

# Identificarea Sistemelor

## LABORATOR 6

Mărgăritescu Vlad - 342B3

### PROBLEMA 1 (Identificare recursiva comparativa cu algoritmul de baza)

Cu ajutorul mini-simulatorului ISLAB\_7A se efectueaza o comparatie intre cele 4 metode, folosind setul de date Dc (generate de procesul cu parametri constanti).

#### ISLAB\_7A

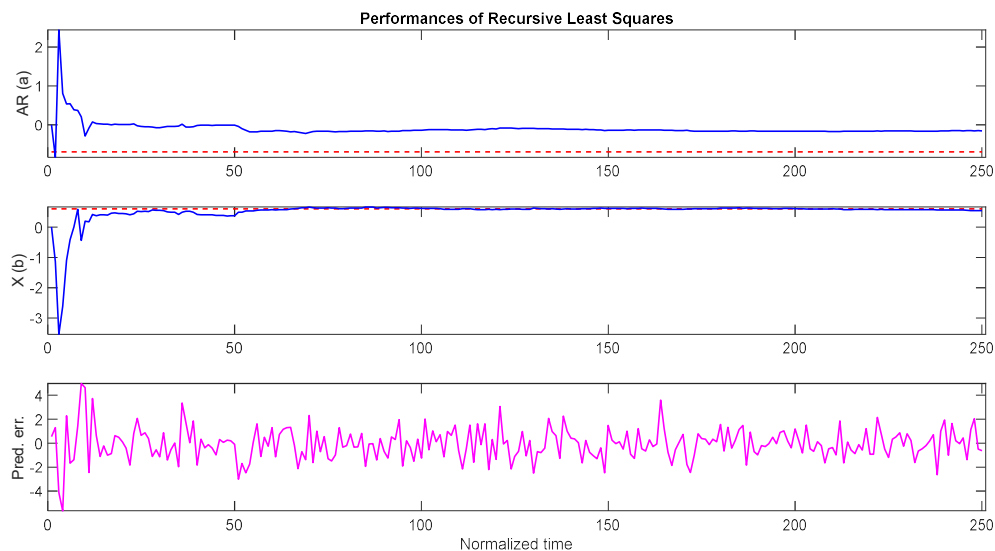


Fig. 1 – MCMMP-R

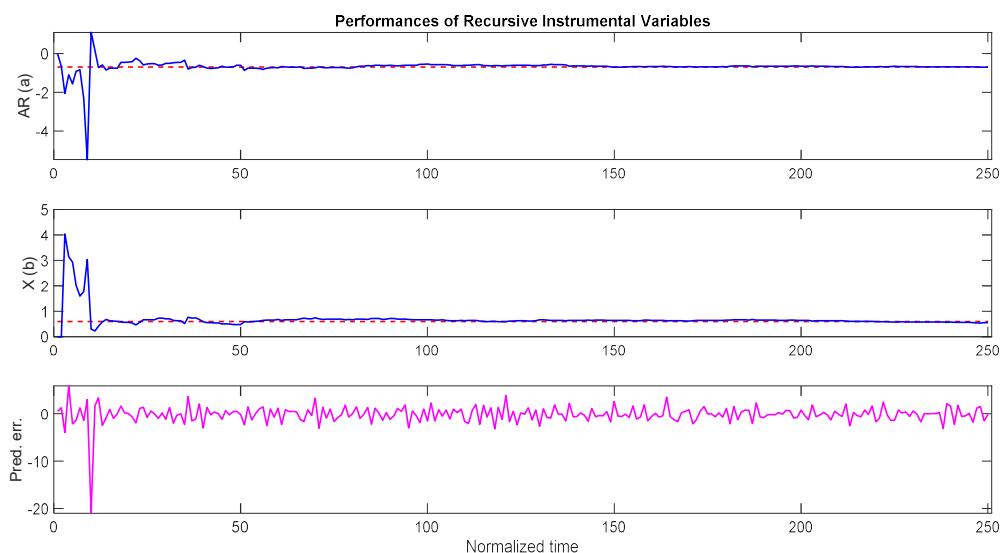


Fig. 2 – MVI-R

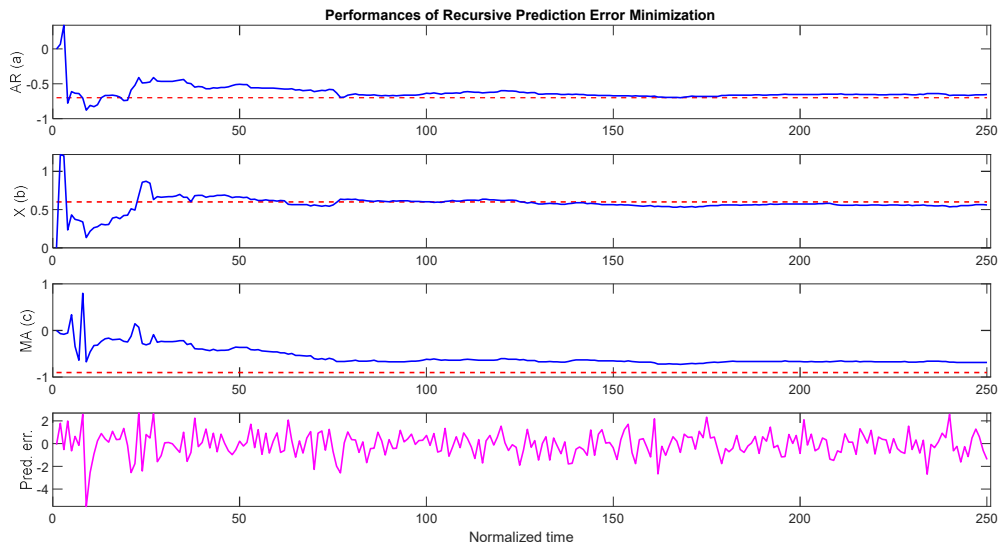


Fig. 3 – MMEP-R

