Facultatea de Inginerie Electrică, Energetică şi Informatică Aplicată (IEEIA)

# Identificarea şi Modelarea Sistemelor

**C10**

**Prof.univ.dr.ing. Marian-Silviu Poboroniuc**

**Reamintire:** Estimatorul celor mai mici pătrate se obţine din ecuaţiile :

$$\hat{\theta}_{CMMP} = \hat{\theta}_{LS}$$

$$V(\theta) = \frac{1}{N}\sum_{k=1}^{N}(y(k) - \varphi^T(k)\theta)^2$$

$$\nabla V(\theta) = \frac{\partial V(\theta)}{\partial \theta} = 2\frac{1}{N}\sum_{k=1}^{N}(-\varphi(k))\left(y(k) - \varphi^T(k)\theta\right) = 0$$

$$\nabla^2 V(\theta) = \frac{\partial^2 V(\theta)}{\partial \theta^2} = 2\frac{1}{N}\sum_{k=1}^{N}\left(\varphi(k)\right)\varphi^T(k) > 0$$

şi este dat de relaţia

$$\hat{\theta}_{LS} = \left(\frac{1}{N}\sum_{k=1}^{N}\varphi(k)\varphi^T(k)\right)^{-1}\left(\frac{1}{N}\sum_{k=1}^{N}\varphi(k)y(k)\right)$$

**Reamintire:** Iesirea era :

$$y(k) = \varphi^T(k)\theta + v(k)$$

*de unde*

$$\varphi(k)y(k) = \varphi(k)\varphi^T(k)\theta + \varphi(k)v(k)$$

$$\frac{1}{N}\sum_{k=1}^{N}\varphi(k)y(k) = \frac{1}{N}\sum_{k=1}^{N}\varphi(k)\varphi^T(k)\theta + \frac{1}{N}\sum_{k=1}^{N}\varphi(k)v(k)$$

De unde doar daca $\varphi(k)$ si $v(k)$ sunt necorelate, rezulta (pentru ca ultima suma sa fie egala cu zero):

$$\hat{\theta}_{LS} = \left(\frac{1}{N}\sum_{k=1}^{N}\varphi(k)\varphi^T(k)\right)^{-1}\left(\frac{1}{N}\sum_{k=1}^{N}\varphi(k)y(k)\right)$$

## 5.5. Metode de variabilă instrumentală
## 5.5.1. Principiul metodei de variabilă instrumentală

Estimatorul celor mai mici pătrate (CMMP) dedus in metoda anterioara:

$$\hat{\theta}_{LS} = [\frac{1}{N}\sum_{k=1}^{N}\varphi(k)\varphi^T(k)]^{-1}[\frac{1}{N}\sum_{k=1}^{N}\varphi(k)y(k)]$$

este consistent dacă sunt îndeplinite condiţiile:

$$R = E[\varphi(k)\varphi^T(k)] > 0;$$
$$E\left[\varphi(k)v(k)\right] = 0$$

Prima condiţie este asigurată, în general, prin ***persistenţa semnalului de intrare***. A doua condiţie este îndeplinită numai dacă *v(k)* este zgomot alb, si constituie o limitare a metodei.

Variabilele instrumentale sunt introduse tocmai în idea de a obţine un estimator similar celor mai mici pătrate care să fie consistent indiferent de natura perturbaţiei. Metoda de variabilă instrumentală utilizează estimatorul:

$$\hat{\theta}_{VI} = \left[ \frac{1}{N} \sum_{k=1}^{N} z(k) \varphi^T(k) \right]^{-1} \left[ \frac{1}{N} \sum_{k=1}^{N} z(k) y(k) \right]$$

în care φ(k) are aceeaşi semnificaţie ca în cazul metodei CMMP ,

$$\varphi(k) = \left[ -y(k-1), -y(k-2), \ldots\ldots, -y(k-na), u(k-1), \ldots\ldots, u(k-nb) \right]^T$$

iar *z(k)*, de dimensiune $\dim z(k) = na + nb$ este un vector oarecare ale cărui componente, numite **variabile instrumentale**, trebuie alese în aşa fel încât estimatorul să fie consistent

Înlocuind $y(k)=\varphi^T(k)\theta^*+v(k)$ în relatia estimatorului, se obţine

$$\hat{\theta}_{VI}=\left[\frac{1}{N}\sum_{k=1}^{N}z(k)\varphi^T(k)\right]^{-1}\frac{1}{N}\sum_{k=1}^{N}z(k)\varphi^T(k)\theta^*+$$
$$+\left[\frac{1}{N}\sum_{k=1}^{N}z(k)\varphi^T(k)\right]^{-1}\frac{1}{N}\sum_{k=1}^{N}z(k)v(k)=\qquad(5.77)$$
$$=\theta^*+\left[\frac{1}{N}\sum_{k=1}^{N}z(k)\varphi^T(k)\right]^{-1}\frac{1}{N}\sum_{k=1}^{N}z(k)v(k)$$

Din (5.77) rezultă condiţiile de consistenţă a estimatorului VI:

$$R=E[z(k)\varphi^T(k)]=\frac{1}{N}\sum_{k=1}^{N}z(k)\varphi^T(k)>0\qquad(5.78)$$

$$E\left[z(k)v(k)\right]=\frac{1}{N}\sum_{k=1}^{N}z(k)v(k)=0\qquad(5.79)$$

Variabilele instrumentale pot fi corelate cu intrările şi ieşirile, dar nu sunt corelate cu perturbaţiile. Pentru a satisface cerinţele (5.78), (5.79) foarte ***frecvent se aleg ca variabile de stare eşantioanele întârziate ale intrării***. La întârzieri mai mari condiţiile sunt mai bine satisfăcute .

