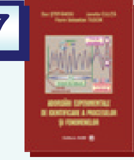
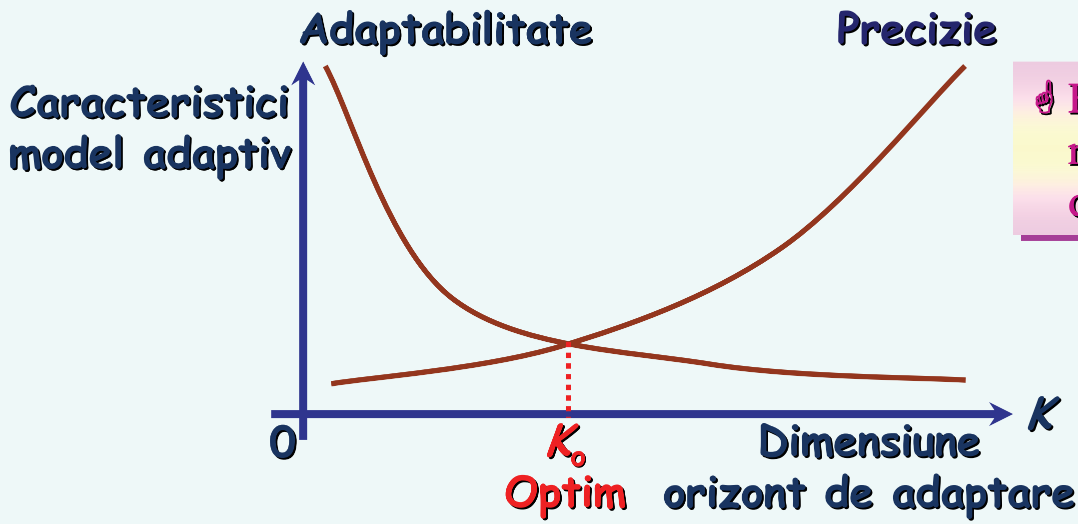


❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

6.5 p

☞ Contextul de lucru

- Majoritatea proceselor furnizoare de date sunt neliniare şi/sau posedă parametri variabili în timp.
- Identificarea proceselor cu parametri variabili în timp se realizează cu ajutorul modelelor şi metodelor adaptive (recursive).
- Prin identificare recursivă, se urmăreşte asigurarea unui compromis între două caracteristici opuse ale estimației parametrilor necunoscuți (variabili pe orizontul de măsură):



☝ Estimația vectorului parametrilor necunoscuți se reactualizează folosind datele măsurate pe orizontul de adaptare.

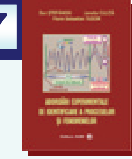
$$\hat{\theta}_{K\left\lfloor\frac{k}{K}\right\rfloor} = \hat{\theta}_{K\left\lfloor\frac{k-1}{K}\right\rfloor} + \boxed{\Delta_{K\left\lfloor\frac{k}{K}\right\rfloor}} \qquad \forall k \in \mathbb{N}^*$$

Corecție

Adaptabilitatea scade, în timp ce precizia creşte odată cu dimensiunea orizontului de adaptare.

Cu cît se achiziționează mai multe date între momentele de reactualizare, cu atît adaptarea se efectuează mai rar, modelul fiind incapabil să surprindă variațiile caracteristicilor procesului între aceste momente.

În schimb, precizia modelului creşte, deoarece parametrii săi sunt determinați cu ajutorul unui set mai bogat de date.

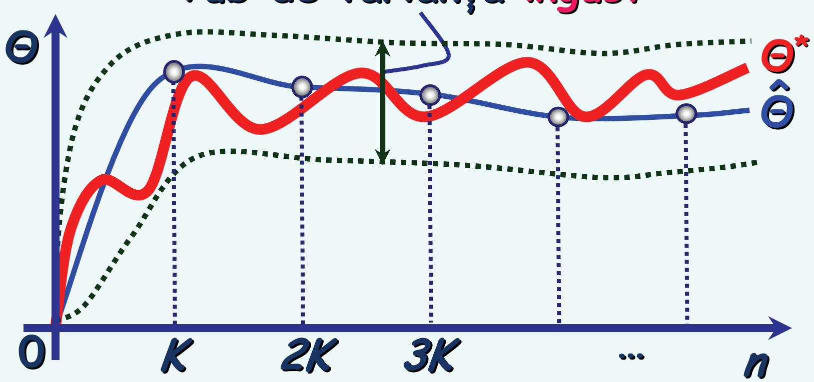


❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Contextul de lucru

☹ Asigurarea compromisului precizie-adaptabilitate este dificilă.

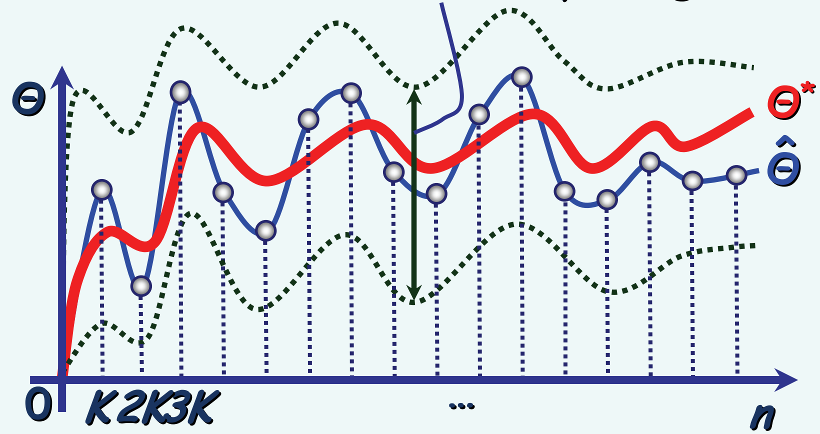
Exemplu Cazul parametrului scalar, variabil în timp.