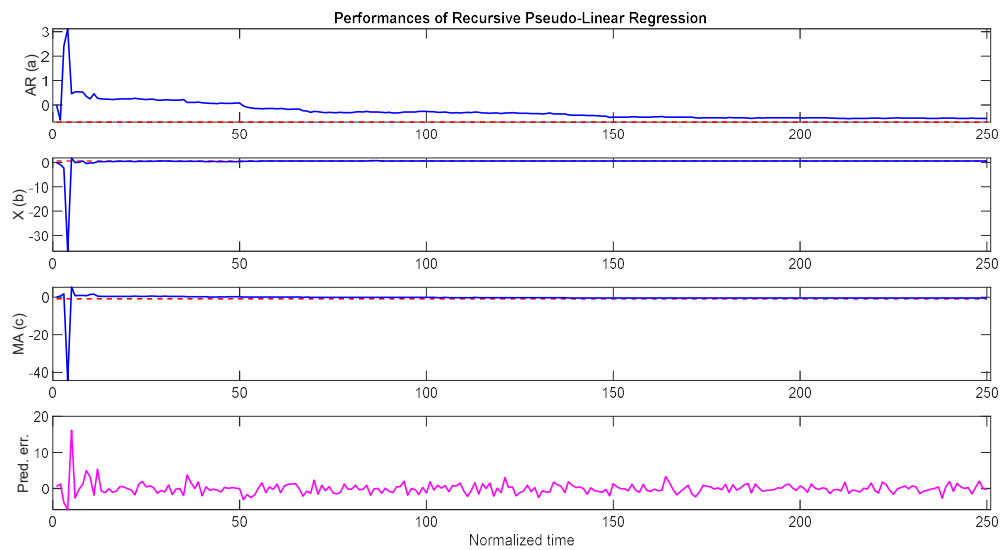


Fig. 4 – MRPL-R

### Concluzii:

a) Care ar fi explicațiile performanțelor mai slabe ale MCMMP-R în estimarea parametrului părții AR?

Soluție: Se observa cum in figura 1, componenta AR nu este identificata corect de catre metoda MCMMP-R, deoarece primele 2 metode sunt aplicate unui model ARX[1,1] extras dintr-un model ARMAX, astfel apar erori de identificare si aproximare.

Ultimele 2 opereaza cu modelul ARMAX[1,1,1] si se vede cum estimatiile sunt consistente si mai precise.

