessianul

$$\nabla^2_{\mathbf{zz}^*}\{f\} \begin{cases} > 0 \; pentru \; un \; minim \\ < 0 \; pentru \; un \; maxim \end{cases}$$

unde semnul de inegalitate semnifică ‘*pozitiv semidefinit*’,

respectiv ‘*negativ semidefinit*’.

3.2 ECUAŢIILE WIENER-HOPF

$$\frac{\partial}{\partial w_i^*}\left\{\mathbf{w}^H\mathbf{p}+\mathbf{p}^H\mathbf{w}\right\}=$$

$$=\frac{1}{2}\left(\frac{\partial}{\partial a_i}+\mathrm{j}\frac{\partial}{\partial b_i}\right)\left\{\sum_{k=0}^{N-1}\left(a_k-\mathrm{jb_k}\right)r_{xd}\left(-k\right)+\sum_{k=0}^{N-1}\left(a_k+\mathrm{jb_k}\right)r_{xd}^*\left(-k\right)\right\}=$$

$$=r_{xd}\left(-i\right)$$

$$\nabla\left\{\mathbf{w}^H\mathbf{p}+\mathbf{p}^H\mathbf{w}\right\}=\mathbf{p}$$

3.2 ECUAŢIILE WIENER-HOPF

$$\frac{\partial}{\partial w_i^*}\left\{\mathbf{w}^H\mathbf{R}\mathbf{w}\right\} =$$

$$= \frac{1}{2}\left(\frac{\partial}{\partial a_i} + \mathrm{j}\frac{\partial}{\partial b_i}\right)\left\{\sum_{k=0}^{N-1}\sum_{l=0}^{N-1}\left(a_k - \mathrm{jb_k}\right)\left(a_l + \mathrm{j}b_l\right)r_{xx}\left(l-k\right)\right\} =$$

$$= \sum_{l=0}^{N-1} w_l r_{xx}\left(l-i\right)$$

$$\nabla\left\{\mathbf{w}^H\mathbf{R}\mathbf{w}\right\} = \mathbf{R}\mathbf{w};$$

$$\nabla J = -\mathbf{p} + \mathbf{R}\mathbf{w} = \mathbf{0}$$

3.2 ECUAŢIILE WIENER-HOPF

Hessianul transformării:

$$\mathbf{H} = \nabla^2_{\mathbf{w}\mathbf{w}^*} J = \nabla_{\mathbf{w}}\left(\nabla_{\mathbf{w}^*}\right) = \nabla_{\mathbf{w}}\left(-\mathbf{p} + \mathbf{R}\mathbf{w}\right) = \mathbf{R}$$

Este pozitiv semidefinit, J(**w**) reprezintă o suprafaţă de forma unui paraboloid având un minim J_{min} pentru $\mathbf{w}=\mathbf{w}_o$ care anulează gradientul

$$\boxed{\mathbf{R}\mathbf{w}_o = \mathbf{p}}$$

$$\boxed{\sum_{l=0}^{N-1} w_{o,l} r_{xx}\left(l-i\right) = r_{xd}(-i), \quad i = 0,\ldots,N-1}$$

$$J_{\min} = \sigma_d^2 - \mathbf{w}_o^H \mathbf{p} - \mathbf{p}^H \mathbf{w}_o + \mathbf{w}_o^H \mathbf{R}\mathbf{w}_o =$$

$$= \sigma_d^2 - \mathbf{w}_o^H \mathbf{p} - \mathbf{p}^H \mathbf{w}_o + \mathbf{w}_o^H \mathbf{p} = \sigma_d^2 - \mathbf{p}^H \mathbf{w}_o = \sigma_d^2 - \mathbf{p}^H \mathbf{R}^{-1}\mathbf{p}$$

3.3 PRINCIPIUL ORTOGONALITĂŢII

Pentru filtrul optim:

$$y_o(n) = \mathbf{w}_o^H \mathbf{x}(n)$$

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

$$\mathrm{E}\left\{\mathbf{x}(n)e_o^*(n)\right\} = \mathrm{E}\left\{\mathbf{x}(n)d_o^*(n)\right\} - \mathrm{E}\left\{\mathbf{x}(n)\mathbf{x}^H(n)\mathbf{w}_o\right\}$$

$$\mathrm{E}\left\{\mathbf{x}(n)e_o^*(n)\right\} = \mathbf{p} - \mathbf{R}\mathbf{w}_o = \mathbf{0}$$

$$\mathrm{E}\left\{\mathbf{x}(n)e_o^*(n)\right\} = \mathbf{0}$$

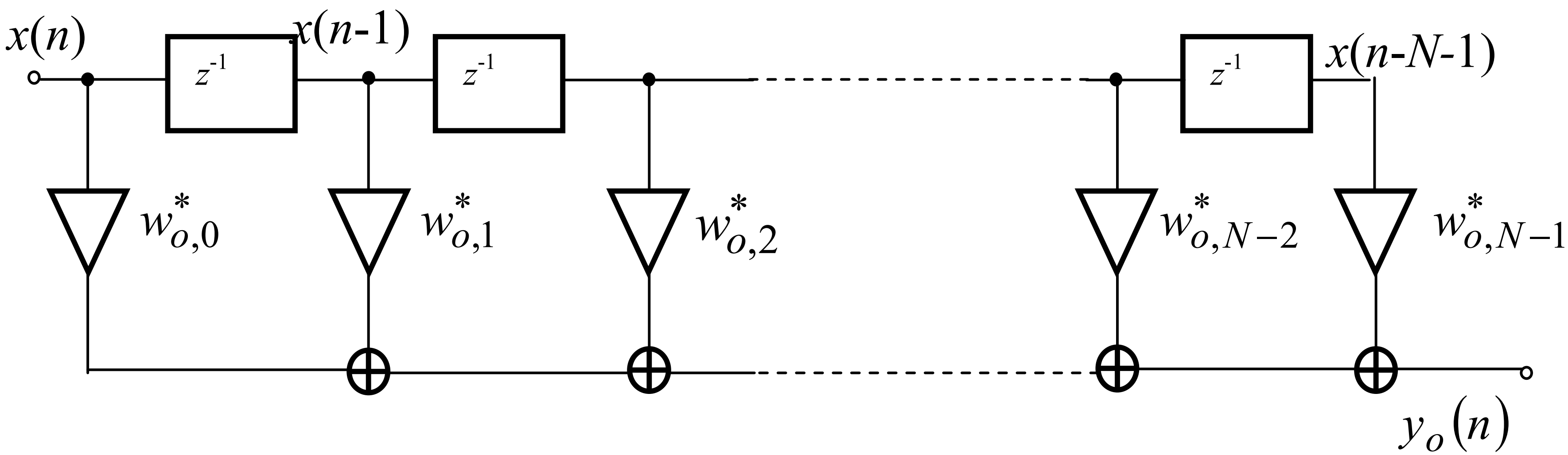
$$\mathrm{E}\left\{x(n-k)e_o^*(n)\right\} = 0, \quad k = 0,\dots,N-1$$

- ***eroarea este ortogonală pe eşantioanele intrării***.

3.3 PRINCIPIUL ORTOGONALITĂŢII

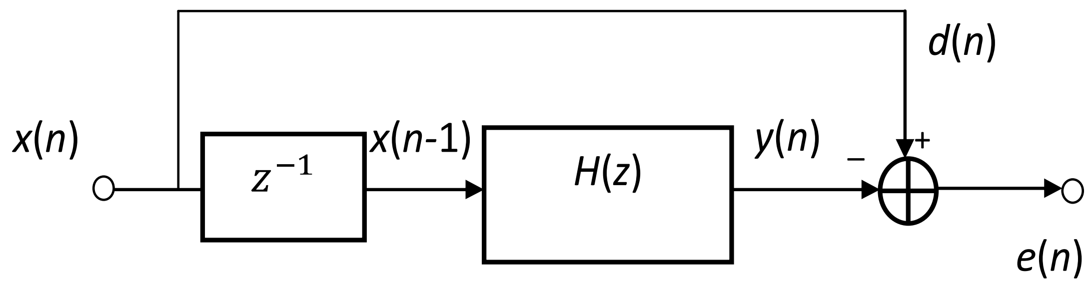
$$\boxed{\mathrm{E}\left\{y_o(n)e_o^*(n)\right\} = 0}$$