Orizont de adaptare larg

☺ Valorile estimate ale parametrului sunt relativ apropiate de cele adevărate şi tubul de varianță este relativ îngust.

☹ Graficul parametrului estimat este neted, deci modelul sesizează mai puțin variațiile locale ale parametrului adevărat.

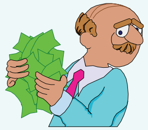


Orizont de adaptare îngust

☹ Valorile estimate ale parametrului sunt relativ depărtate de cele adevărate şi tubul de varianță este relativ larg.

☺ Graficul parametrului estimat urmăreşte variațiile locale ale parametrului adevărat, cu o anumită acuratețe.

$K = ?$



Se sacrifică precizia în favoarea adaptabilității.

$K = 1$

$\hat{\theta}_k = \hat{\theta}_{k-1} + \Delta_k \quad \forall k \in \mathbb{N}^*$

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Contextul de lucru

Algoritmul recursiv CMMP/VI de bază în IS

➢ Date de intrare:

a. ordinele modelului de identificare: na, nb, nc, nd şi nf;

b. o colecţie redusă de date intrare-ieşire măsurate (dacă este posibil): $\mathcal{D}_{N_0} = \{u[n]\}_{n\in\overline{1,N_0}} \cup \{y[n]\}_{n\in\overline{1,N_0}}$ (cu N_0 de ordinul zecilor cel mult);

c. un semnal instrumental extern: $\{f[n]\}_{n\geq 1}$ (eventual).

1. Dacă nu a fost specificat nici un semnal instrumental, vectorul variabilelor instrumentale, ζ, este identic cu vectorul regresorilor φ. Altfel, ζ este definit ca în cazul MVI, dar folosind în general semnalul instrumental extern f în locul intrării u (în particular, este posibil ca $f \equiv u$).
2. Iniţializare. Fie se setează arbitrar vectorul parametrilor $\hat{\theta}_0$ şi matricea $\mathbf{P}_0 = \alpha \mathbf{I}_{n\theta}$ (cu $\alpha \in \mathbb{R}_+^*$) (în cazul în care nu se dispune de setul de date redus $\mathcal{D}_{N_0}$), fie se estimează valoarea iniţială a parametrilor ($\hat{\theta}_0$) folosind o metodă off-line adecvată modelului particular utilizat (din clasa MCMMP-MVI) şi se egalează matricea $\mathbf{P}_0$ cu inversa matricii de covarianţă $\mathbf{R}_0$ folosită în calculul lui $\hat{\theta}_0$ (în cazul în care setul de date redus $\mathcal{D}_{N_0}$ este disponibil).
3. Pentru $k \geq 1$:

   3.1. Se evaluează eroarea de predicţie curentă: $\varepsilon[k] = y[k] - \varphi^T[k]\hat{\theta}_{k-1}$.

   3.2. Se evaluează vectorul auxiliar: $\xi_k = \mathbf{P}_{k-1}\zeta[k]$.

   3.3. Se evaluează cîştigul de senzitivitate: $\gamma_k = \dfrac{\xi_k}{1+\varphi^T[k]\xi_k}$.

   3.4. Se reactualizează inversa matricii $\mathbf{R}_k$, adică: $\mathbf{P}_k = \mathbf{P}_{k-1} - \gamma_k\varphi^T[k]\mathbf{P}_{k-1}$ (cu evitarea inversării explicite a matricilor).

   3.5. Se reactualizează vectorul parametrilor: $\hat{\theta}_k = \hat{\theta}_{k-1} + \gamma_k\varepsilon[k]$.

➢ Date de ieşire: parametrii $\hat{\theta}_k$ ai modelului de identificare la fiecare pas de reactualizare $k \geq 0$.

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Contextul de lucru

Algoritmul recursiv CMMP/VI de tip gradient

➢ *Date de intrare*:

a. ordinele modelului de identificare: na , nb , nc , nd şi nf ;

b. cîştigul de gradient: $\gamma \in \mathbb{R}_{+}^{*}$;

c. o colecţie redusă de date intrare-ieşire măsurate (dacă este posibil): $\mathcal{D}_{N_0} = \{u[n]\}_{n\in\overline{1,N_0}} \cup \{y[n]\}_{n\in\overline{1,N_0}}$ (cu N_0 de ordinul zecilor, cel mult);

d. un semnal instrumental extern: $\{f[n]\}_{n\geq 1}$ (eventual).

1. Dacă nu a fost specificat nici un semnal instrumental, vectorul variabilelor instrumentale, ζ , este identic cu vectorul regresorilor φ . Altfel, ζ este definit ca în cazul MVI, dar folosind în general semnalul instrumental extern f în locul intrării u (în particular, este posibil ca $f \equiv u$).

2. Iniţializare. Fie se setează arbitrar vectorul parametrilor $\hat{\theta}_0$ (în cazul în care nu se dispune de setul de date redus $\mathcal{D}_{N_0}$), fie se estimează valoarea iniţială a parametrilor ($\hat{\theta}_0$) folosind o metodă off-line adecvată modelului particular utilizat (din clasa MCMMP-MVI) (în cazul în care setul de date redus $\mathcal{D}_{N_0}$ este disponibil). Se setează $\mathbf{P}_0 = \gamma \mathbf{I}_{n\theta}$.