## ISLAB\_7B

La acest punct trebuie sa apelam functia `gdata_vp`, dar cu `cv!=0`, intrucat `cv` este parametrul care face schimbarea dintre parametri constanti si parametri variabili.

```
[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**);

```
% Generating the identification data (ARMAX)
```

```
% ~~~~~
```

```
[D,~,P] = gdata_vp(1,N,sigma,lambda,0) ;
```

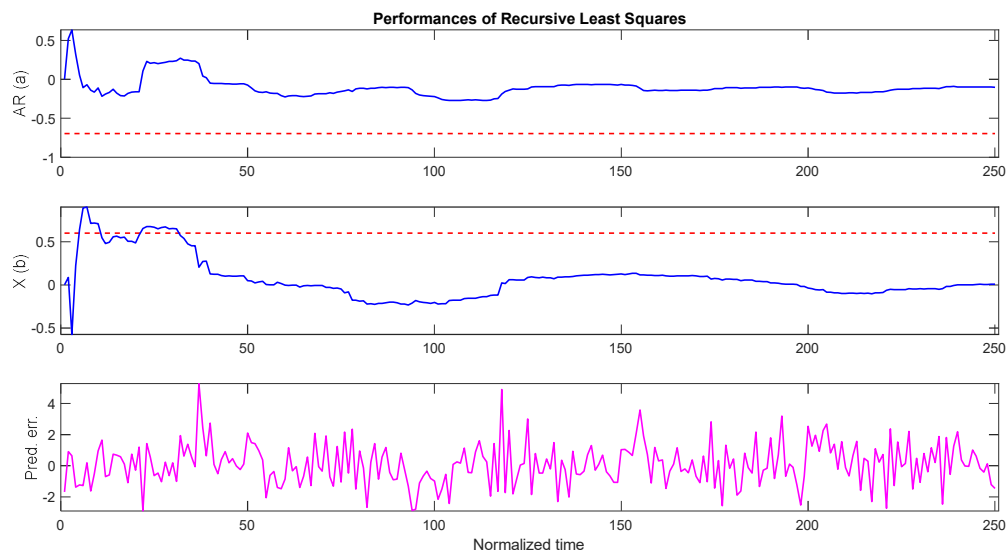


Fig. 5 – MCMMP - R – param variabili

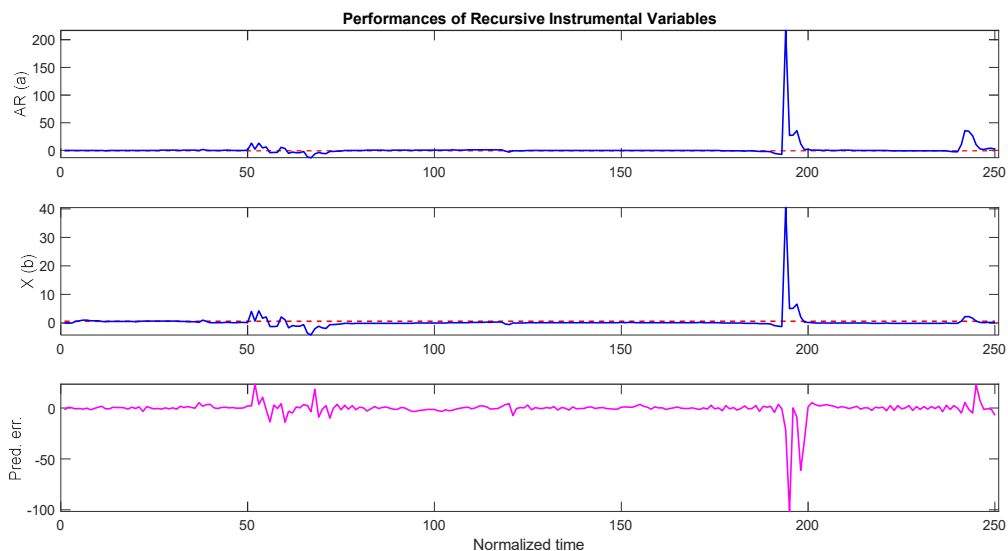