Ca şi estimatorul CMMP şi estimatorul VI este insensibil la o transformare liniară. Această proprietate se poate utiliza la construirea vectorilor de variabilă instrumentală. Într-adevăr, dacă înlocuim *z(k)* cu *Tz(k)* , unde *T* este o matrice $n\theta$ x $n\theta$ de transformare nesingulară, estimatorul devine

$$T\hat{\theta}_{VI} = \left[\frac{1}{N}\sum_{k=1}^{N} Tz(k)\varphi^T(k)\right]^{-1}\frac{1}{N}\sum_{k=1}^{N} Tz(k)y(k) =$$

$$= \left[\frac{1}{N}\sum_{k=1}^{N} z(k)\varphi^T(k)\right]^{-1} T^{-1}\cdot T\,\frac{1}{N}\sum_{k=1}^{N} z(k)y(k) =$$

$$= \left[\frac{1}{N}\sum_{k=1}^{N} z(k)\varphi^T(k)\right]^{-1}\frac{1}{N}\sum_{k=1}^{N} z(k)y(k) = \hat{\theta}_{VI}$$

## 5.5.2. Alegerea variabilelor instrumentale de bază

Se consideră o formă generală a vectorului de variabilă instrumentală

$$z(k) = K(q^{-1})\left[-\eta(k-1),\ldots,-\eta(k-na),u(k-1),\ldots.,u(k-nb)\right]^T \qquad (5.80)$$

unde $\eta(k)$ este obţinut prin filtrarea datelor de intrare:

$$\eta(k) = \frac{C(q^{-1})}{D(q^{-1})} u(k)$$
$$C(q^{-1}) = c_0 + c_1 q^{-1} + .... + c_{nc} q^{-nc}$$
$$D(q^{-1}) = 1 + d_1 q^{-1} + .... + d_{nd} q^{-nd} \qquad (5.81)$$

$K(q^{-1})$ şi $K^{-1}(q^{-1})$ sunt filtre asimptotic stabile, iar polinoamele $C(q^{-1})$ şi $D(q^{-1})$ au zerourile în afara cercului unitar şi sunt prime între ele.

Există o mare varietate de variabile instrumentale pentru cazuri particulare de forme ale filtrului $K(q^{-1})$ şi polinoamelor $C(q^{-1})$ şi $D(q^{-1})$.

Pentru fiecare din aceste cazuri se determină parametrii astfel ca relaţiile (5.78), (5.79) să fie îndeplinite. Dacă de exemplu $nc=nd=na$ relaţia (5.80) devine:

$$
\begin{aligned}
z(k) &= \frac{K(q^{-1})}{D(q^{-1})}[-C(q^{-1})u(k-1),\ldots,-C(q^{-1})u(k-na), D(q^{-1})u(k-1)\ldots \\
&\ldots D(q^{-1})u(k-nb)]^T = S(-C,D)\frac{K(q^{-1})}{D(q^{-1})}[u(k-1),u(k-2),\ldots,u(k-na-nb)]^T
\end{aligned}
\tag{5.82}
$$

unde *S(-C, D)* este matricea Sylvester asociată polinoamelor *-C* şi *D* :

$$
S(-C,D) = \underbrace{\begin{bmatrix}
-c_0 & -c_1 \ldots & -c_{na} & \overbrace{0 \quad 0 \ldots. \quad 0 \quad 0}^{n_b} \\
0 & -c_0 \ldots & -c_{na-1} & -c_{na} \quad 0\ldots. \quad 0 \quad 0 \\
0 & 0 \ \ldots & -c_0 & -c_1 \qquad\qquad -c_{na} \\
 & & & \\
1 & \mathrm{d}_1\ldots & d_{na} & 0 \qquad\qquad 0 \quad 0 \\
0 & 1 & \mathrm{d}_1\ldots & d_{na-1} \quad d_{na} \,..\, 0 \quad 0 \\
\ldots\ldots & \ldots\ldots & \ldots\ldots & \ldots\ldots\ldots\ldots\ldots\ldots \\
0 & 0\ldots & 1 & d_1 \ldots \qquad\qquad d_{na}
\end{bmatrix}}_{n_a+n_b}
\begin{matrix} \left.\vphantom{\begin{matrix}a\\a\\a\\a\end{matrix}}\right\} n_b \\ \left.\vphantom{\begin{matrix}a\\a\\a\\a\end{matrix}}\right\} n_a \end{matrix}
$$

Dacă $C(q^{-1})$ şi $D(q^{-1})$ sunt polinoame prime între ele rezultă că rangul matricei *S(-C, D)* este egal cu suma *na* + *nb*, deci matricea *S* este nesingulară.

## Estimarea parametrilor prin metoda variabilelor instrumentale

În MATLAB modelul este de forma unei ecuaţii cu diferenţe:

$$A(q^{-1})y(k) = B(q^{-1})u(k-ni) + e(k)$$

Este necesar să se definească matricea de date intrare-ieşire **z=[*y, u*]** sau *data* şi să se cunoască structura definită prin vectorul **nn= [*na nb ni* ],** în care *na* şi *nb* sunt gradele polinoamelor *A* şi respectiv *B*, iar *ni* este întârzierea. Modelul *m* obţinut este în forma polinomială IDPOLY pentru sistemele cu o singură ieşire sau în forma IDARX pentru sisteme cu mai multe ieşiri, sau sub forma THETA. Se utilizează **funcţia *iv4()*** care este apelată (una dintre forme):

*m =iv4(z,nn)*;

**Exemplu:** Se consideră sistemul monovariabil discret definit prin polinoamele **A=[1 -1.2 0.6]; B=[0 1 0.5];** pentru partea deterministă şi **C=[1 -0.8 0.4]** pentru caracterizarea perturbaţiilor. Se asociază acestui sistem modelul *m*

```
>> m=idpoly(A,B,C).

Discrete-time IDPOLY model: A(q)y(t) = B(q)u(t) + C(q)e(t) ;t=k
A(q) = 1 - 1.2 q^-1 + 0.6 q^-2
B(q) = q^-1 + 0.5 q^-2
C(q) = 1 - 0.8 q^-1 + 0.4 q^-2
```

Se generează un semnal de intrare pseudoaleator binar *u(k)* şi un semnal perturbator aleator *e(k):*

```
>> u=sign(randn(500,1));
>> e=0.1*randn(500,1);
```