- ***ieşirea $y_o(n)$ şi eroarea $e_o(n)$ corespunzătoare sunt ortogonale***



4. PREDICŢIA LINIARĂ

4.1 Predicţia înainte (directă)



$x(n)$ un proces aleator staţionar cu valoare medie nulă.
Se cunosc N eşantioane anterioare

$$x(n-1), x(n-2), \ldots, x(n-N) \;\Rightarrow\; X_{n-1}$$

Pe baza lor se doreşte o estimare a lui $x(n)$:

$$\hat{x}(n|X_{n-1}) = \sum_{k=1}^{N} w_k^* x(n-k) = \mathbf{w}^H \mathbf{x}(n-1),$$

$$\mathbf{x}(n-1) = [x(n-1), x(n-2), \ldots, x(n-N)]^T, \quad \mathbf{w} = [w_1, w_2, \ldots, w_N]^T$$

4.1 Predicţia înainte (directă)

eroarea de predicţie:

$$e_N^f(n) = x(n) - \hat{x}(n|X_{n-1})$$

Puterea erorii de predicţie:

$$P_N = \mathrm{E}\left\{\left|e_N^f(n)\right|^2\right\}.$$

4.1 Predicţia înainte (directă)

Analogii cu problema filtrării optimale:

$$d(n) \Rightarrow x(n)$$

$$\mathbf{x}(n) \Rightarrow \mathbf{x}(n-1)$$

$$e(n) \Rightarrow e_N^f(n)$$

$$J_{\min} \Rightarrow P_N$$

$$\mathbf{R} \Rightarrow \mathrm{E}\left\{\mathbf{x}(n-1)\mathbf{x}^H(n-1)\right\} = \mathbf{R}$$

$$\mathbf{p} \Rightarrow \mathbf{r} = \mathrm{E}\left\{\mathbf{x}(n-1)x^*(n)\right\} = \left[r_{xx}(-1), r_{xx}(-2), \ldots, r_{xx}(-N)\right]^T$$

$$\sigma_d^2 = \mathrm{E}\left\{\left|x(n)\right|^2\right\} = \sigma_x^2 = r_{xx}(0)$$

4.1 Predicţia înainte (directă)

Se pot prelua direct următoarele rezultate:

- ***ecuaţia Wiener-Hopf*** **(*ecuaţia normală*)**

$$\mathbf{R}\mathbf{w}_o = \mathbf{r} \quad \text{sau} \quad \mathbf{w}_o = \mathbf{R}^{-1}\mathbf{r}$$

- ***principiul ortogonalităţii***

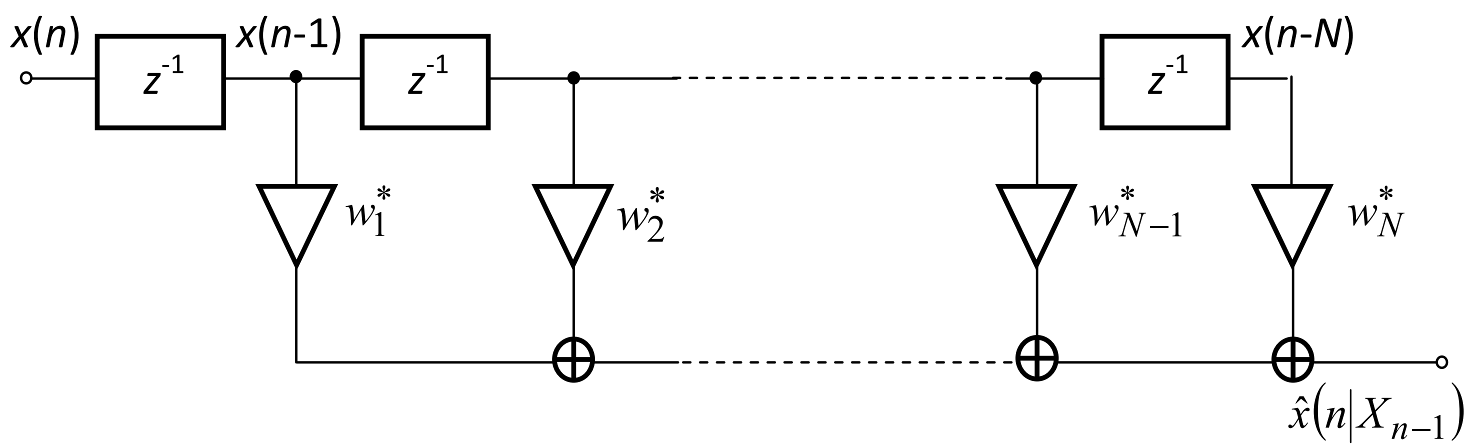
$$\mathrm{E}\left\{\mathbf{x}(n-1)e_N^{f*}(n)\right\} = \mathbf{0} \qquad \mathrm{E}\left\{\hat{x}\left(n \middle| X_{n-1}\right)e_N^{f*}(n)\right\} = 0$$

- ***Expresia puterii erorii de predicţie***, în cazul coeficienţilor optimi,

$$P_N = \sigma_x^2 - \mathbf{r}^H\mathbf{w}_o = r_{xx}(0) - \mathbf{r}^H\mathbf{w}_o = r(0) - \mathbf{r}^H\mathbf{w}_o$$

4.1 Predicţia înainte (directă)

Filtrul predictor:



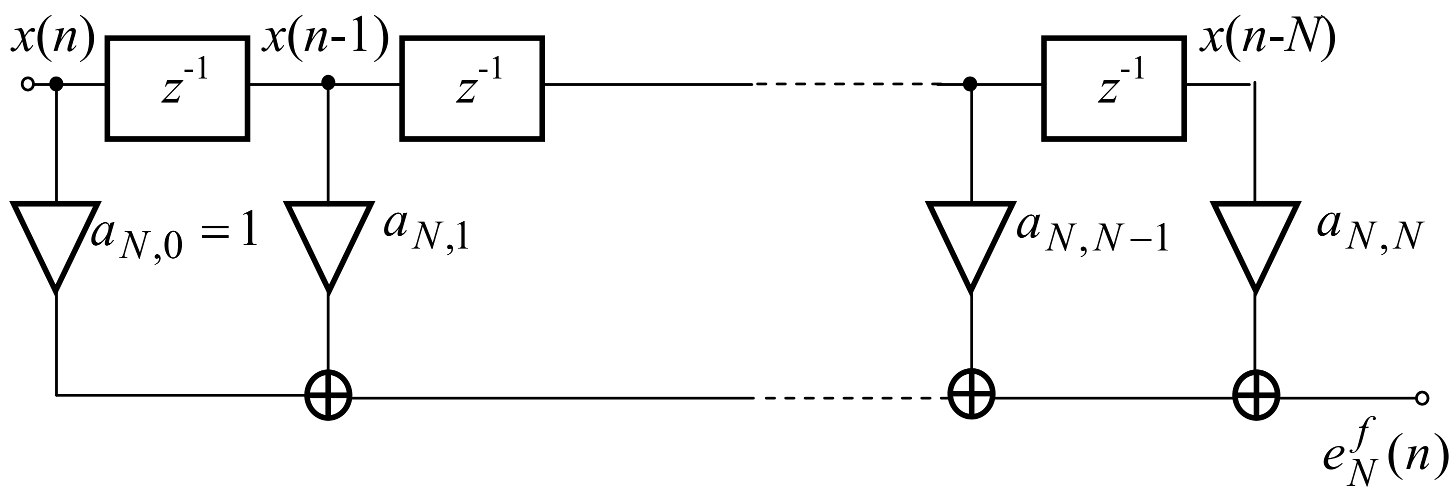
4.1 Predicţia înainte (directă)

Filtrul erorii de predicţie inainte

$$e_N^f(n) = x(n) - \sum_{k=1}^{N} w_{o,k}^* x(n-k) = \sum_{k=0}^{N} a_{N,k} x(n-k)$$

unde

$$a_{N,k} = \begin{cases} 1, & k = 0 \\ -w_{o,k}^*, & k = 1, \ldots, N \end{cases}$$



4.1 Predicţia înainte (directă)

Ecuaţiile Wiener-Hopf extinse:

$$\mathbf{r} - \mathbf{R}\mathbf{w}_o = 0$$
$$r(0) - \mathbf{r}^H \mathbf{w}_o = P_N$$

$$\begin{bmatrix} r(0) & \mathbf{r}^H \\ \mathbf{r} & \mathbf{R} \end{bmatrix} \begin{bmatrix} 1 \\ -\mathbf{w}_o \end{bmatrix} = \begin{bmatrix} P_N \\ \mathbf{0} \end{bmatrix}$$