3. Pentru $k \geq 1$:

   3.1. Se evaluează eroarea de predicţie: $\varepsilon[k] = y[k] - \varphi^T[k]\hat{\theta}_{k-1}$.

   3.2. Se reactualizează matricea $\mathbf{P}_k$ astfel: $\mathbf{P}_k = \mathbf{P}_0$ (gradient ne-normalizat) sau $\mathbf{P}_k = \mathbf{P}_0 / \|\varphi[k]\|^2$ (gradient normalizat).

   3.3. Se reactualizează vectorul parametrilor: $\hat{\theta}_k = \hat{\theta}_{k-1} + \mathbf{P}_k\varphi[k]\varepsilon[k]$.

➢ *Date de ieşire*: parametrii $\hat{\theta}_k$ ai modelului de identificare la fiecare pas de reactualizare $k \geq 0$.

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Contextul de lucru

Algoritmul recursiv CMMP/VI cu filtrare de tip Kalman-Bucy

➢ *Date de intrare:*

a. ordinele modelului de identificare: na, nb, nc, nd şi nf;

b. matricea de răspîndire: $\mathbf{R}_v > 0$;

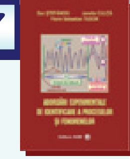
c. dispersia estimată a zgomotului alb de proces: λ^2;

d. o colecţie redusă de date intrare-ieşire măsurate (dacă este posibil): $\mathcal{D}_{N_0} = \{u[n]\}_{n\in\overline{1,N_0}} \cup \{y[n]\}_{n\in\overline{1,N_0}}$ (cu N_0 de ordinul zecilor, cel mult);

e. un semnal instrumental extern: $\{f[n]\}_{n\geq 1}$ (eventual).

1. Dacă nu a fost specificat nici un semnal instrumental, vectorul variabilelor instrumentale, ζ, este identic cu vectorul regresorilor φ. Altfel, ζ este definit ca în cazul MVI, dar folosind în general semnalul instrumental extern f în locul intrării u (în particular, este posibil ca $f \equiv u$).
2. Iniţializare. Fie se setează arbitrar vectorul parametrilor $\hat{\theta}_0$ şi matricea $\mathbf{P}_0 = \alpha \mathbf{I}_{n\theta}$ (cu $\alpha \in \mathbb{R}_+^*$) (în cazul în care nu se dispune de setul de date redus $\mathcal{D}_{N_0}$), fie se estimează valoarea iniţială a parametrilor ($\hat{\theta}_0$) folosind o metodă off-line adecvată modelului particular utilizat (din clasa MCMMP-MVI) şi se egalează matricea $\mathbf{P}_0$ cu inversa matricii de covarianţă $\mathbf{R}_0$ folosită în calculul lui $\hat{\theta}_0$ (în cazul în care setul de date redus $\mathcal{D}_{N_0}$ este disponibil).
3. Pentru $k \geq 1$:
   - 3.1. Se evaluează eroarea de predicţie: $\varepsilon[k] = y[k] - \varphi^T[k]\hat{\theta}_{k-1}$.
   - 3.2. Se evaluează vectorul auxiliar: $\xi_k = \mathbf{P}_{k-1}\zeta[k]$.
   - 3.3. Se evaluează cîştigul de senzitivitate: $\gamma_k = \dfrac{\xi_k}{\lambda^2 + \varphi^T[k]\xi_k}$.
   - 3.4. Se reactualizează matricea $\mathbf{P}_k$, adică: $\mathbf{P}_k = \mathbf{P}_{k-1} + \mathbf{R}_v - \gamma_k \varphi^T[k]\mathbf{P}_{k-1}$ (cu evitarea inversării explicite a matricilor).
   - 3.5. Se reactualizează vectorul parametrilor: $\hat{\theta}_k = \hat{\theta}_{k-1} + \gamma_k \varepsilon[k]$.

➢ *Date de ieşire:* parametrii $\hat{\theta}_k$ ai modelului de identificare la fiecare pas de reactualizare $k \geq 0$.



❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Contextul de lucru

Procesele generatoare de date

ARMAX[1,1,1] $\left(1+a_1[n]q^{-1}\right)y[n]=b_1[n]q^{-1}u[n]+\left(1+c_1[n]q^{-1}\right)e[n]$ $\forall n\in\mathbb{N}^*$

Parametri constanţi

$a_1[n]=a_{10}=-0.7$

$b_1[n]=b_{10}=0.6$

$c_1[n]=c_{10}=-0.9$

$\forall n\in\mathbb{N}^*$

Parametri variabili

$a_1[n]\overset{def}{=}a_{10}\cos\left(\frac{10\pi n}{N}\right)$

$b_1[n]\overset{def}{=}b_{10}\,\mathrm{sgn}\left[\cos\left(\frac{4\pi n}{N}\right)\right]$

$c_1[n]\overset{def}{=}c_{10}\,\mathrm{Sc}\left(\frac{18\pi n}{N}\right)$

$\forall n\in\mathbb{N}^*$

e ➔ SPAB Gaussian sau bipolar de medie nulă şi dispersie unitară

☝ Pe un orizont mare de timp, modelul ARMAX tinde să devină un model ARX.