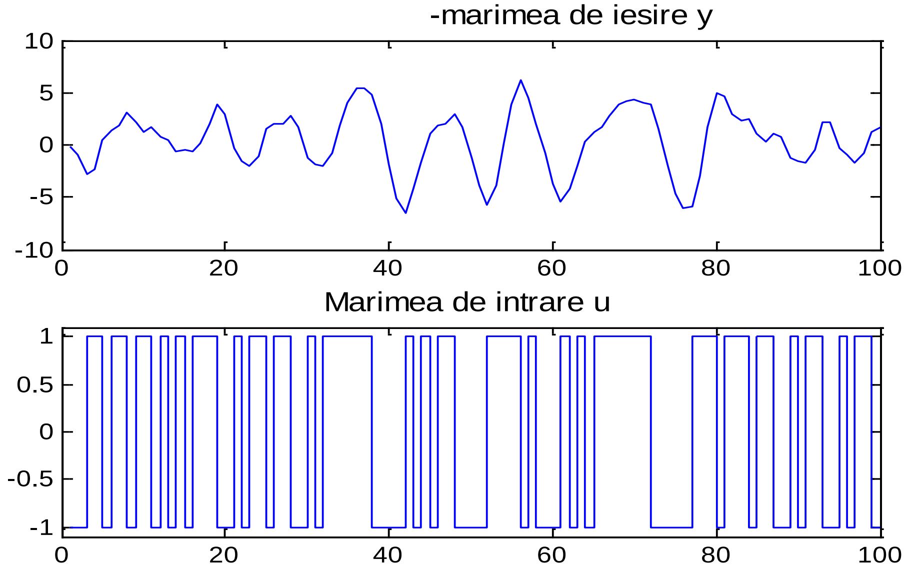
Se determină prin simulare numerică răspunsul sistemului *y(k)* pentru aceste semnale de intrare

```
>>y=idsim([u,v],m);
```

Se consideră o matrice *z* cu vectori coloană conţinând datele de ieşire şi de intrare, eşantioanele cu numerele de la 1 la 300.

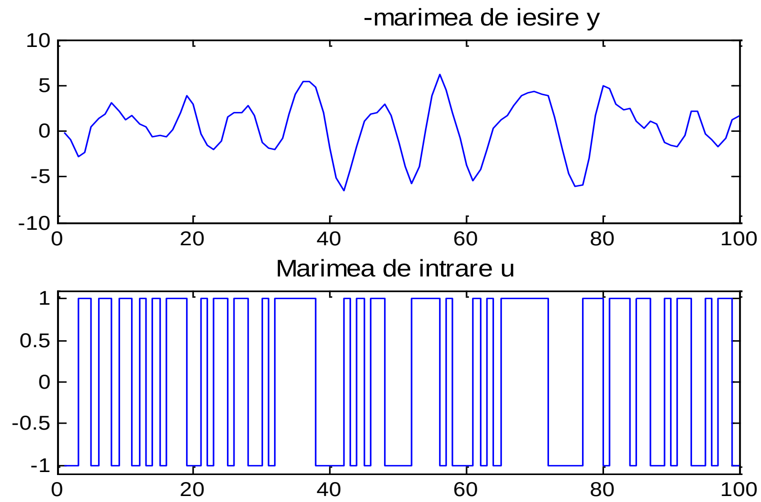
```
>>z=[y(1:300) u(1:300)];
```

- Se reprezintă grafic primele 100 de eşantioane de intrare-ieşire, prezentate în fig.5.3
- >>idplot(z, 1:100)




Prin metoda VI se caută un model cu doi poli, *na*=2, cu două zerouri , *nb*=2 şi întârzierea *ni*=1. Cu funcţia *iv4* se obţine modelul *thi*, care este afişat cu funcţia *present*.

```
>>nn=[2 2 1]
>>thi=iv4(z,nn) ; >>present(thi)
```




A(q) = 1 - 1.199 (+-0.00315) q^-1 + 0.5999 (+-0.002534) q^-2
B(q) = 0.9942 (+-0.005995) q^-1 + 0.4983 (+-0.00876) q^-2
Estimated using IV4 from data set z
Loss function 0.0107966 and FPE 0.0110884

Sunt prezentate valorile coeficienţilor polinoamelor şi abaterile lor standard. Cu funcţia ***resid*** se determină reziduurile ***ei*** ale modelului şi se reprezintă grafic în fig. 5.9.

```
>>ei=resid(z,thi);
```

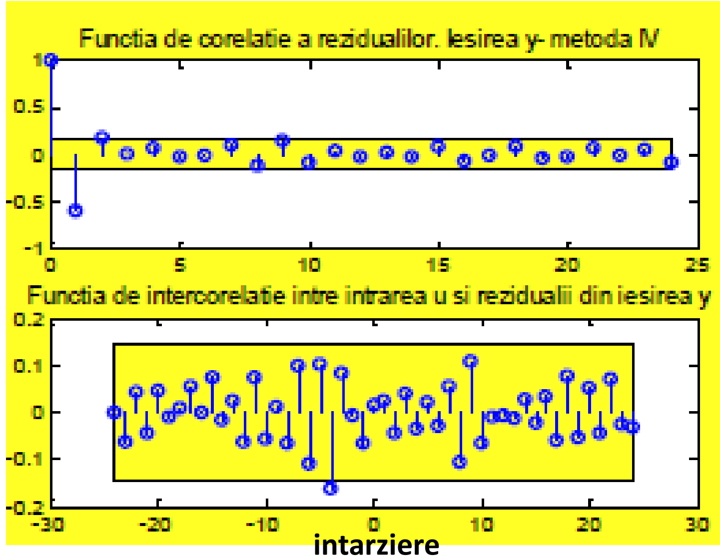



Fig. 5.9

Se reprezintă grafic răspunsul sistemului, *y* şi a modelului IV4, *ym,* pentru primele 50 de eşantioane, în fig.5.10

```
>>plot(t1,y(1:50),'b',t1,ym(1:50),'g');grid
>>title('Răspuns sistem y, răspuns model – ym. Metoda IV4')
```