4.1 Predicţia înainte (directă)

$$\begin{bmatrix} r(0) & \mathbf{r}^H \\ \mathbf{r} & \mathbf{R} \end{bmatrix} \begin{bmatrix} 1 \\ -\mathbf{w}_o \end{bmatrix} = \begin{bmatrix} P_N \\ \mathbf{0} \end{bmatrix}$$

Sau

$$\mathbf{R}_{N+1}\mathbf{a}_N^* = \begin{bmatrix} P_N \\ 0 \end{bmatrix}$$

unde

$$\mathbf{a}_N = \begin{bmatrix} 1 \\ -\mathbf{w}_o^* \end{bmatrix}$$

şi $\mathbf{R}_{N+1}$ este matricea de autocorelaţie extinsă, de dimensiuni (N+1)x(N+1)

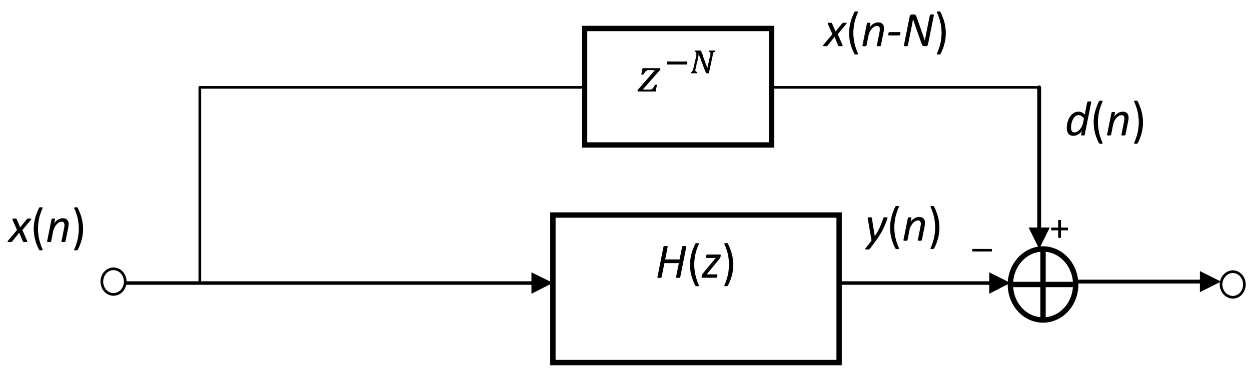
4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

Se cunosc:

$$x(n), x(n-1), \cdots, x(n-N+1) \;\Rightarrow\; X_n$$

Se doreşte un estimat al lui

$$x(n-N)$$



4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

$$\hat{x}(n-N|X_n)=\sum_{k=1}^{N} g_k^* x(n-k+1)$$

Eroarea de predicţie:

$$e_N^b(n)=x(n-N)-\hat{x}(n-N|X_n)$$

$$\mathbf{g}=[g_1, g_2, ..., g_N]^T$$

Puterea erorii de predicţie

$$P_N = \mathrm{E}\left\{\left|e_N^b(n)\right|^2\right\}$$

4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

Analogii cu problema filtrării optimale:

$$d(n) \Rightarrow x(n-N)$$

$$\mathbf{w}_o \Rightarrow \mathbf{g}_o$$

$$e(n) \Rightarrow e_N^b(n)$$

$$J_{\min} \Rightarrow P_N$$

$$\mathbf{p} \;\; \Rightarrow \;\; \mathbf{r}^{B*} = \mathrm{E}\{\mathbf{x}(n)x^*(n-N)\} = [r_{xx}(N),\ r_{xx}(N-1), \ldots, r_{xx}(1)]^T.$$

4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

Se preiau:

- Ecuaţia normală $$\mathbf{R}\mathbf{g} = \mathbf{r}^{B*}$$

- Puterea erorii de predicţie $$P_N = r(0) - \mathbf{r}^{BT}\mathbf{g}$$

- Principiul ortogonalităţii $$\mathrm{E}\{\mathbf{x}(n)e_N^{b*}(n)\} = \mathbf{0}$$

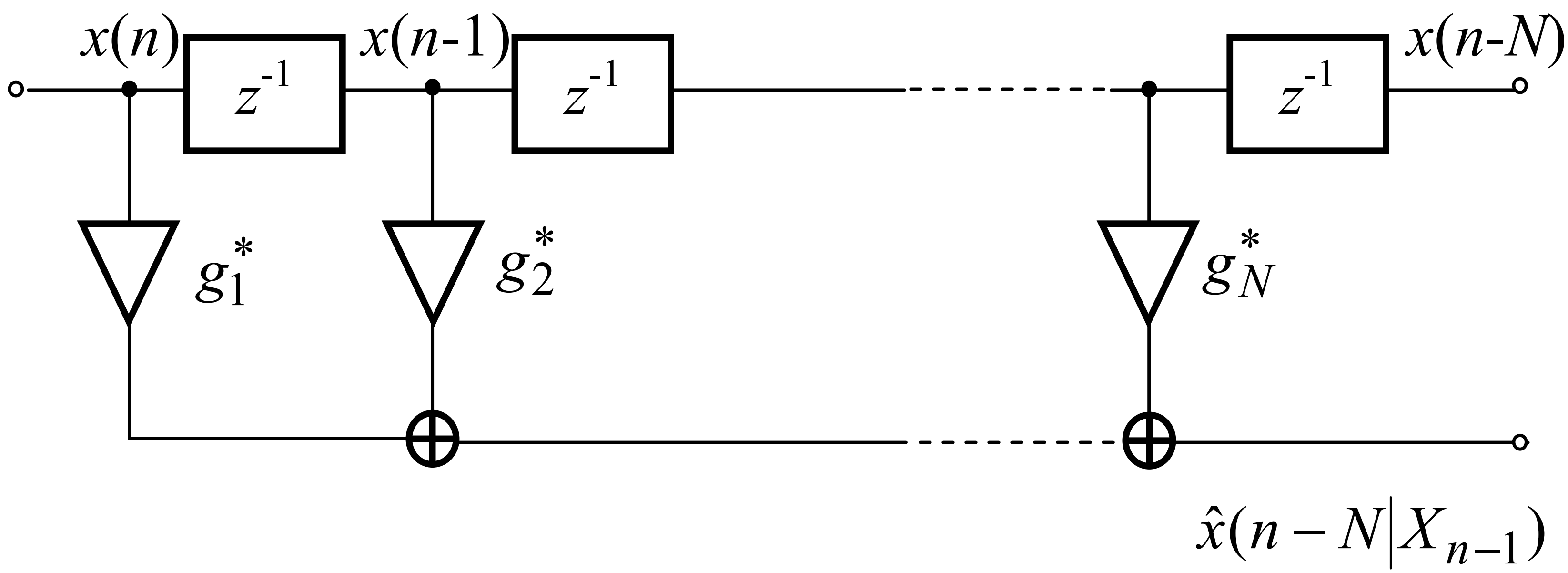
sau

$$\mathrm{E}\{x(n-k)e_N^{b*}(n)\} = 0, \quad k = 0,1,\ldots,N-1$$

4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

Filtrul ce realizează predicţia inversă

$$\hat{x}\left(n-N\middle|X_n\right)=\sum_{k=1}^{N} g_k^* x(n-k+1)$$



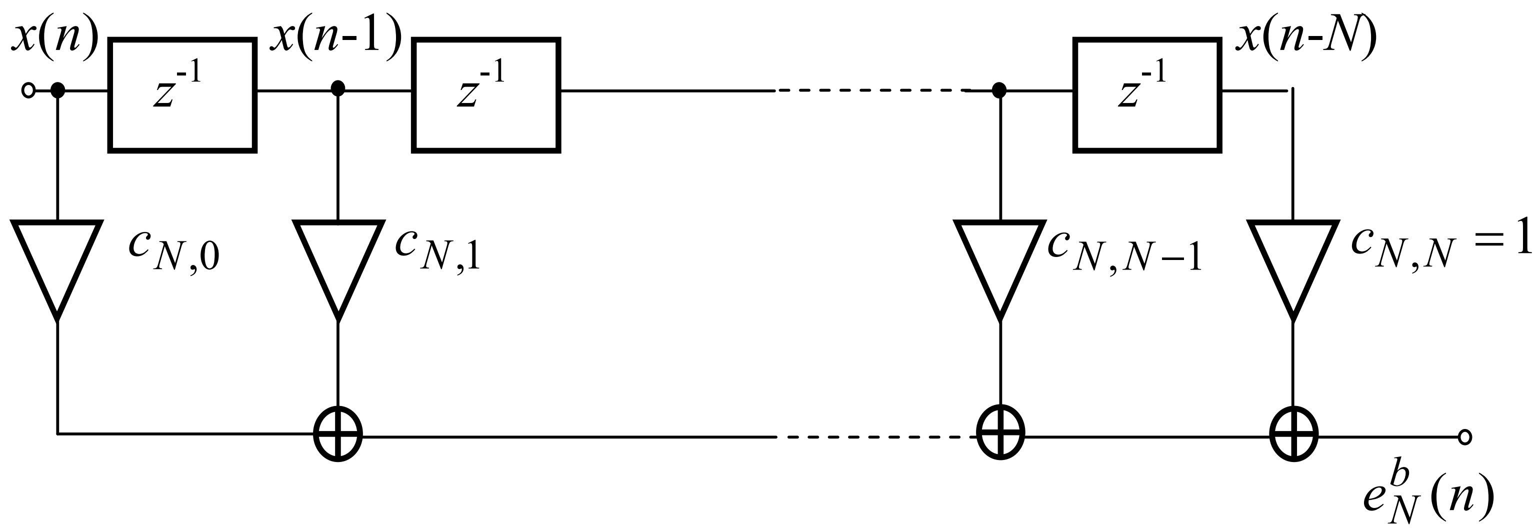
4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