Date generate $\mathcal{D}=\{u[n]\}_{n=\overline{1,N}}\cup\{y[n]\}_{n=\overline{1,N}}$

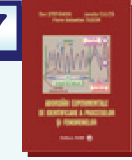
ales liber de către utilizator ➔ $N\geq 200$

Indici structurali sunt cunoscuţi $na=nb=nc=1$

Test de stop ➔ Epuizarea datelor de pe orizontul de măsură.

Obiectiv

- Compararea performanţelor metodelor recursive de identificare fără fereastră, în cazul modelelor din clasa ARMAX.



❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Contextul de lucru

Rutine preliminare [●○○○○]

```
[theta,ypred] = rnume(D,si,ma,pa) ;
```

Rutine de bibliotecă MATLAB-IS

rarmax	rarx	rbj	rpem	rplr	roe
ARMAX	ARX	BJ	RSISO	ARMAX	OE
MMEP-R	MCMMP-R	MMEP-R	MMEP-R	MRPL-R	MMEP-R

Metoda de Regresie Pseudo-Liniară

MMEP în care s-a înlocuit MGN cu o metodă de optimizare mai precisă: Metoda Newton-Raphson (MNR)

☝ Argumentele de intrare al acestor funcţii pot fi completate cu variabile care să indice explicit o anumită iniţializare a procesului recursiv.

```
[theta,ypred,P,phi] = rnume(D,si,ma,pa,theta0,P0,phi0) ;
```

D este obiectul de tip **IDDATA** corespunzător datelor generate (intrarea se regăseşte în **D.u**, iar ieşirea în **D.y**);

si este vectorul indicilor structurali şi al întîrzierii modelului (ca în cazul rutinei **pem**):

si = [na nb nc nd nf nk];

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Contextul de lucru

Rutine preliminare [●●○○○]

```
[theta,ypred,P,phi] = rnume(D,si,ma,pa,theta0,P0,phi0) ;
```

`ma` este un argument care indică metoda de adaptare a algoritmului off-line la algoritmul on-line (şir de 2 caractere):

- **`'ff'`** se va opera cu criteriu pătratic afectat de fereastră exponenţială (i.e. cu factor de uitare; `'ff'` = *forgetting factor*);
- **`'ug'`** se va utiliza o metodă de gradient (Newton-Raphson) ne-normalizată (`'ug'` = *unnormalized gradient*);
- **`'ng'`** se va utiliza o metodă de gradient (Newton-Raphson) normalizată (`'ng'` = *normalized gradient*);
- **`'kf'`** se va utiliza reprezentarea pe stare şi filtrarea Kalman (aici: `'kf'` = *Kalman filtering*);

`pa` este un parametru de adaptare corespunzător metodei de adaptare a algoritmului off-line la algoritmul on-line, adică argumentului `ma` (scalar sau matrice):

- λ factorul de uitare (scalar) pentru fereastra exponenţială (cînd argumentul `ma` este setat cu `'ff'`);
- γ cîştigul dorit (scalar) în cazul utilizării algoritmilor de gradient (cînd argumentul `ma` este setat cu `'ug'` sau `'ng'`);
- R_v matricea de răspîndire în cazul utilizării algoritmului bazat pe filtrare Kalman (cînd argumentul `ma` este setat cu `'kf'`); în acest caz, parametrul λ^2 (dispersia zgomotului alb) este considerat implicit egal cu 1; dacă, în realitate, λ^2 nu este unitară, se poate demonstra că estimaţia parametrilor nu este afectată dacă se scalează matricile R_v şi P_0 cu valoarea estimată a sa (urmînd să se lucreze tot cu $\lambda^2 = 1$, ca în cazul implicit).

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Contextul de lucru

Rutine preliminare [●●●○○]

```
[theta,ypred,P,phi] = rnume(D,si,ma,pa,theta0,P0,phi0) ;
```

theta este matricea parametrilor estimați variabili în timp; fiecare linie a matricii memorează valoarea parametrilor la un anumit moment de timp; numărul de linii este egal cu lungimea orizontului de măsură (adică a vectorilor **D.u** şi **D.y**); pe fiecare linie, parametrii sunt precizați în ordinea alfabetică a numelor polinoamelor pe care le reprezintă (A, B, C, D, F);

ypred este vectorul ieşirii predictate a procesului la fiecare moment de timp, folosind modelul matematic reactualizat; lungimea sa este egală cu lungimea orizontului de măsură.

theta0 este vectorul inițial al parametrilor;

P0 este matricea inițială $\mathbf{P}_0$;

phi0 este vectorul inițial al regresorilor $\varphi[0]$;

P este matricea finală $\mathbf{P}_N$;

phi este vectorul final al regresorilor $\varphi[N]$.

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Contextul de lucru

☝ Inexistentă în biblioteca MATLAB-IS. Disponibilă pe site.

Rutine preliminare [●●●●○]

```
[theta,ypred,P,phi,z] = riv(D,si,f,lambda,theta0,P0,phi0,z0) ;
```

MVI-R

`f`	este semnalul instrumental (implicit: `f=D.u`);
`lambda`	este factorul de uitare ($\lambda \in (0,1]$) (implicit: `lambda=1`);
`z0`	este vectorul inițial al instrumentelor $\zeta[0]$;
`z`	este vectorul final al instrumentelor $\zeta[N]$.

Restul argumentelor funcției au fost explicitate mai înainte, iar indicii structurali **si** conțin numai ordinele modelului ARX.