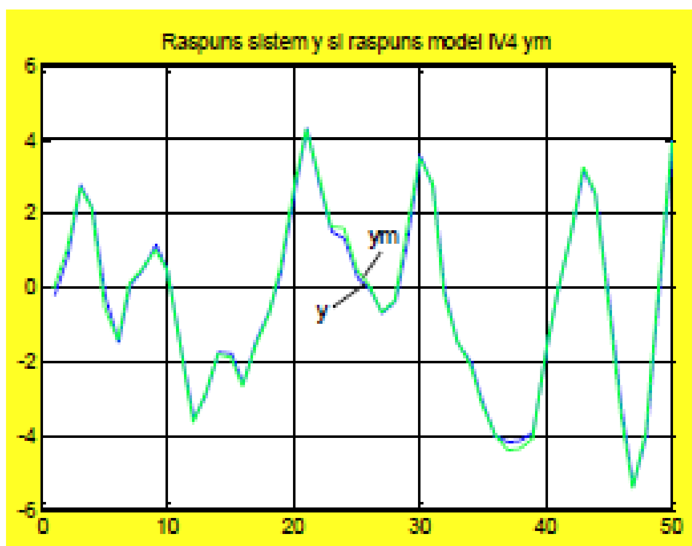



Fig. 5.10

# Metode recursive de identificare parametrică a sistemelor

- **Metodele de identificare** a parametrilor de tip **off-line** prezentate anterior prelucrează simultan întregul şir de măsurători intrare-ieşire.

- În cazul **metodelor de identificare recursive** sau **on-line** datele sunt prelucrate pe măsură ce ele devin disponibile prin măsurători.

- Procedurile **on-line** se impun în următoarele situaţii:
  - A-când se doreşte continuarea experimentului de identificare până la obţinerea unei precizii aprioric impuse;
  - B-când sistemul are parametri variabili în timp;
  - C-în cazul sistemelor de control adaptiv conform schemei bloc din fig. 5.11

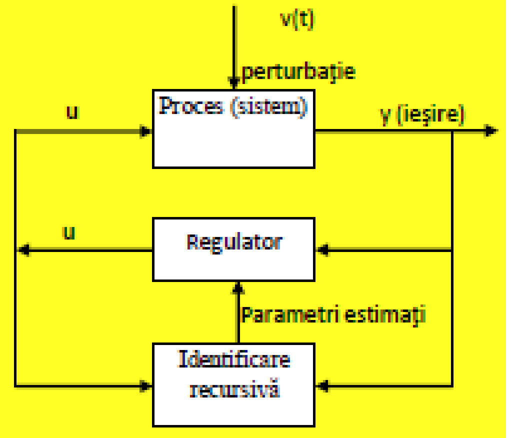


Fig. 5.11

# Metode recursive de identificare parametrică a sistemelor

- Procedurile **on-line** se impun în următoarele situaţii:
  - A-când se doreşte continuarea experimentului de identificare până la obţinerea unei precizii aprioric impuse;
  - B-când sistemul are parametri variabili în timp (***Parametrii procesului fiind variabili în timp, procesul nu poate fi reprezentat printr-un model valabil pentru orice moment de timp***);
  - C-în cazul sistemelor de control adaptiv conform schemei bloc din fig. 5.11 (***Se utilizează un model variabil în timp pentru a determina parametrii unui regulator, de asemenea, variabili în timp. În acest fel regulatorul va fi dependent de comportarea previzibilă a procesului***).

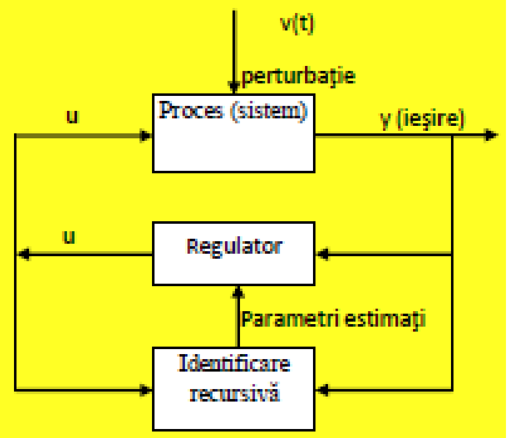


Fig. 5.11

# Metode recursive de identificare parametrică a sistemelor

- Algoritmii de estimare ***on-line***, pot fi relativ simplu obţinuţi din cei off-line printr-o organizare corespunzătoare a calculelor, astfel ca estimaţia θ(N+1) bazată pe N+1 date, să poată fi calculata recursiv utilizand numai θ(N) , masuratorile la momentul N+1 şi alte câteva măsurători anterioare.

- Estimaţia recursivă este totuşi o aproximare a estimaţiei off-line şi deci este mai puţin precisă.

# Metode recursive de identificare parametrică a sistemelor

- Metodele de identificare on-line cele mai utilizate sunt:
  1. Metoda recursivă a celor mai mici pătrate **(RCMMP)**.
  2. Metoda recursivă a celor mai mici pătrate generalizate **(RCMMPG)**
  3. Metoda recursivă a variabilelor instrumentale **(RVI)**
  4. Metoda recursivă a verosimilităţii maxime **(RVM)**

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Se consideră un sistem real monovariabil *(SISO)* descris de ecuaţiile cu diferenţe:

$$A^*(q^{-1})y(k) = B^*(q^{-1})u(k) + v(k);\ \mathrm{v(k) = e(k)}$$

$$A^*(q^{-1}) = 1 + a_1^* q^{-1} + \dots\dots\dots\dots a_{na*}^* q^{-na*}$$

$$B^*(q^{-1}) = b_1^* q^{-1} + \dots\dots\dots\dots\dots b_{nb*}^* q^{-nb*}\quad ;\ \mathrm{k} = 1, 2, \dots$$

Modelul este:

$$A(q^{-1})y_m(k) = B(q^{-1})u(k)$$

$$A(q^{-1}) = 1 + a_1 q^{-1} + \dots + a_{na} q^{-na}$$

$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2}. + \dots + b_{nb} q^{-nb} \qquad .$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Pentru model:

$$A(q^{-1})y_m(k) = B(q^{-1})u(k)$$
$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na}$$
$$B(q^{-1}) = b_1\, q^{-1} + b_2\, q^{-2}. + \ldots + b_{nb} q^{-nb} \quad .$$