***Filtrul erorii de predicţie înapoi*.**

$$e_N^b(n) = x(n-N) - \sum_{k=1}^{N} g_k^* x(n-k+1) = \sum_{k=0}^{N} c_{N,k}\, x(n-k)$$

unde

$$c_{N,k} = \begin{cases} -g_{k+1}^* & k = 0,\ldots,N-1 \\ 1, & k = N \end{cases}$$



4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

Ecuaţiile Wiener-Hopf extinse pentru predicţia inversă

$$-\mathbf{R}\mathbf{g} + \mathbf{r}^{B*} = \mathbf{0}$$
$$r(0) - \mathbf{r}^{BT}\mathbf{g} = P_N$$

$$\begin{bmatrix} \mathbf{R} & \mathbf{r}^{B*} \\ \mathbf{r}^{BT} & r(0) \end{bmatrix} \begin{bmatrix} -\mathbf{g} \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ P_N \end{bmatrix}$$

sau

$$\boxed{\mathbf{R}_{N+1}\mathbf{c}_N^{*} = \begin{bmatrix} \mathbf{0} \\ P_N \end{bmatrix}}$$

unde

$$\boxed{\mathbf{c}_N = \begin{bmatrix} -\mathbf{g}^{*} \\ 1 \end{bmatrix}}$$

4.2. PREDICŢIA ÎNAPOI (INVERSĂ)

Relaţie între coeficienţii filtrelor de predicţie directă şi inversă

$$\mathbf{R}\mathbf{g} = \mathbf{r}^{B*} \;\Rightarrow\; \mathbf{R}^{T}\mathbf{g}^{B} = \mathbf{r}^{*} \;\Rightarrow\; \mathbf{R}^{H}\mathbf{g}^{B*} = \mathbf{r} \;\Rightarrow\; \mathbf{R}\mathbf{g}^{B*} = \mathbf{r}$$

$$\mathbf{g}^{B*} = \mathbf{w} \;\Leftrightarrow\; \mathbf{w}^{B*} = \mathbf{g}$$

sau

$$g_k = w^{*}_{N-k+1}, \quad k = 1,\ldots,N$$

$$\mathbf{c}_{N} = \begin{bmatrix} -\mathbf{g}^{*} \\ 1 \end{bmatrix} = \begin{bmatrix} -\mathbf{w}^{B} \\ 1 \end{bmatrix} \qquad \mathbf{a}_{N} = \begin{bmatrix} 1 \\ -\mathbf{w}^{*} \end{bmatrix}$$

$$\mathbf{c}_N = \mathbf{a}_N^{B*} \qquad c$$

sau

$$c_{N,k} = a^{*}_{N,N-k}, \quad k = 0,\ldots,N$$

$$\mathbf{r}^{BT}\mathbf{g} = \mathbf{r}^{T}\mathbf{g}^{B} = \left(\mathbf{r}^{T}\mathbf{g}^{B}\right)^{*} = \mathbf{r}^{H}\mathbf{g}^{B*} = \mathbf{r}^{H}\mathbf{w} \quad \Rightarrow$$

puterile erorilor de predicţie sunt identice în cele două cazuri.

4.3 ALGORITMI EFICIENŢI DE REZOLVARE A ECUAŢIEI NORMALE A FILTRĂRII LINIARE OPTIMALE

Ecuaţia normală este un sistem $N \times N$
Rezolvat prin metoda Gauss rezultă o complexitate de tip $O(N^3)$
Algoritm rapid ar însemna $O(N)$

4.3.1 Algoritmul Levinson-Durbin

Vom căuta o constantă k_m, astfel încât să fie posibilă o relaţie de forma:

$$\mathbf{a}_m = \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} + k_m \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix}$$

sau

$$\mathbf{a}_m^* = \begin{bmatrix} \mathbf{a}_{m-1}^* \\ 0 \end{bmatrix} + k_m^* \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B} \end{bmatrix}$$

$$a_{m,k} = a_{m-1,k} + k_m a_{m-1,m-k}^*, \qquad k = 0,\ldots,m$$

$$a_{m-1,0} = 1, \qquad a_{m-1,m} = 0$$

4.3.1 Algoritmul Levinson-Durbin

$$\mathbf{a}_m^* = \begin{bmatrix} \mathbf{a}_{m-1}^* \\ 0 \end{bmatrix} + k_m^* \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B \end{bmatrix}$$

$$\mathbf{R}_{m+1}\mathbf{a}_m^* = \begin{bmatrix} P_m \\ \mathbf{0}_m \end{bmatrix}$$

$$\mathbf{R}_{m+1} = \begin{bmatrix} \mathbf{R}_m & \mathbf{r}_m^{B*} \\ \mathbf{r}_m^{BT} & r(0) \end{bmatrix}$$

$$\mathbf{R}_{m+1} \begin{bmatrix} \mathbf{a}_{m-1}^* \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_m & \mathbf{r}_m^{B*} \\ \mathbf{r}_m^{BT} & r(0) \end{bmatrix} \begin{bmatrix} \mathbf{a}_{m-1}^* \\ 0 \end{bmatrix} = \begin{bmatrix} \mathbf{R}_m \mathbf{a}_{m-1}^* \\ \mathbf{r}_m^{BT} \mathbf{a}_{m-1}^* \end{bmatrix}$$

4.3.1 Algoritmul Levinson-Durbin

$$= \begin{bmatrix} P_{m-1} \\ \mathbf{0}_{m-1} \end{bmatrix}$$

$$\mathbf{R}_{m+1}\begin{bmatrix}\mathbf{a}^*_{m-1} \\ 0\end{bmatrix} = \begin{bmatrix} \boxed{\mathbf{R}_m \mathbf{a}^*_{m-1}} \\ \boxed{\mathbf{r}_m^{BT} \mathbf{a}^*_{m-1}} \end{bmatrix}$$

$$= \Delta_{m-1}$$

Notăm

$$\Delta_{m-1} = \mathbf{r}_m^{BT}\mathbf{a}^*_{m-1} = \sum_{k=0}^{m-1} a^*_{m-1,k}\, r(k-m)$$

şi având în vedere forma extinsă a ecuaţiilor Wiener-Hopf pentru predictorul de ordin m-1,

$$\mathbf{R}_{m+1}\begin{bmatrix}\mathbf{a}^*_{m-1} \\ 0\end{bmatrix} = \begin{bmatrix} P_{m-1} \\ \mathbf{0}_{m-1} \\ \Delta_{m-1} \end{bmatrix}$$

4.3.1 Algoritmul Levinson-Durbin

$$\mathbf{a}_m^* = \begin{bmatrix} \mathbf{a}_{m-1}^* \\ 0 \end{bmatrix} + k_m^* \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B \end{bmatrix}$$

$$\mathbf{R}_{m+1} \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B \end{bmatrix} = \begin{bmatrix} r(0) & \mathbf{r}_m^H \\ \mathbf{r}_m & \mathbf{R}_m \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B \end{bmatrix} = \begin{bmatrix} \mathbf{r}_m^H \mathbf{a}_{m-1}^B \\ \mathbf{R}_m \mathbf{a}_{m-1}^B \end{bmatrix}$$

$$\mathbf{r}_m^H \mathbf{a}_{m-1}^B = \Delta_{m-1}^*, \quad \mathbf{R}_m \mathbf{a}_{m-1}^B = \mathbf{R}_m \mathbf{c}_{m-1}^* = \begin{bmatrix} \mathbf{0}_{m-1} \\ P_{m-1} \end{bmatrix}$$

$$\mathbf{R}_{m+1} \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^B \end{bmatrix} = \begin{bmatrix} \Delta_{m-1}^* \\ \mathbf{0}_{m-1} \\ P_{m-1} \end{bmatrix}$$