Cu toate acestea, algoritmul este implementat numai în varianta cu fereastră exponențială (de aceea argumentul **ma** lipsește).

```
[D,V,P] = gdata_vp(cv,N,sigma,lambda,bin) ;
```

(generează date)

`cv`	este un comutator care arată tipul de proces: cu parametri constanți (`cv=0`) sau cu parametri variabili (`cv~=0`); (implicit: `cv=0`);
`N`	este orizontul de măsură (implicit: `N=250`);
`sigma`	este deviația standard a intrării SPA (implicit: `sigma=1`);

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Contextul de lucru

Rutine preliminare [●●●●●]

`[D,V,P] = gdata_vp(cv,N,sigma,lambda,bin) ;` (generează date)

`lambda` este deviația standard a zgomotului alb Gaussian (implicit: `lambda=1`);

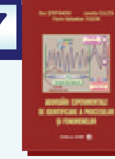
`bin` este un parametru care arată tipul de intrări dorit: `bin=0` (intrare SPAB Gaussiană); `bin~=0` (implicit, intrare SPAB Gaussiană bipolară);

`D` este obiectul de tip `IDDATA` corespunzător datelor generate (intrarea se regăseşte în `D.u`, iar ieşirea în `D.y`);

`V` este obiectul de tip `IDDATA` corespunzător zgomotelor generate (zgomotul alb se regăseşte în `V.u`, iar zgomotul colorat (adică MA-filtrat) în `V.y`);

`P` este obiectul de tip `IDMODEL` corespunzător modelului de proces furnizor de date; în cazul parametrilor constanți: `P.a=[1 a0]`, `P.b=[0 b0]`, `P.c=[1 c0]`; în cazul parametrilor variabili: `P.a=[1 a]`, `P.b=[0 b]`, `P.c=[1 c]` (unde `a`, `b` şi `c` sunt vectorii de variație).

☝ Se vor genera 2 seturi de date: unul provenit de la procesul cu parametri constanți şi altul – de la procesul cu parametri variabili.



❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Problema 7.1 (Identificare recursivă comparativă cu algoritmul de bază)

Mini-simulatorul **ISLAB_7A** efectuează o comparație între cele 4 metode de identificare recursive menționate, adică: MCMMP-R, MVI-R, MMEP-R şi MRPL-R, folosind setul de date $\mathcal{D}_c$ (generate de procesul cu parametri constanți). Primele două metode operează cu modelul ARX[1,1] (extras din modelul ARMAX prin anularea componentei MA), în timp ce ultimele două – cu modelul ARMAX[1,1,1]. Pentru aprecierea performanțelor lor, sunt afişate 4 ferestre grafice care includ variațiile parametrilor reali (aici constanți) suprapuse peste variațiile parametrilor estimați şi variația ieşirii simulate suprapuse peste ieşirea reală (măsurată) a procesului.

a. Să se comenteze rezultatele obținute cu ajutorul mini-simulatorului **ISLAB_7A**. Care ar fi explicațiile performanțelor mai slabe ale MCMMP-R în estimarea parametrului părții AR?

0.5p

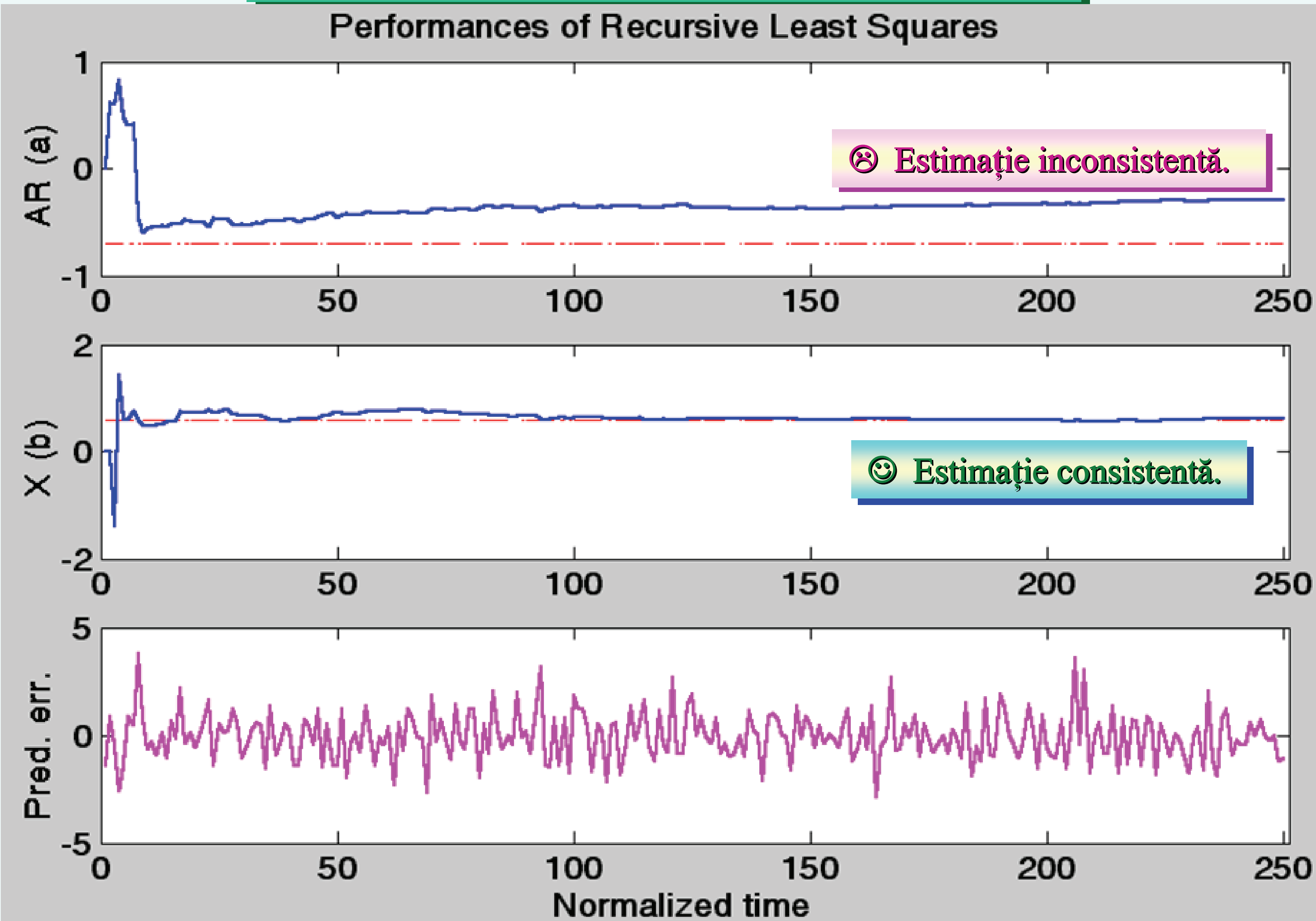
b. Să se proiecteze mini-simulatorul **ISLAB_7B**, similar ca structură cu **ISLAB_7A**, dar care operează cu datele $\mathcal{D}_v$ (generate de procesul cu parametri variabili). Comentați rezultatele obținute.