Se introduc notaţiile:

$$\theta \; = [a_1 ,\ldots, a_{na} , b_1 ,\ldots, b_{nb} ]^T$$

$$\varphi(k) = [-y(k-1),\ldots\ldots,-y(k-n_a), u(k-1),\ldots, u(k-n_b)]^T$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Pentru model:

$$A(q^{-1})y_m(k) = B(q^{-1})u(k)$$
$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na}$$
$$B(q^{-1}) = b_1 q^{-1} + b_2 q^{-2}. + \ldots + b_{nb} q^{-nb} \quad .$$

Se ajunge la forma:

$$y_m(k) = \varphi^T(k)\theta$$

## Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Pentru o functie criteriu:

$$V(\theta)=\frac{1}{N}\sum_{k=1}^{N}A^2(q^{-1})\left(y(k)-y_m(k,\theta)\right)^2=\frac{1}{N}\sum_{k=1}^{N}A^2(q^{-1})e_m^2(k,\theta)$$

$$V(\theta)=\frac{1}{N}\sum_{k=1}^{N}(A(q^{-1})y(k)-B(q^{-1})u(k))^2=$$

$$=\frac{1}{N}\sum_{k=1}^{N}(y(k)-\varphi^T(k)\theta)^2$$

se determină un estimator al celor mai mici pătrate pentru N măsurători.

$$\hat{\theta}(N)=\left(\sum_{k=1}^{N}\varphi(k)\varphi^T(k)\right)^{-1}\left(\sum_{k=1}^{N}\varphi(k)y(k)\right)$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Pentru o functie criteriu generala:

$$V(\theta) = \frac{1}{N}\sum_{k=1}^{N}\lambda^{N-k}(A(q^{-1})y(k) - B(q^{-1})u(k))^2 =$$
$$= \frac{1}{N}\sum_{k=1}^{N}\lambda^{N-k}(y(k) - \varphi^T(k)\theta)^2$$

unde *λ* este un ***coeficient de ponderare*** a erorii pătratice, pentru N măsurători, se obţine estimatorul.

$$\hat{\theta}(N) = \left(\sum_{k=1}^{N}\lambda^{N-k}\varphi(k)\varphi^T(k)\right)^{-1}\left(\sum_{k=1}^{N}\lambda^{N-k}\varphi(k)y(k)\right)$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Se introduce notatia:

$$P(N) = \left( \sum_{k=1}^{N} \lambda^{N-k} \varphi(k) \varphi^T(k) \right)^{-1}$$

Si estimatorul devine:

$$\hat{\theta}(N) = P(N) \left( \sum_{k=1}^{N} \lambda^{N-k} \varphi(k) y(k) \right)$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Forma recursiva:

$$P(N+1) = [\sum_{k=1}^{N+1} \lambda^{N+1-k} \varphi(k)\varphi^T(k)]^{-1} =$$

$$= [\lambda \sum_{k=1}^{N} \lambda^{N-k} \varphi(k)\varphi^T(k) + \varphi(N+1)\varphi^T(N+1)]^{-1} =$$

$$= [\lambda P^{-1}(N) + \varphi(N+1)\varphi^T(N+1)]^{-1}$$

Tine cont de identitatea de inversiune matriciala:

$$(A + bb^T)^{-1} = A^{-1} - A^{-1}b(1 + b^T A^{-1} b)^{-1} b^T A^{-1}$$

Utilizata cu ***$A=\lambda*P^{-1}(N)$*** si ***$b=\phi(N+1)$***.

## Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Se obtine:

$$P(N+1)=\frac{P(N)}{\lambda}-\frac{P(N)}{\lambda}\varphi(N+1)[1+\varphi^T(N+1)\frac{P(N)}{\lambda}\varphi(N+1)]^{-1}\cdot$$
$$\cdot\varphi^T(N+1)\frac{P(N)}{\lambda}=\frac{1}{\lambda}[P(N)-\frac{P(N)\varphi(N+1)\varphi^T(N+1)P(N)}{\lambda+\varphi^T(N+1)P(N)\varphi(N+1)}]$$

Si notatia:

$$K(N+1)=\frac{P(N)\varphi(N+1)}{\lambda+\varphi^T(N+1)P(N)\varphi(N+1)}$$

Rezulta

$$P(N+1)=\frac{1}{\lambda}[P(N)-K(N+1)\varphi^T(N+1)P(N)]=$$
$$=\frac{1}{\lambda}[I-K(N+1)\varphi^T(N+1)]P(N)$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Forma recursive pentru estimator:

$$\hat{\theta}(N+1) = P(N+1)\sum_{k=1}^{N+1}\lambda^{N-k+1}\varphi(k)y(k) =$$

$$= P(N+1)[\lambda\sum_{k=1}^{N}\lambda^{N-k}\varphi(k)y(k) + \varphi(N+1)y(N+1)] =$$

$$= P(N)\sum_{k=1}^{N}\lambda^{N-k}\varphi(k)y(k) - K(N+1)\varphi^T(N+1)P(N)\times$$

$$\times\sum_{k=1}^{N}\lambda^{N-k}\varphi(k)y(k) + \frac{P(N)\varphi(N+1)y(N+1)}{\lambda} -$$

$$-\frac{K(N+1)\varphi^T(N+1)P(N)\varphi(N+1)y(N+1)}{\lambda} =$$

$$= \hat{\theta}(N) - K(N+1)\varphi^T(N+1)\hat{\theta}(N) +$$

$$+\frac{1}{\lambda}[P(N)\varphi(N+1) - K(N+1)\varphi^T(N+1)P(N)\varphi(N+1)]y(N+1) =$$

$$= \hat{\theta}(N) - K(N+1)\varphi^T(N+1)\hat{\theta}(N) + \frac{\lambda K(N+1)y(N+1)}{\lambda} =$$

$$= \hat{\theta}(N) + K(N+1)[y(N+1) - \varphi^T(N+1)\hat{\theta}(N] =$$

$$= \hat{\theta}(N) + K(N+1)\varepsilon(N+1)$$

Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Unde s-a tinut seama de:

$$P(N)\varphi(N+1)-K(N+1)\varphi^T(N+1)P(N)\varphi(N+1)=\lambda K(N+1)$$

Si s-a introdus notatia:

$$\varepsilon(N+1)=y(N+1)-\varphi^T(N+1)\hat{\theta}(N)$$

Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

Implementarea relaţiilor recurente este simplă. În esenţă următoarele operaţii trebuie efectuate pentru adaptarea informaţiilor disponibile la momentul *k :*

a). Se adaptează $\varphi(k) : \varphi(k) \rightarrow \varphi(k+1)$

$$\varphi(k) = \begin{bmatrix} -y(k-1) \\ -y(k-2) \\ \vdots \\ -y(k-\hat{n}_a) \\ u(k-1) \\ u(k-2) \\ \vdots \\ u(k-\hat{n}_b) \end{bmatrix} \begin{array}{l} \leftarrow \text{y(k) - noua măsurătoare a lui y} \\ \\ \\ \leftarrow \text{se şterge} \\ \leftarrow \text{u(k) noua măsurătoare a lui u} \\ \\ \\ \leftarrow \text{se şterge} \end{array}$$

**Metoda recursivă a celor mai mici pătrate (RCMMP).**

Se calculează :

b). $x_a = P(k)\varphi(k+1)$

c). $\sigma_a = \lambda + \varphi^T(k+1)\, x_a$

d). $K(k+1) = \dfrac{x_a}{\sigma_a}$

e). $\varepsilon(k+1) = y(k+1) - \varphi^T(k+1)\hat{\theta}(k)$

f). $\hat{\theta}(k+1) = \hat{\theta}(k) + K(k+1)\varepsilon(k+1)$

g). $P(k+1) = [P(k) - K(k+1)\, x_a^T]\dfrac{1}{\lambda}$

În algoritmul recursiv RCMMP trebuie date valorile iniţiale $\hat{\theta}(0), P(0)$ şi trebuie precizate valorile lui $\lambda$. Când nu se dispune de estimaţii iniţiale ale parametrilor $\hat{\theta}$ se alege de obicei $\hat{\theta}(0) = 0$

Pentru *P(0)* se alege frecvent

$$P(0) = \alpha I$$

# Metoda recursivă a celor mai mici pătrate **(RCMMP)**.

$$P(0) = \alpha I$$

unde $\alpha$ este un scalar cu valori de ordinul sutelor sau miilor, iar *I* este matricea unitate de acelaşi ordin cu *P.*

La valori mici ale lui $\alpha$ convergenţa estimaţiilor este lentă, iar la valori mari pot apărea oscilaţii de amplitudini mari pentru componentele vectorilor $\hat{\theta}(\cdot)$

Pentru sisteme cu parametri variabili în timp $\lambda$ se alege subunitar în domeniul 0,95 - 0,995.

Pentru sisteme invariante în timp $\lambda$ se alege astfel:

$\lambda < 1$, ($\lambda$ = 0,95 - 0.995) pentru primele câteva zeci de eşantioane $k < N_1$ . $\lambda = 1$, pentru $k > N_1$.

În acest fel primele reziduuri care corespund unor estimaţii imprecise ale parametrilor sunt afectate de ponderi mici în criteriul CMMP.

## Estimarea recursivă a parametrilor modelelor discrete

Pentru estimarea recursivă a parametrilor modelelor discrete, în MATLAB, se utilizează funcţiile: ***rarx, roe,rarmax, rpem, rplr*** asociate diferitelor tipuri de modele ale sistemelor.
Astfel funcţia ***rarx*** calculează estimaţiile recursive pentru modele de tip ARX. Această funcţie se apelează în forma

$$[thm, yhat] = rarx(z, nn, adm, adg) \qquad (5.140)$$

unde: *z* este un obiect *iddata* sau o matrice de date intrare-ieşire ***z* = [*y u* ];** vectorul sau matricea (în cazul intrărilor multiple) ***nn*=[*na nb ni*]** defineşte structura modelului; ***adm*** - defineşte mecanismul de adaptare; ***adg*** – defineşte factorul de amplificare de adaptare

Algoritmii implementaţi se bazează pe unul din următoarele principii de adaptare:

1) algoritmul bazat pe factorul de uitare – măsurătorile vechi sunt deponderate exponenţial pe baza factorului de uitare *lambda*; în acest caz *adm='ff'*, *adg=lam* (*lam* = factorul de uitare);
2) algoritmul bazat pe filtrul Kalman – se consideră că parametrii adevăraţi variază cu un pas aleator cu matricea de covarianţă incrementală $R_1$ ; în acest caz *adm='kf'*, *adg*= $R_1$;
3) algoritmul bazat pe gradientul nenormalizat – pasul parametrilor este luat ca un pas de gradient de lungime *gamma*=*gam* ( *th_nou=th_vechi* + *gamma*psi*epsilon*); în acest caz *adm='ng'*, *adg=gam*; 4)
4) algoritmul bazat pe gradientul normalizat – se procedează ca în cazul 3 dar *gamma* este înlocuit cu *gamma/(psi'*psi)*;

***thm*** – conţine estimaţiile obţinute;

linia *k* a matricei *thm* conţine estimaţiile în “ordine alfabetică” ce corespund datelor din linia *k* din matricea *z* şi momentului de timp *k*;

***yhat*** – conţine valorile predictate ale ieşirii; linia *k* corespunde momentului de timp *k*.

Valorile iniţiale ale parametrilor **th0** şi a “***matricei P***” ,($P(0)=P_0$) pot fi date prin apelarea următoare

$$[thm, yhat, P] = rarx(z, nn, adm, adg, th0, P_0)$$

Valorile iniţiale şi finale ale vectorului auxiliar *phi* sunt obţinute prin apelarea

$$[thm, yhat, P, phi] = rarx(z, nn, adm, adg, th0, P_0, phi_0)$$

Cu funcţia ***roe*** se calculează estimaţiile recursive pentru modelul erorii de ieşire definit de ecuaţia cu diferenţe

$$y(k) = \frac{B(q^{-1})}{F(q^{-1})} u(k - ni) + e(k)$$

Funcţia *roe* se poate apela sub forme asemănătoare cu cele ale funcţiei *rarx*. Astfel se utilizează apelările:

$$[thm, yhat] = roe(z, nn, adm, adg);$$

$$[thm, yhat, P] = roe(z, nn, adm, adg, th0, P_0);$$

$$[thm, yhat, P, phi, psi] = roe(z, nn, adm, adg, th0, P_0, phi_0, psi_0).$$

Parametrii au aceleaşi semnificaţii ca pentru funcţia *rarx*, cu excepţiile: vectorul *nn* = [*nb nf ni*] şi un nou vector auxiliar *psi* cu valoarea iniţială $psi_0$.