4.3.1 Algoritmul Levinson-Durbin

$$\begin{bmatrix} P_m \\ \mathbf{0}_m \end{bmatrix} = \begin{bmatrix} P_{m-1} \\ \mathbf{0}_{m-1} \\ \Delta_{m-1} \end{bmatrix} + k_m^* \begin{bmatrix} \Delta_{m-1}^* \\ \mathbf{0}_{m-1} \\ P_{m-1} \end{bmatrix}$$

$$P_m = P_{m-1} + k_m^* \Delta_{m-1}^* ; \quad 0 = \Delta_{m-1} + k_m^* P_{m-1}$$

4.3.1 Algoritmul Levinson-Durbin

$$P_m = P_{m-1} + k_m^* \Delta_{m-1}^* ; \quad 0 = \Delta_{m-1} + k_m^* P_{m-1}$$

$$P_m = P_{m-1}\left(1 - \left|k_m\right|^2\right)$$

$$P_m \geq 0 \quad \Rightarrow \quad \left|k_m\right| \leq 1$$

$$e_0^f(n) = e_0^b(n) = x(n); \qquad P_0 = r(0)$$

$$P_N = P_0 \prod_{m=1}^{N}\left(1 - \left|k_m\right|^2\right)$$

4.3.1 Algoritmul Levinson-Durbin

Observaţii

- Puterea erorii de predicţie scade o dată cu creşterea ordinului predictorului.
- Coeficienţii k_m - *coeficienţi de reflexie*

Pentru $k=m$,

$$k_m = a_{m,m}$$

- Se poate uşor arăta, folosind definiţiile şi principiul ortogonalităţii, că:

$$\Delta_{m-1} = \mathrm{E}\left\{e_{m-1}^{b}(n-1)e_{m-1}^{f\,*}(n)\right\}$$

unde $e_{m-1}^{f}(n)$ - răspunsul filtrului erorii de predicţie directă de ordinul m-1, pentru secvenţa de intrare $x(n), x(n-1), \ldots, x(n-m+1)$,

$e_{m-1}^{b}(n-1)$ - răspunsul filtrului erorii de predicţie inversă, de ordinul m-1, la secvenţa $x(n-1), x(n-2), \ldots, x(n-m)$.

4.3.1 Algoritmul Levinson-Durbin

Observaţii

- Având în vedere că:

$$e_0^f(n) = e_0^b(n) = x(n)$$

$$\Delta_0 = \mathrm{E}\left\{e_0^b(n-1)e_0^{f\,*}(n)\right\} = \mathrm{E}\left\{x(n-1)x^*(n)\right\} = r(-1) = r^*(1)$$

- Cunoscând Δ_o şi P_o se pot calcula, folosind relaţiile de recurenţă

$$k_1 = -\frac{\Delta_0^*}{P_0} = -\frac{r(1)}{r(0)}; \quad P_1 = r(0) - \frac{|r(1)|^2}{r(0)}$$

4.3.1 Algoritmul Levinson-Durbin

$$P_0 = r(0); \ \Delta_0 = r^*(1)$$

for $m = 1:1:N$

$$k_m = -\frac{\Delta_{m-1}^*}{P_{m-1}}$$

$$P_m = P_{m-1}\left(1 - |k_m|^2\right)$$

for $k = 1:1:m$

$$a_{m,k} = a_{m-1,k} + k_m a_{m-1,m-k}^*$$

end

$$\Delta_m = \sum_{k=0}^{m} a_{m,k}^* r(k-m-1)$$

end

4.3.1 Algoritmul Levinson-Durbin
Complexitatea aritmetică a algoritmului

În pasul *m* al algoritmului trebuie efectuate:

- o împărţire;
- $2m+1$ înmulţiri (două pentru calculul lui P_m, m-1 pentru reactualizarea coeficienţilor $a_{m,k}$, şi m pentru calculul lui Δ_m).
- $2m$ adunări.

Pentru efectuarea întregului algoritm, rezultă

$$\sum_{m=1}^{N}(2m+2) = N^2 + 3N$$

înmulţiri / împărţiri şi

$$\sum_{m=1}^{N}(2m) = N^2 + N$$

Dacă se utilizează o singură unitate aritmetică şi aceasta efectuează o operaţie aritmetică într-o unitate de timp, timpul total de calcul este $O(N^2)$.

Când se dispune de N unităţi aritmetice ce pot lucra în paralel timpul de calcul este $O(N \log_2 N)$

4.3.2 Algoritmul Schur -
definirea funcţiilor de tip '*f*'

Răspunsul filtrului erorii de predicţie de ordin m la secvenţa de autocorelaţie

$$y_m^f(i) = \sum_{k=0}^{m} a_{m,k} r_{xx}(i-k) = a_{m,i} * r_{xx}(i)$$

Conform ecuaţiei Wiener_Hopf pentru predictorul de ordin m

$$\mathbf{R}_{m+1}\mathbf{a}_m^* = \begin{bmatrix} P_m \\ \mathbf{0} \end{bmatrix}$$

Elementele matricei de autocorelaţie sunt

$$R_{i,k} = r_{xx}(k-i), \;\; k, i = 0,1,\cdots,m$$

4.3.2 Algoritmul Schur –

definirea funcţiilor de tip 'f'

$$\sum_{k=0}^{m} R_{i,k}\, a_{m,k}^{*} = 0, \quad \rightarrow \sum_{k=0}^{m} a_{m,k}^{*} r_{xx}(k-i) = 0, \quad i = 1, \cdots, m$$

$$\sum_{k=0}^{m} a_{m,k} r_{xx}(i-k) = 0, \quad i = 1, \ldots, m$$

$$y_m^f(i) = 0, \quad i = 1, \ldots, m$$

4.3.2 Algoritmul Schur – definirea funcţiilor de tip '*f*'

$$\sum_{k=0}^{m} R_{0,k}\, a_{m,k}^{*} = P_m\,, \quad \rightarrow \sum_{k=0}^{m} a_{m,k}^{*} r_{xx}(k) = P_m, \quad i = 0$$

rezultă că

$$y_m^f(0) = \sum_{k=0}^{m} a_{m,k} r_{xx}(-k) = P_m$$

4.3.2 Algoritmul Schur –

definirea funcţiilor de tip ‘*b*’

Pornind de la predicţia inversă

$$y_m^b(i) = \sum_{k=0}^{m} c_{m,k} r_{xx}(i-k) = c_{m,i} * r_{xx}(i)$$

Dar

$$c_{m,k} = a_{m,m-k}^{*}$$

aşa încât

$$y_m^b(i) = y_m^{f*}(m-i)$$

$$y_m^b(i) = 0, \quad i = 0,\ldots,m-1$$

$$y_m^b(m) = P_m$$

4.3.2 Algoritmul Schur –

relaţii de recurenţă

Din relaţiile

$$a_{m,k} = a_{m-1,k} + k_m a^*_{m-1,m-k} = a_{m-1,k} + k_m c_{m-1,k-1}$$

$$c_{m,k} = k^*_m a_{m-1,k} + c_{m-1,k-1}$$

se obţin nişte relaţii de recurenţă asemănătoare pentru aceste funcţii:

$$y^f_m(i) = y^f_{m-1}(i) + k_m y^b_{m-1}(i-1)$$

$$y^b_m(i) = k^*_m y^f_{m-1}(i) + y^b_{m-1}(i-1)$$

cu condiţiile iniţiale:

$$y^f_0(i) = y^b_0(i) = r_{xx}(i)$$

4.3.2 Algoritmul Schur – coeficientul de reflexie

În particular

$$y_m^f(m) = 0 \Rightarrow y_{m-1}^f(m) + k_m y_{m-1}^b(m-1) = 0$$

de unde

$$k_m = -\frac{y_{m-1}^f(m)}{y_{m-1}^b(m-1)}$$

4.3.2 Algoritmul Schur

for $k = 0 : N$

$$y_0^f(k) = y_0^b(k) = r_{xx}(k)$$

end

for $m = 1 : N$

$$k_m = -\frac{y_{m-1}^f(m)}{y_{m-1}^b(m-1)}$$

for $i = m+1 : N$

$$y_m^f(i) = y_{m-1}^f(i) + k_m y_{m-1}^b(i-1)$$

end

for $i = m : N$

$$y_m^b(i) = k_m^* y_{m-1}^f(i) + y_{m-1}^b(i-1)$$

end

end

$$P_N = y_N^b(N)$$

4.3.2 Algoritmul Schur –
complexitate aritmetică

În pasul m se efectuează:
o împărţire, pentru calculul lui k_m;
N-m înmulţiri şi N-m adunări, în primul ciclu după i;
N-m+1 înmulţiri şi N-m+1 adunări, în al doilea ciclu după i.
Rezultă un număr de $2N$-$2m$+2 înmulţiri / împărţiri şi $2N$-$2m$+1 adunări, deci în total

$$\sum_{m=1}^{N}(2N-2m+2)=N^2+N$$

înmulţiri / împărţiri şi

$$\sum_{m=1}^{N}(2N-2m+1)=N^2$$

adunări.