0.5p

Program existent | **ISLAB_7A** | Program ce trebuie proiectat | **ISLAB_7B** | Comentarii | 0.5p

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

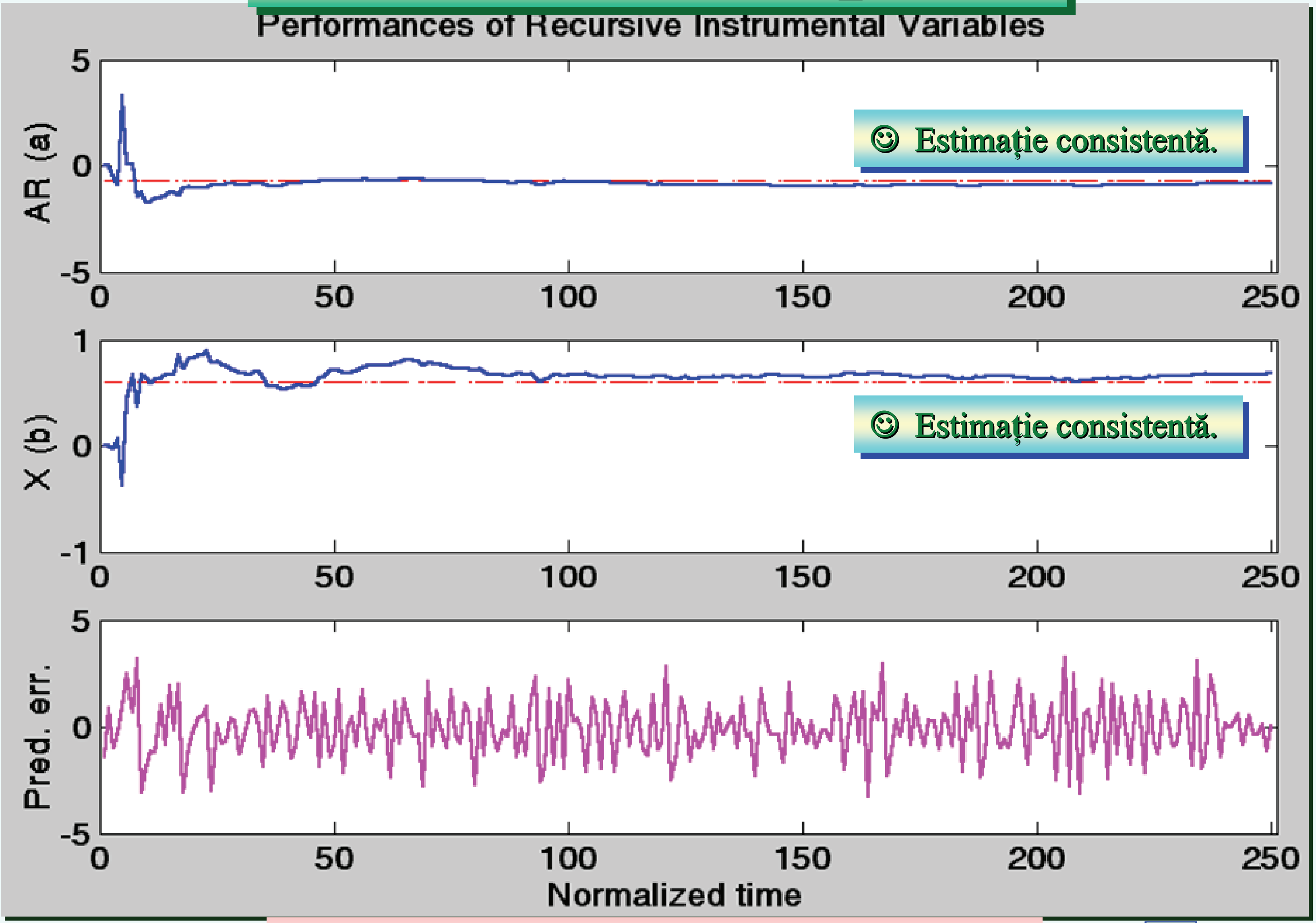
Ce afişează mini-simulatorul **ISLAB_7A** [●○○○]