Funcţia ***rarmax*** calculează recursiv estimaţiile pentru parametrii unui model de tip ARMAX. Această funcţie se poate apela sub formele

$$[thm, yhat] = rar\max(z, nn, adm, adg);$$

$$[thm, yhat, P] = rar\max(z, nn, adm, adg, th0, P_0);$$

$$[thm, yhat, P, phi, psi] = rar\max(z, nn, adm, adg, th0, P_0, phi_0, psi_0)$$

Faţă de funcţiile *rarx* şi *roe* diferă parametrul de structură *nn*: *nn* = [*na nb nc ni*].

Cu funcţia ***rpem*** sunt calculate estimaţiile recursive pentru un model general descris de ecuaţia cu diferenţe de forma

$$A(q^{-1})y(k)=\frac{B(q^{-1})}{F(q^{-1})}u(k-ni)+\frac{C(q^{-1})}{D(q^{-1})}e(k) \qquad (5.146)$$

Funcţia ***rpem*** se poate apela sub formele:

$$[thm, yhat]=rpem(z,nn,adm,adg);$$

$$[thm, yhat, P]=rpem(z,nn,adm,adg,th0,P_0); \qquad (5.147)$$

$$[thm, yhat, P, phi, psi]=rpem(z,nn,adm,adg,th0,P_0,phi_0,psi_0)$$

în acest caz *nn* = [*na nb nc nd nf ni*].

Cu funcţia ***rplr*** se calculează estimaţiile recursive pentru un model general (5.146) prin metoda regresiei pseudoliniare. Funcţia **rplr** se apelează sub formele

$$[thm, yhat]=rplr(z,nn,adm,adg); \qquad (5.148)$$

$$[thm, yhat, P]=rplr(z,nn,adm,adg,th0,P_0);$$

$$[thm, yhat, P, phi]=rplr(z,nn,adm,adg,th0,P_0,phi_0)$$

**Exemplul 5.5.** Se consideră un sistem definit de polinoamele *A,B,C*. Să se determine estimaţiile recursive ale parametrilor diferitelor modele asociate acestui sistem. Iniţial se consideră pentru acest sistem un model *idpoly*

```
>>th=idpoly(A,B,C);
Discrete-time IDPOLY model: A(q)y(t) = B(q)u(t) + C(q)e(t)
A(q) = 1 - 1.2 q^-1 + 0.6 q^-2
B(q) = q^-1 + 0.5 q^-2
C(q) = 1 - 0.8 q^-1 + 0.4 q^-2
```

Se generează o matrice *z* de date intrare-ieşire, din care primele 50 de eşantioane se reprezintă în fig. 5.12

```
>>u=sign(randn(500,1));
>>e=0.1*randn(500,1);
>>y=idsim([u e],th);
>>z=[y(1:250),u(1:250)];
>>idplot(z,1:50),pause
```

Se consideră un model de ordinul 2, *na*=*nb*=2, cu o întârziere *ni* =1. Se estimează parametrii unui model *thm1* cu funcţia *rarx*, aplicând algoritmul factorului de uitare *('ff')* cu factorul de uitare *lambda*=0.96').

>>***thm1 = rarx(z,[2 2 1],'ff',0.96);***

Se reprezintă grafic parametrii estimaţi ca funcţii de timp în fig.5.13.

>> ***plot(thm1); axis([0 50 –2 2])***

Se determină parametrii unui model de tip ARMAX, *thm3*, cu structura *nn* = [2 2 2 1], cu algoritmul de adaptare cu filtru Kalman ('kf'), cu o varianţă a parametrilor de 0.01.

>>***thm3 = rarmax(z,[2 2 2 1],'kf',0.01*eye(6));***

Parametrii estimaţi sunt reprezentaţi grafic în fig. 5.15.

>>***plot(thm3); axis([0 50 -2 2]); pause***

Se determină parametrii unui model general *thm4* cu structura *nn*=[2 2 2 2 1 1] cu funcţia *rpem,* cu algoritmul factorului de uitare *('ff')*, cu factorul *lambda*=0.96

>>thm4=rpem(z,[2 2 2 2 1 1],'ff', 0.96);

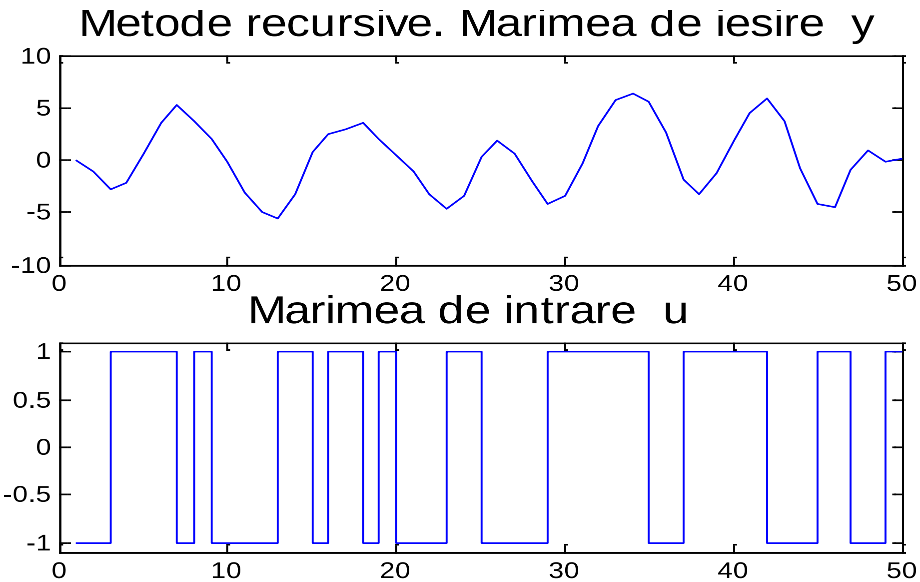



Fig. 5.12.