4.3.2 Algoritmul Schur –
Modalitate practică de calcul

Iniţializarea algoritmului. Se constituie “matricea generatoare”

$$\mathbf{G}_0 = \begin{bmatrix} 0 & y_0^f(1) & \cdots & y_0^f(N) \\ y_0^b(0) & y_0^b(1) & \cdots & y_0^b(N) \end{bmatrix}$$

sau

$$\mathbf{G}_0 = \begin{bmatrix} 0 & r_{xx}(1) & r_{xx}(2) & \ldots & r_{xx}(N) \\ r_{xx}(0) & r_{xx}(1) & r_{xx}(2) & \ldots & r_{xx}(N) \end{bmatrix}$$

4.3.2 Algoritmul Schur – Modalitate practică de calcul

Prin deplasarea spre dreapta cu o unitate a liniei a doua se obţine

$$\mathbf{G}_0' = \begin{bmatrix} 0 & y_0^f(1) & \cdots & y_0^f(N) \\ 0 & y_0^b(0) & \cdots & y_0^b(N-1) \end{bmatrix}$$

sau

$$\mathbf{G'}_0 = \begin{bmatrix} 0 & r_{xx}(1) & r_{xx}(2) & \dots & r_{xx}(N) \\ 0 & r_{xx}(0) & r_{xx}(1) & \dots & r_{xx}(N-1) \end{bmatrix}$$

Raportul elementelor coloanei a doua, cu semnul schimbat, determină primul coeficient de reflexie, k_1 .

$$k_1 = -\frac{y_0^f(1)}{y_0^b(0)}$$

4.3.2 Algoritmul Schur –
Modalitate practică de calcul

Se constituie matricea

$$\mathbf{K}_1 = \begin{bmatrix} 1 & k_1 \\ k_1^* & 1 \end{bmatrix}$$

Se calculează

$$\mathbf{K}_1\mathbf{G}'_0 = \begin{bmatrix} 1 & k_1 \\ k_1^* & 1 \end{bmatrix} \begin{bmatrix} 0 & y_0^f(1) & \cdots & y_0^f(N) \\ 0 & y_0^b(0) & \cdots & y_0^b(N-1) \end{bmatrix}$$

$$\mathbf{G}_1 = \mathbf{K}_1\mathbf{G}'_0 = \begin{bmatrix} 0 & 0 & y_1^f(2) & \ldots & y_1^f(N) \\ 0 & y_1^b(1) & y_1^b(2) & \ldots & y_1^b(N) \end{bmatrix}$$

unde s-au avut în vedere relaţiile de recurenţă şi faptul că $y_1^f(1) = 0$.

4.3.2 Algoritmul Schur –
Modalitate practică de calcul

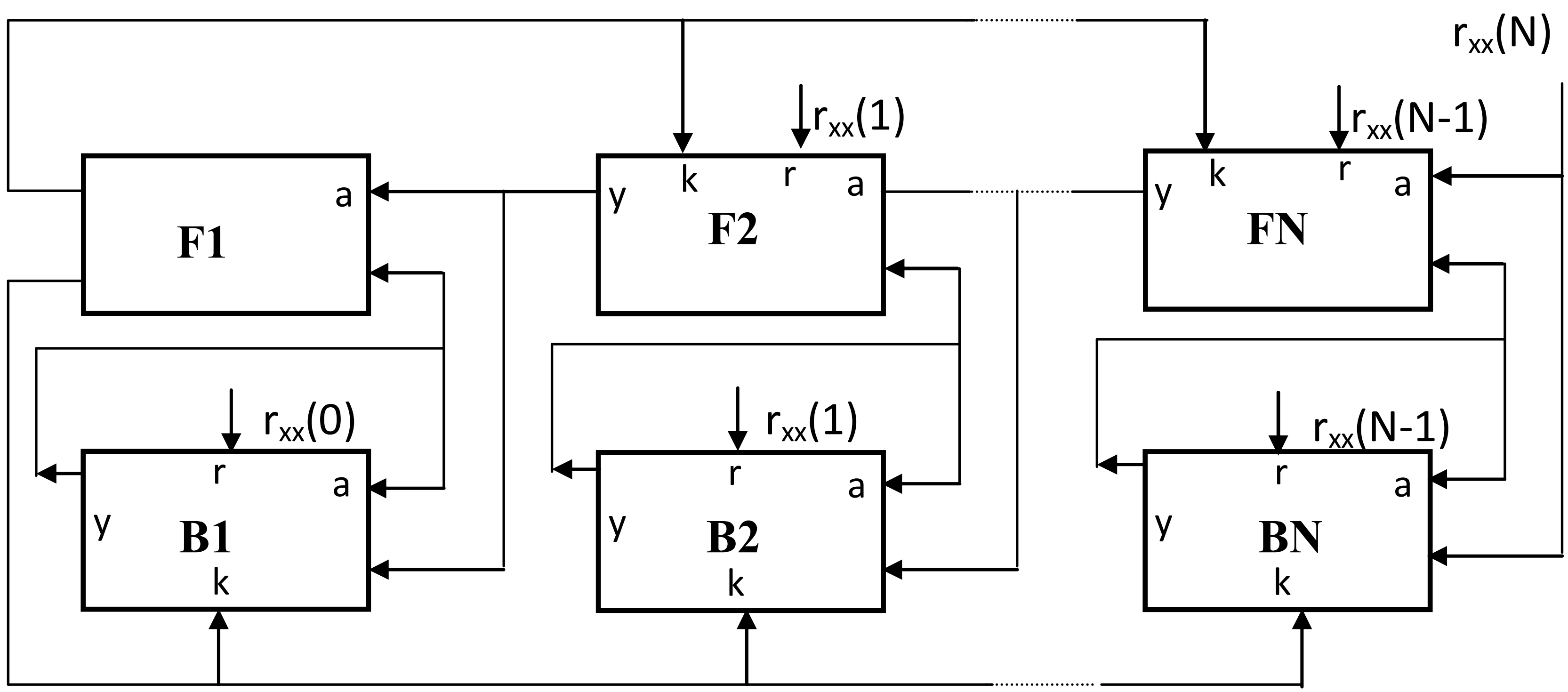
În continuare se repetă ultimele trei operaţii, deci în pasul m:

- Se formează $\mathbf{G}'_m$ din $\mathbf{G}_m$ prin deplasarea spre dreapta cu o poziţie a liniei a doua.
- Se calculează k_{m+1} ca raport cu semnul schimbat a elementelor coloanei m+2 şi se formează matricea $\mathbf{K}_{m+1}$.
- Se calculează

$$\mathbf{G}_{m+1} = \mathbf{K}_{m+1}\mathbf{G}'_m$$

Procedeul se reia până se calculează toţi coeficienţii de reflexie.