Performanţele MCMMP-R în cazul modelului ARX

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

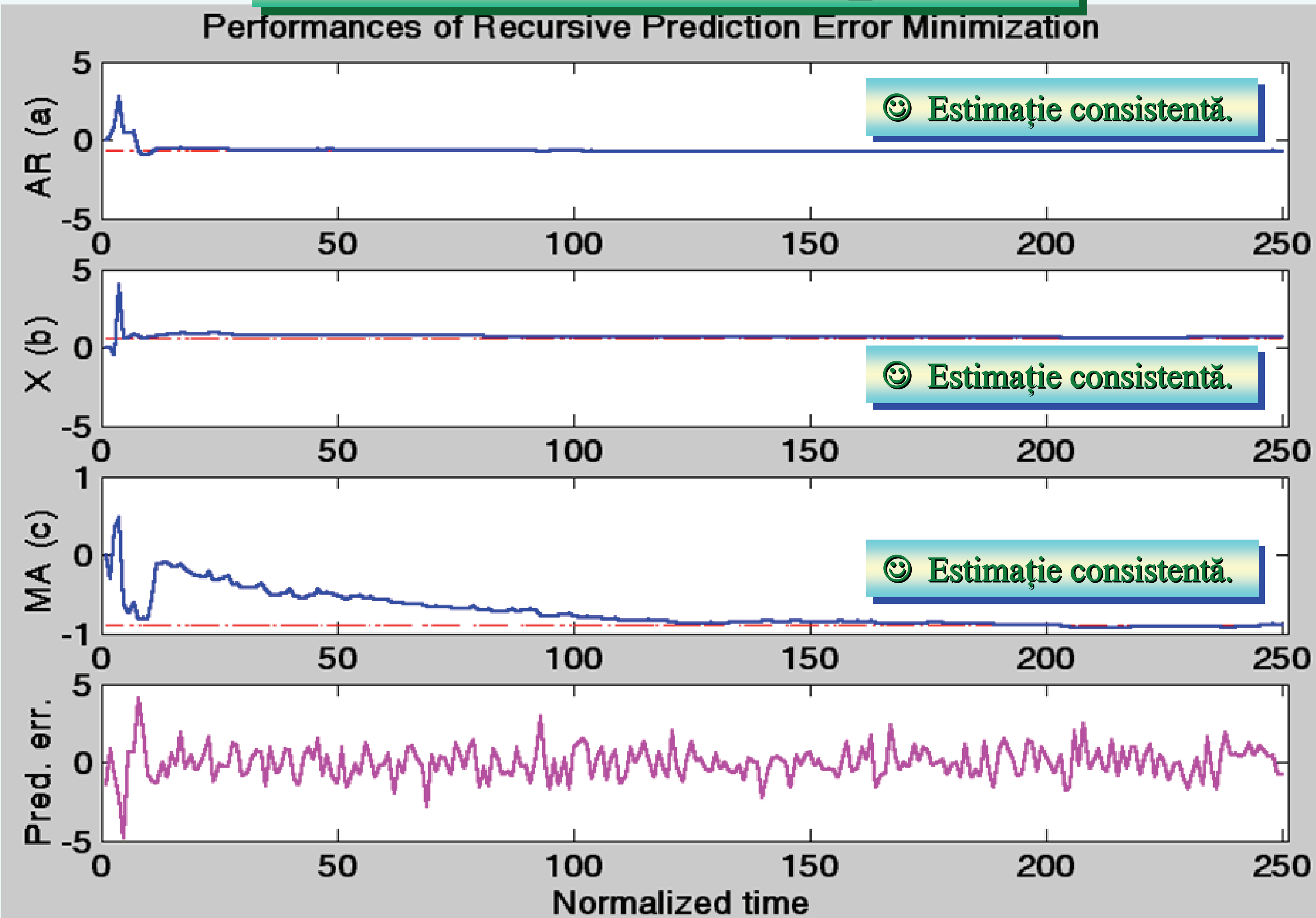
Ce afişează mini-simulatorul **ISLAB_7A** [●●○○]



Performanţele MVI-R în cazul modelului ARX

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

Ce afişează mini-simulatorul **ISLAB_7A** [●●●○]