Fig. 6 – MVI - R – param variabili

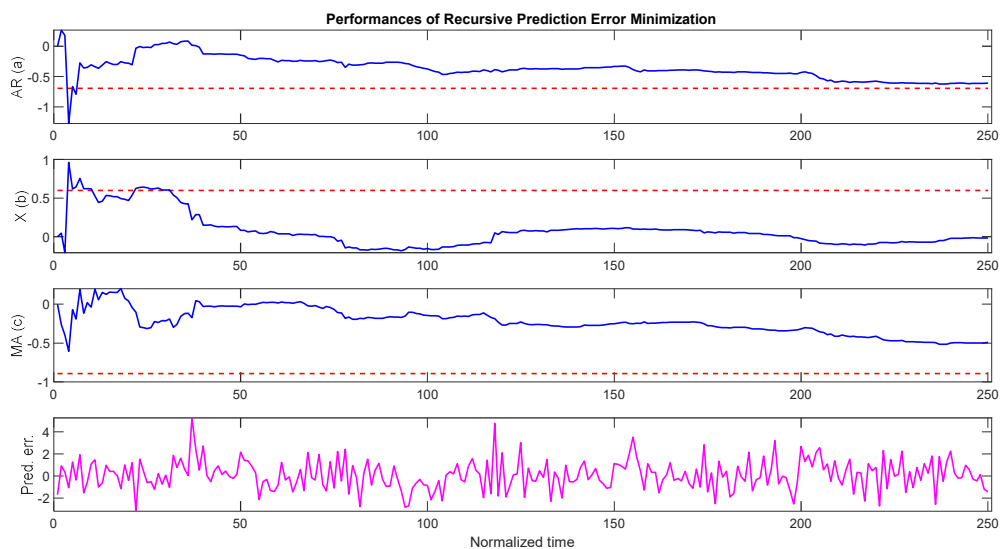


Fig. 7 – MMEP - R – param variabili

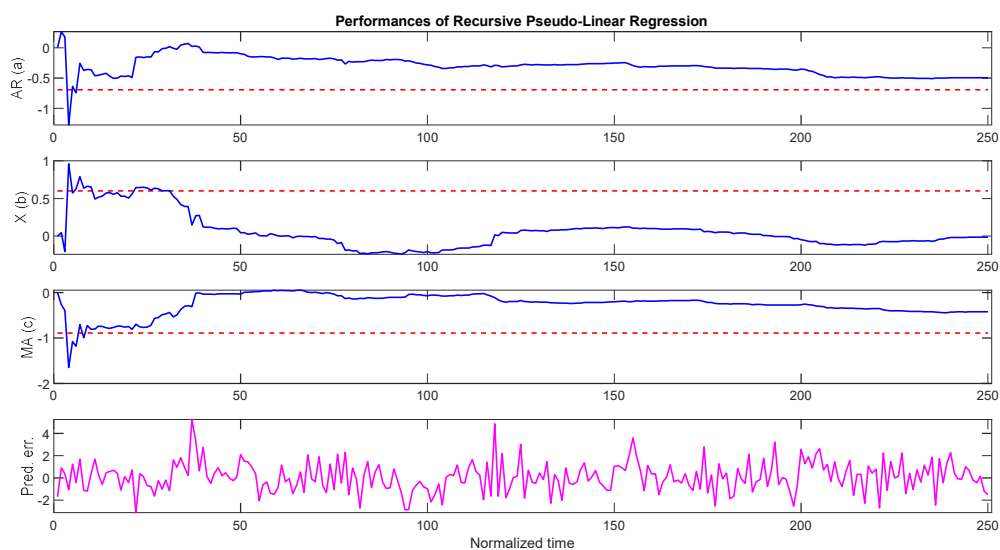


Fig. 8 – MRPL - R – param variabili

### Concluzii:

Se poate observa cum aceasta metoda ofera niste solutii foarte slabe.

Aproape toate estimatiile sunt inconsistente. Doar in cazul MVI avem estimatii consistente.

Astfel, folosind parametri variabili, duce la un rezultat chiar mai slab fata de metodele cu parametri constanti.