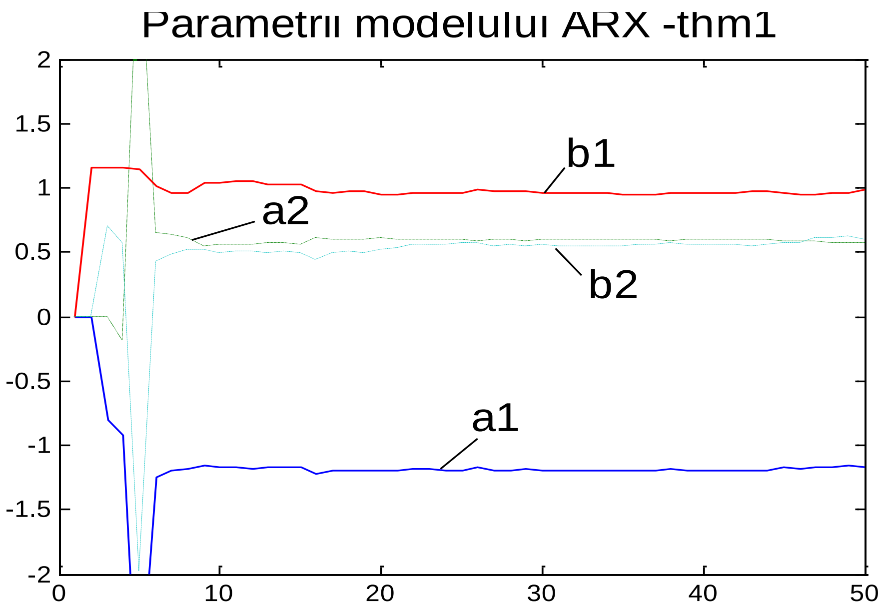



Fig. 5.13.

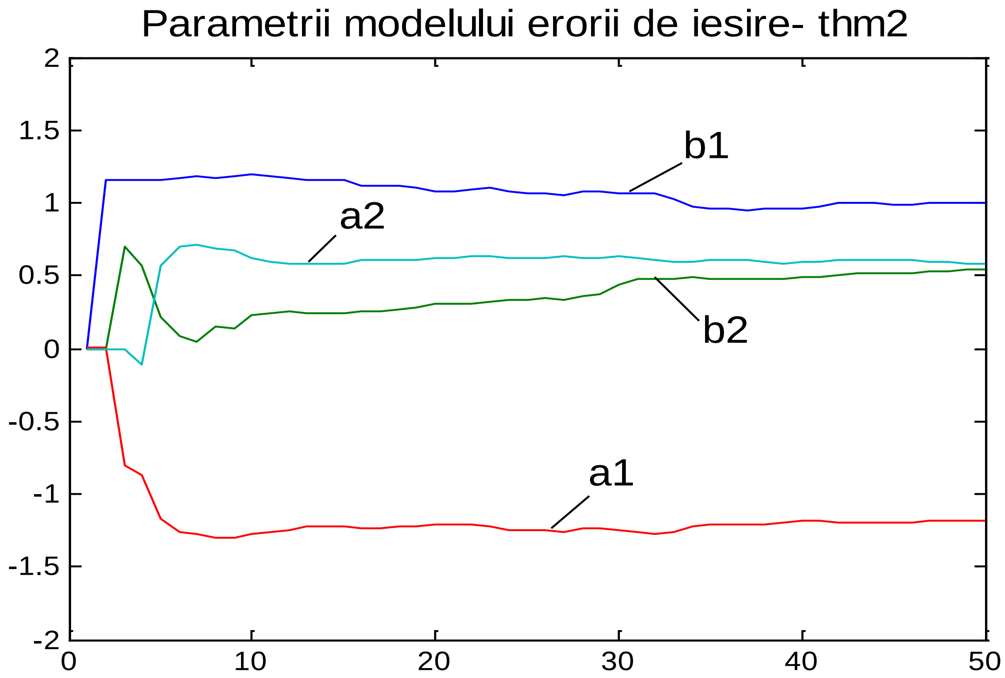



Fig. 5.14.

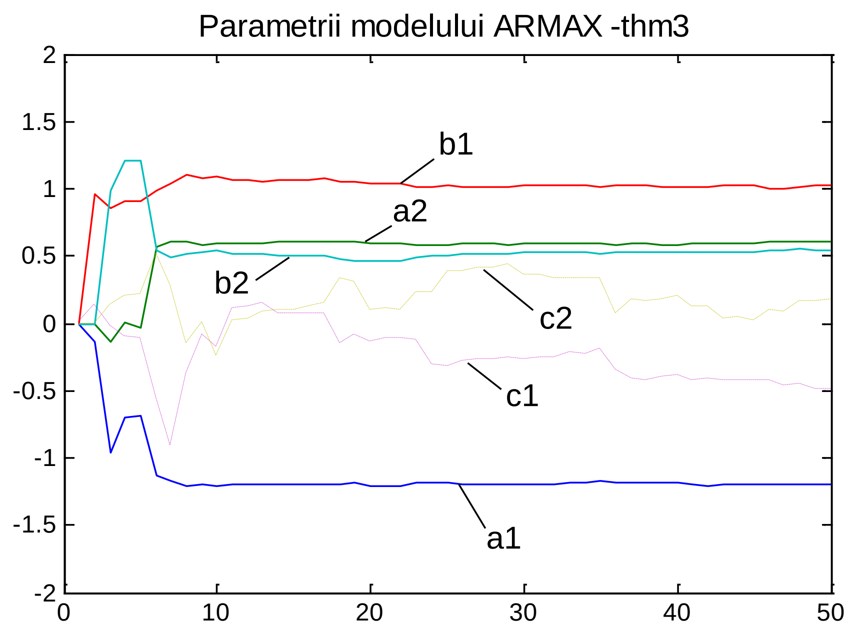



Fig. 5.15.