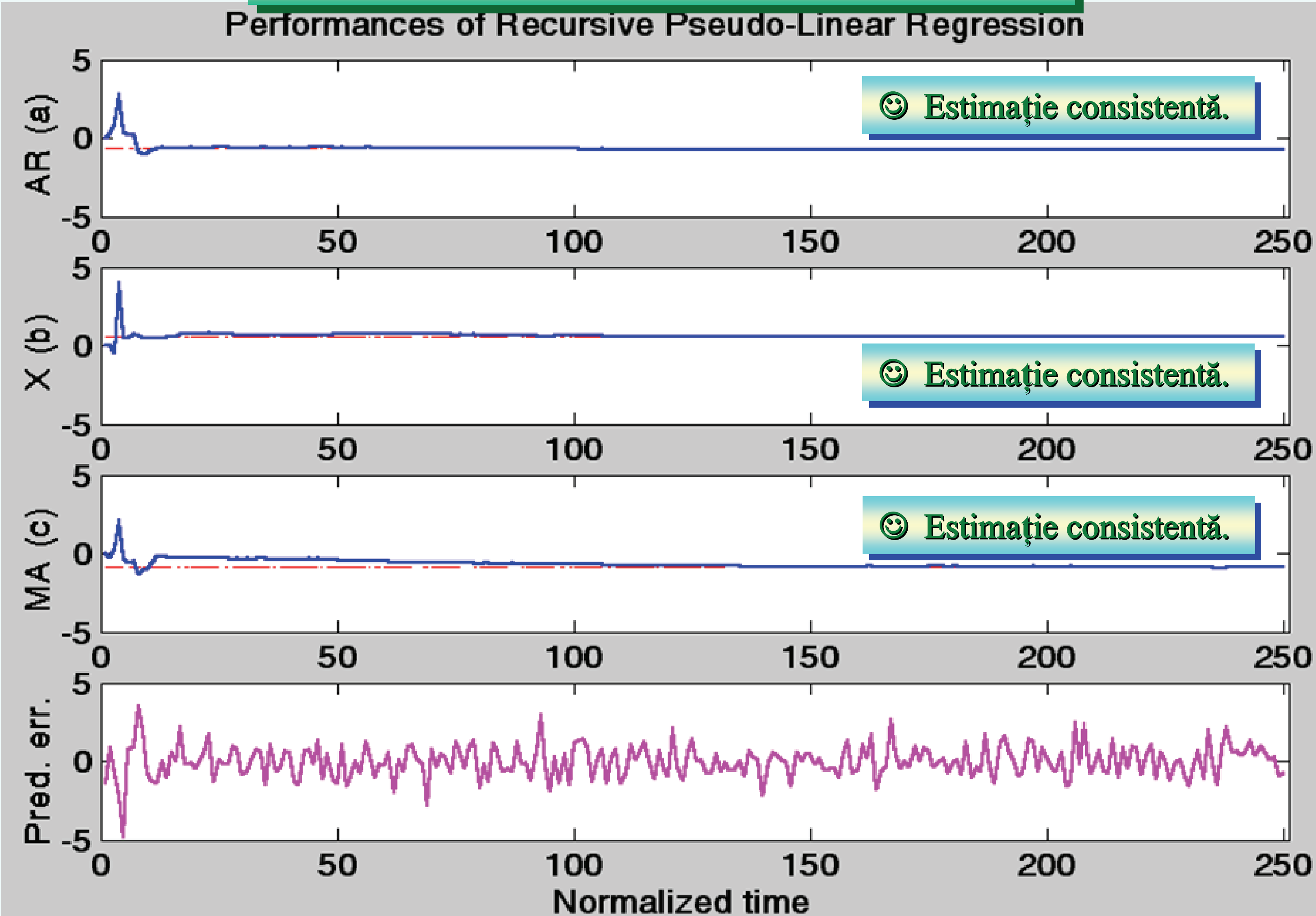


☝ Eficienţă superioară.

Performanţele MMEP-R în cazul modelului ARMAX

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

Ce afişează mini-simulatorul **ISLAB_7A** [●●●●]



☝ Eficienţă maximă, dar şi complexitate ridicată.

Performanţele MPRL-R în cazul modelului ARMAX

❻ Algoritmi rapizi CMMP-R şi VI-R fără fereastră

☞ Probleme de simulare

Problema 7.2 (Identificare recursivă cu diferite inițializări)

Să se proiecteze mini-simulatorul **ISLAB_7C**, care să afişeze performanțele MCMMP-R şi MVI-R în cazul modelelor ARX cu parametri constanți şi variabili pentru inițializările $\mathbf{P}_0 = \alpha \mathbf{I}$, cu $\alpha \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. Descrieți influența factorului α şi efectuați o analiză comparativă.

Comentarii 0.5p

Program ce trebuie proiectat **ISLAB_7C** 0.5p

Problema 7.3 (Identificare recursivă comparativă)

Comentarii 1p

Se consideră doar procesul ARMAX cu parametri variabili, pentru valori mari ale momentelor de timp. Din acest motiv, procesul va fi aproximat cu unul de tip ARX, adică se va neglija variația coeficientului c_1. Pentru a genera datele corespunzătoare, se va utiliza tot funcția **gdata_vp**, dar cu o durată a simulării stabilită la 1000 de eşantioane. Pentru experimentul care urmează, se vor selecta însă doar ultimele 250 de date I/O.

a. Să se proiecteze rutinele **arx_nabla** şi **arx_KB**, care implementează algoritmii recursvi CMMP/VI de tip gradient, respectiv cu filtrare de tip Kalman-Bucy. Rutinele vor avea argumentele principale de intrare şi ieşire similare rutinelor **rarx**, **riv**.

b. Să se proiecteze mini-simulatorul **ISLAB_7D**, similar ca structură cu **ISLAB_7B**, dar care oferă posibilitatea utilizatorului de a selecta oricare dintre cei 6 algoritmi recursivi (cei 4 din cadrul **ISLAB_7B**, plus cei doi implementați prin rutinele de la punctul anterior). Efectuați diferite simulări cu ajutorul programului **ISLAB_7D**, care să pună în evidență deosebirile dintre cei 6 algoritmi. Comentați rezultatele obținute.

Software ce trebuie proiectat **ISLAB_7D** 0.5p **arx_nabla** 1p **arx_KB** 1.5p