In concluzie, trebuie evitata aceasta abordare.

## PROBLEMA 2 (Identificare recursiva cu diferite initializari)

Se adauga in cod vectorul alpha si se modifica functiile rarx si riv pentru a include si P0.

Se combina cele 2 subrutine precedente: ISLAB\_7A si ISLAB\_7B pentru a include cazurile cu parametri constanti si parametri variabili. (parametrul cv=0/1)

Pentru fiecare alpha se afiseaza cele 4 grafice, adica un total de 28 de grafice.

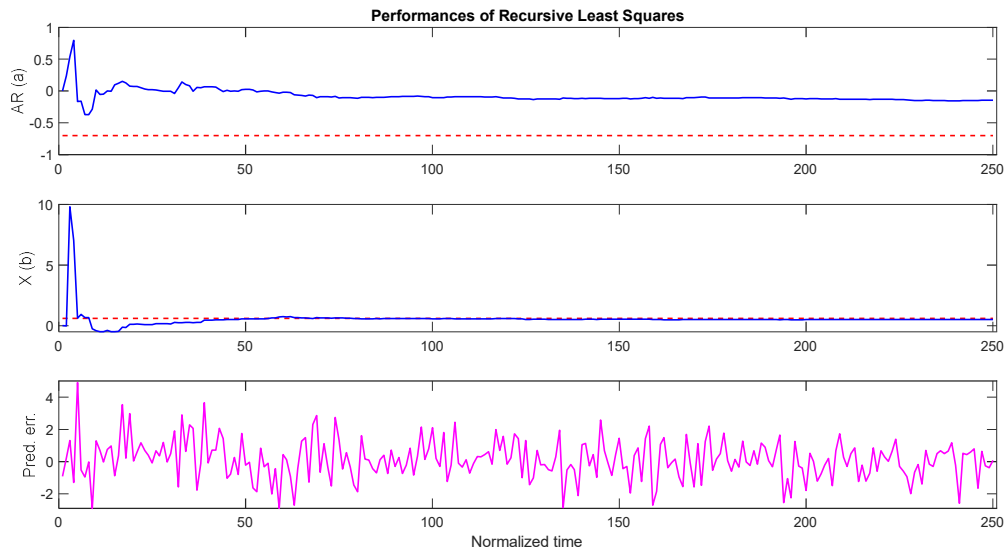