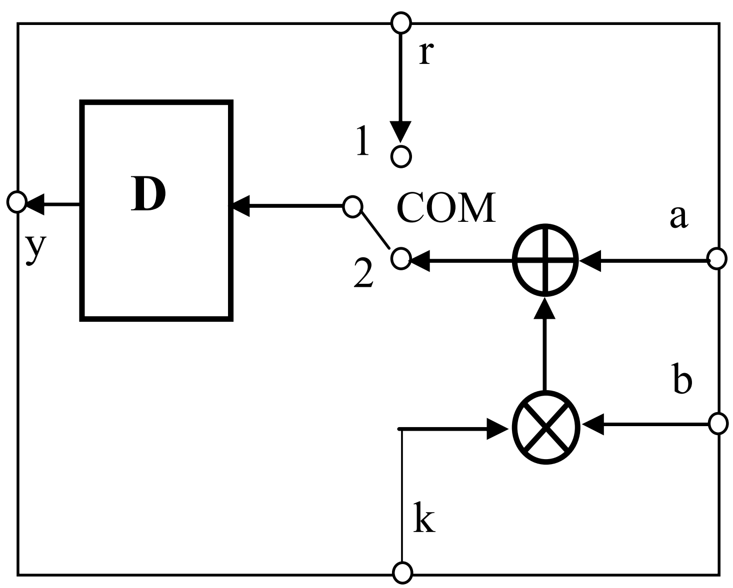
4.3.2 Algoritmul Schur –
Variantă paralel de implementare a algoritmului Schur



4.3.2 Algoritmul Schur –

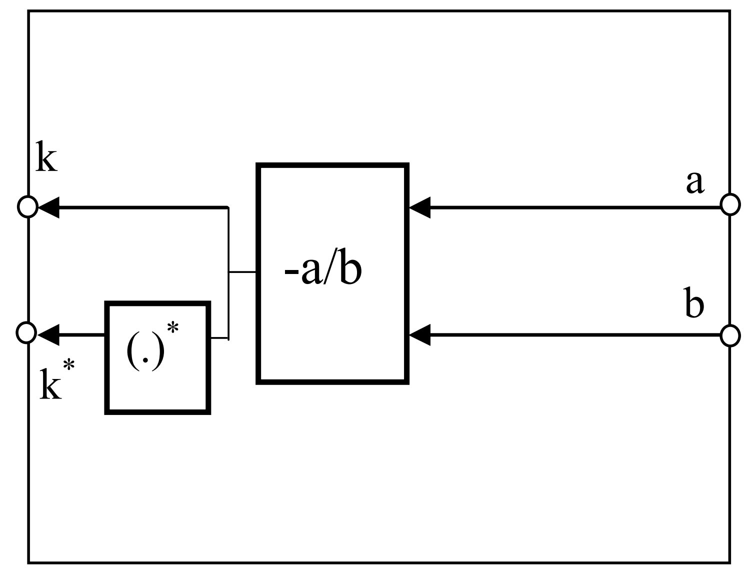
Variantă paralel de implementare a algoritmului Schur

F2,…,FN,B1,…,BN



a.

F1



b.

4.4 FILTRELE ERORII DE PREDICŢIE

4.4.1 Proprietăţi ale filtrelor erorii de predicţie

Vom nota funcţiile de transfer ale filtrelor erorii de predicţie directă şi inversă de ordinul m cu:

$$H_m^f(z) = \sum_{k=0}^{m} a_{m,k} z^{-k}; \quad H_m^b(z) = \sum_{k=0}^{m} c_{m,k} z^{-k} = \sum_{k=0}^{m} a_{m,m-k}^{*} z^{-k}$$

1) Între funcţiile de transfer ale celor două filtre există relaţia

$$H_m^b(z) = z^{-m} H_m^{f*}\left(\frac{1}{z^*}\right)$$

2) ***Caracteristicile amplitudine-frecvenţă*** ale celor două filtre sunt identice.

$$H_m^b\left(e^{j\omega}\right) = e^{-jm\omega}\, H_m^{f*}\left(e^{j\omega}\right) \;\Rightarrow\; \left|H_m^b\left(e^{j\omega}\right)\right| = \left|H_m^f\left(e^{j\omega}\right)\right|$$

4.4 FILTRELE ERORII DE PREDICŢIE

4.4.1 Proprietăţi ale filtrelor erorii de predicţie

3) ***Relaţie de recurenţă***. Având în vedere formulele

$$a_{m,k} = a_{m-1,k} + k_m a^*_{m-1,m-k}, \qquad k = 0,\ldots,m$$

rezultă:

$$H_m^f(z) = H_{m-1}^f(z) + k_m z^{-1} H_{m-1}^b(z)$$

4.4 FILTRELE ERORII DE PREDICŢIE

4.4.1 Proprietăţi ale filtrelor erorii de predicţie

4) ***Filtrul erorii de predicţie directă este de fază minimă***. Pentru demonstraţie vom avea în vedere faptul că $|k_m|<1$, ceea ce, ţinând seama şi de proprietatea 2 conduce la:

$$\left|k_m z^{-1} H_{m-1}^{b}(z)\right| < \left|H_{m-1}^{b}(z)\right| = \left|H_{m-1}^{f}(z)\right| \quad \text{pe} \quad |\mathrm{z}| = 1$$

***Teorema lui Rouché*:**

Dacă două funcţii $F(z)$, $G(z)$, sunt analitice pe conturul C din planul z şi în interiorul conturului, şi $|G(z)|<|F(z)|$ pe contur, atunci funcţia $F(z)+G(z)$ are acelaşi număr de zerouri în interiorul conturului C ca şi $F(z)$.

4.4 FILTRELE ERORII DE PREDICŢIE

4.4.1 Proprietăţi ale filtrelor erorii de predicţie

Fie conturul C cercul $|z|=1$ şi vom aplica teorema aceasta pentru funcţiile:

$$F(z)=H_{m-1}^{f}(z),\ \ G(z)=k_m z^{-1}H_{m-1}^{b}(z)$$

C în sens direct trigonometric $\Rightarrow$ $\operatorname{int}\{C\} = \{|z|<1\}$

C în sens invers trigonometric $\Rightarrow$ $\operatorname{int}\{C\} = \{|z|>1\}$

Conform teoremei enunţate, dacă $H_{m-1}^{f}(z)$ nu are zerouri în $|z|>1$,

de aceeaşi proprietate se bucură şi $H_m^f(z)$. Cum $H_0^f(z)=1$, rezultă

prin inducţie completă că $H_m^f(z)$ nu are zerouri în afara cercului $|z|=1$.

4.4 FILTRELE ERORII DE PREDICŢIE

4.4.1 Proprietăţi ale filtrelor erorii de predicţie

5) ***Filtrul erorii de predicţie inversă este de fază maximă*** (are toate zerourile în exteriorul cercului $|z|=1$).

Dacă se exprimă $H_m^f(z)$ sub forma

$$H_m^f(z)=\prod_{i=1}^{m}\left(1-z_i z^{-1}\right), \qquad |z_i|<1$$

având în vedere proprietatea 1, rezultă

$$H_m^b(z)=z^{-m}\prod_{i=1}^{m}\left(1-z_i^* z\right)=\prod_{i=1}^{m}\left(z^{-1}-z_i^*\right)$$

Nulurile acestei funcţii de transfer sunt de forma $1/z_i^*$, $i=1,\ldots,m$, şi sunt evident situate în $|z|>1$, simetric faţă de z_i în raport cu cercul $|z|=1$.

4.4 FILTRELE ERORII DE PREDICȚIE

4.4.2 Forma "latice" de realizare a filtrului erorii de predicție

Relații de recurență pentru eroarea de predicție

$$e_m^f(n) = \mathbf{a}_m^T \mathbf{x}_{m+1}(n) = \left(\begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} + k_m \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} \right)^T \mathbf{x}_{m+1}(n) =$$

$$= \left[\mathbf{a}_{m-1}^T, 0\right] \begin{bmatrix} \mathbf{x}_m(n) \\ x(n-m) \end{bmatrix} + k_m \left[0, \mathbf{a}_{m-1}^{BH}\right] \begin{bmatrix} x(n) \\ \mathbf{x}_m(n-1) \end{bmatrix} =$$

$$= \mathbf{a}_{m-1}^T \mathbf{x}_m(n) + k_m \boxed{\mathbf{a}_{m-1}^{BH} \mathbf{x}_m(n-1)}$$

$$\boxed{\mathbf{a}_{m-1}^{BH} \mathbf{x}_m(n-1) = \mathbf{c}_{m-1}^T \mathbf{x}_m(n-1) = e_{m-1}^b(n-1)}$$

$$e_m^f(n) = e_{m-1}^f(n) + k_m e_{m-1}^b(n-1)$$

4.4 FILTRELE ERORII DE PREDICŢIE

4.4.2 Forma "latice" de realizare a filtrului erorii de predicţie

În mod asemănător se găseşte:

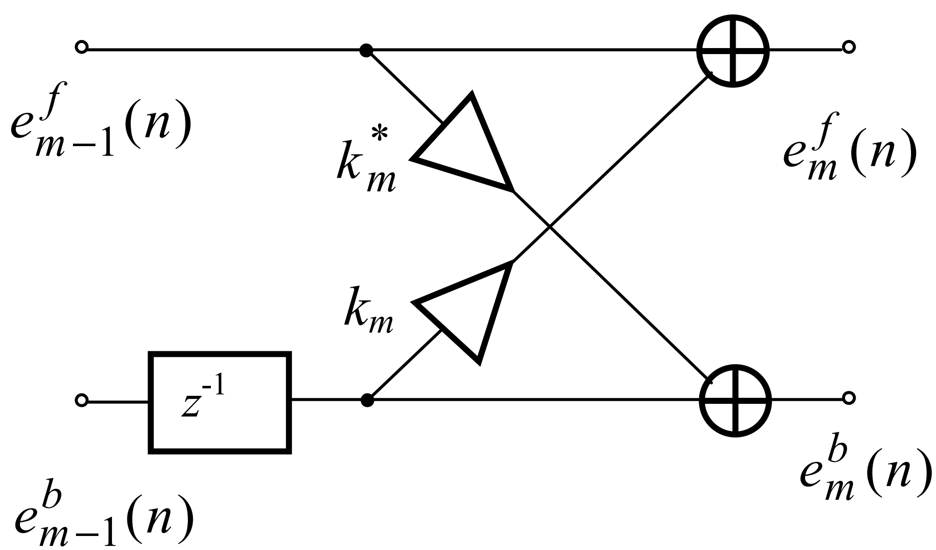
$$e_m^b(n) = e_{m-1}^b(n-1) + k_m^* e_{m-1}^f(n-1)$$

Cele două relaţii pot fi grupate

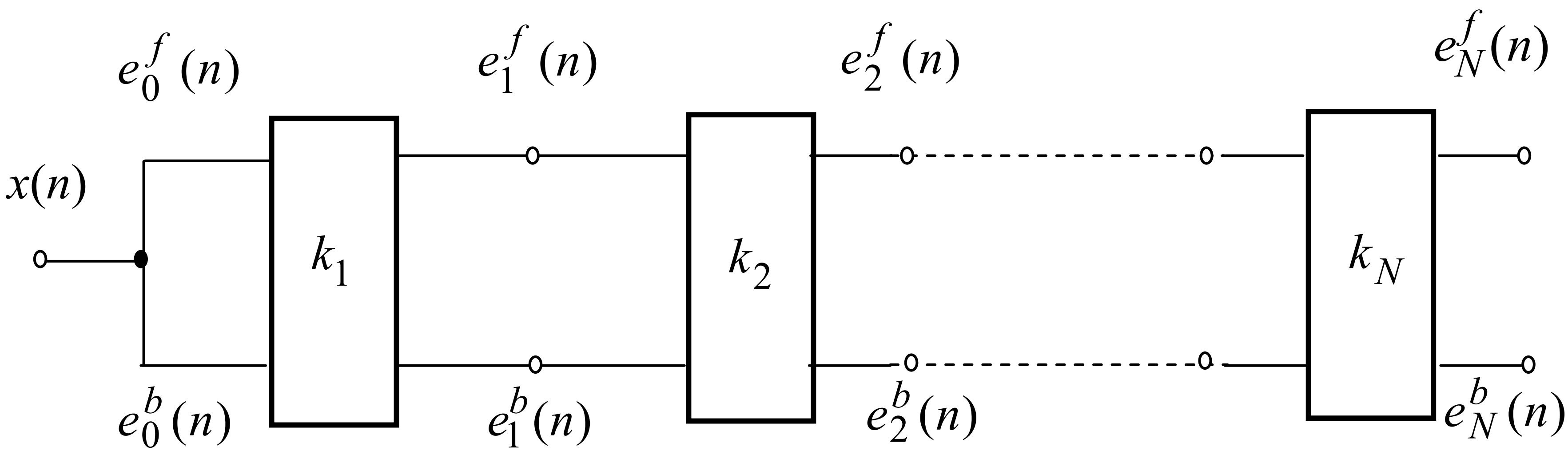
$$\begin{bmatrix} e_m^f(n) \\ e_m^b(n) \end{bmatrix} = \begin{bmatrix} 1 & k_m \\ k_m^* & 1 \end{bmatrix} \begin{bmatrix} e_{m-1}^f(n) \\ e_{m-1}^b(n-1) \end{bmatrix}$$

4.4 FILTRELE ERORII DE PREDICȚIE

$$\begin{bmatrix} e_m^f(n) \\ e_m^b(n) \end{bmatrix} = \begin{bmatrix} 1 & k_m \\ k_m^* & 1 \end{bmatrix} \begin{bmatrix} e_{m-1}^f(n) \\ e_{m-1}^b(n-1) \end{bmatrix}$$



4.4 FILTRELE ERORII DE PREDICȚIE



$e_0^f(n) = e_0^b(n) = x(n)$ - intrarea filtrului

4.4 FILTRELE ERORII DE PREDICŢIE

Se poate pune problema calculului coeficienţilor k_k, cunoscând $a_{N,1},...,a_{N,N}$. Pentru aceasta vom porni de la ecuaţia:

$$a_{m,k} = a_{m-1,k} + k_m a^*_{m-1,m-k}, \qquad k = 0,1,..., m$$

căreia îi vom alătura ecuaţia obţinută din aceasta prin conjugare complexă şi înlocuind k cu m-k. Rezultă sistemul:

$$\begin{aligned} a_{m,k} &= a_{m-1,k} + k_m a^*_{m-1,m-k} \\ a^*_{m,m-k} &= k^*_m a_{m-1,k} + a^*_{m-1,m-k} \end{aligned}$$

din care se calculează $a_{m-1,k}$,

$$a_{m-1,k} = \frac{a_{m,k} - k_m a^*_{m,m-k}}{1-|k_m|^2} = \frac{a_{m,k} - a_{m,m} a^*_{m,m-k}}{1-|a_{m,m}|^2}$$

4.4 FILTRELE ERORII DE PREDICȚIE

Cunoscând setul de coeficienţi corespunzători predictorului de ordin N, $\{a_{N,k}\}$, se trece, cu formula de mai sus, la cei corespunzători predictorului de ordin N-1, $\{a_{N-1,k}\}$ şi se determină $k_{m-1}=a_{m-1,m-1}$, şi aşa mai departe până la N=1.

4.4 FILTRELE ERORII DE PREDICŢIE

$$\text{for } j = 0:1:N$$

$$a_{N,j} = a_j$$

$$\text{end}$$

$$\text{for } i = N:\text{-}1:2$$

$$k_i = a_{i,i}$$

$$\text{for } j = 1:1:i\text{-}1$$

$$a_{i-1,j} = \frac{a_{i,j} - k_i a^*_{i,i-j}}{1 - |k|^2_i}$$

$$\text{end}$$

$$\text{end}$$

$$k_1 = a_{1,1}$$

4.4 FILTRELE ERORII DE PREDICŢIE

Cunoscând setul de coeficienţi corespunzători predictorului de ordin N, $\{a_{N,k}\}$, se trece, cu formula de mai sus, la cei corespunzători predictorului de ordin N-1, $\{a_{N-1,k}\}$ şi se determină $k_{m-1}=a_{m-1,m-1}$, şi aşa mai departe până la N=1.

Problema inversă – se cunosc coeficienţii de reflexie (de exemplu s-au determinat cu alggoritmul Schur) şi se doresc coeficienţii filtrului erorii de predicţie în forma tranversală.

Se pormeşte în sens invers, de la filtrul de ordinul 1, pentru care se cunosc $a_{1,0} = 1$, $a_{1,1} = k_1$ şi se măreşte succesiv ordinul.

4.4 FILTRELE ERORII DE PREDICŢIE

$$a_{1,1} = k_1, \quad a_{1,0} = 1$$

$$\text{for } i = 2:1:N$$

$$a_{i,i} = k_i, \quad a_{i,0} = 1$$

$$\text{for } j = 1:1:i\text{-}1$$

$$a_{i,j} = a_{i-1,j} + k_i a^*_{i-1,i-j}$$

$$\text{end}$$

$$\text{end}$$

$$\text{for } j = 0:1:N$$

$$a_j = a_{N,j}$$

$$\text{end}$$

4.4 FILTRELE ERORII DE PREDICŢIE

O proprietate remarcabilă a structurii latice este aceea că se poate mări ordinul predictorului, adăugând pur şi simplu încă o celulă, fără a modifica în rest structura existentă. Faptul că toate celulele au aceeaşi structură este favorabil din punctul de vedere al posibilităţilor de integrare pe scară foarte largă.