Fig. 9 – MCMMP - R – param constanti

Toate graficele arata la fel pentru orice valoare a lui alpha

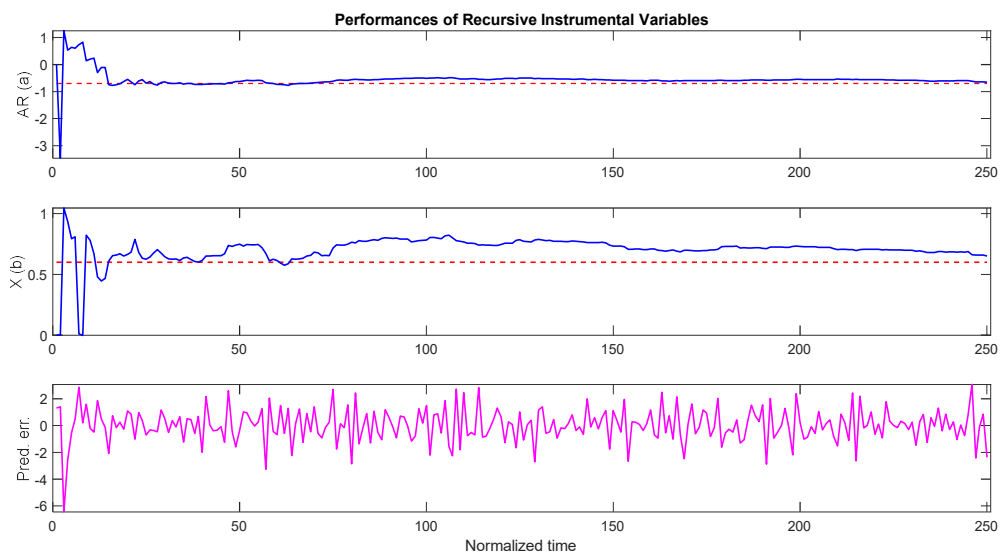


Fig. 10 – MVI - R – param constanti

Toate graficele arata la fel pentru orice valoare a lui alpha

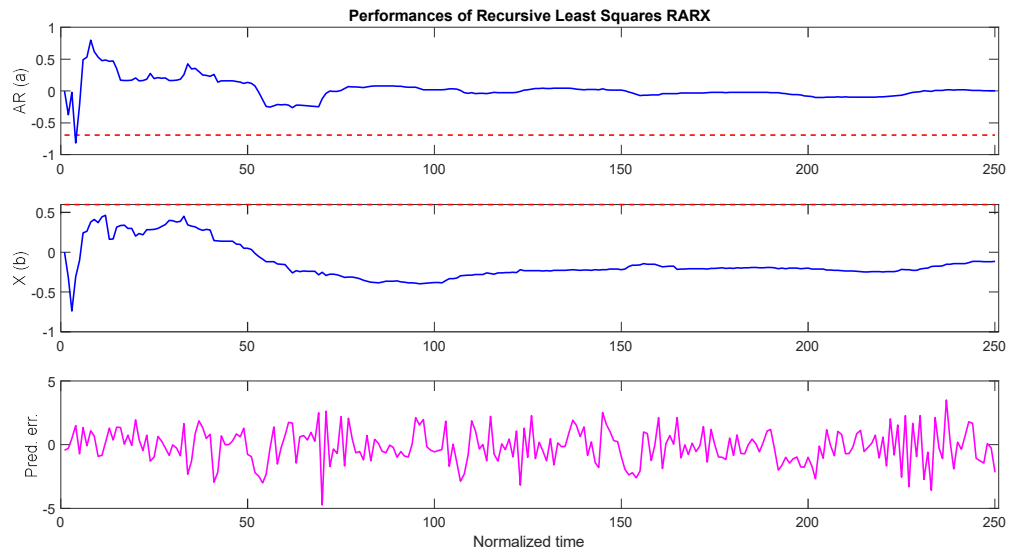


Fig. 11 – MCMMP - R – param variabili

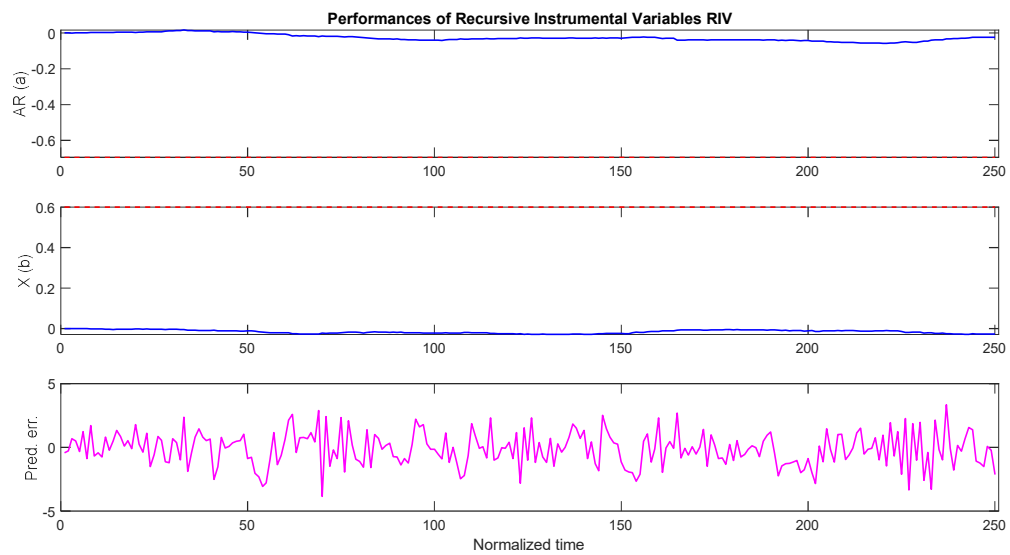


Fig. 12 – MVI - R – param variabili, alpha = 0.001

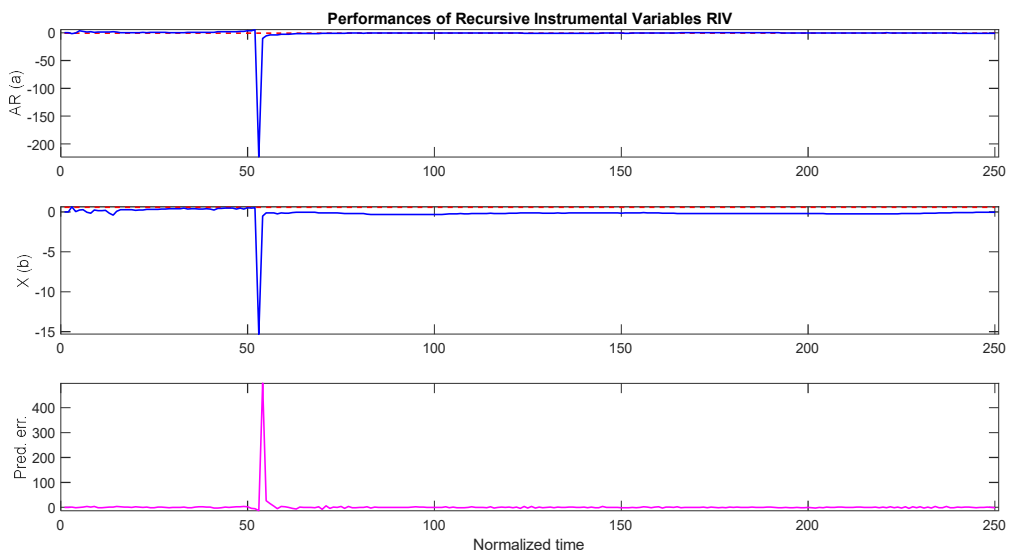


Fig. 13 – MVI - R – param variabili, alpha = 1

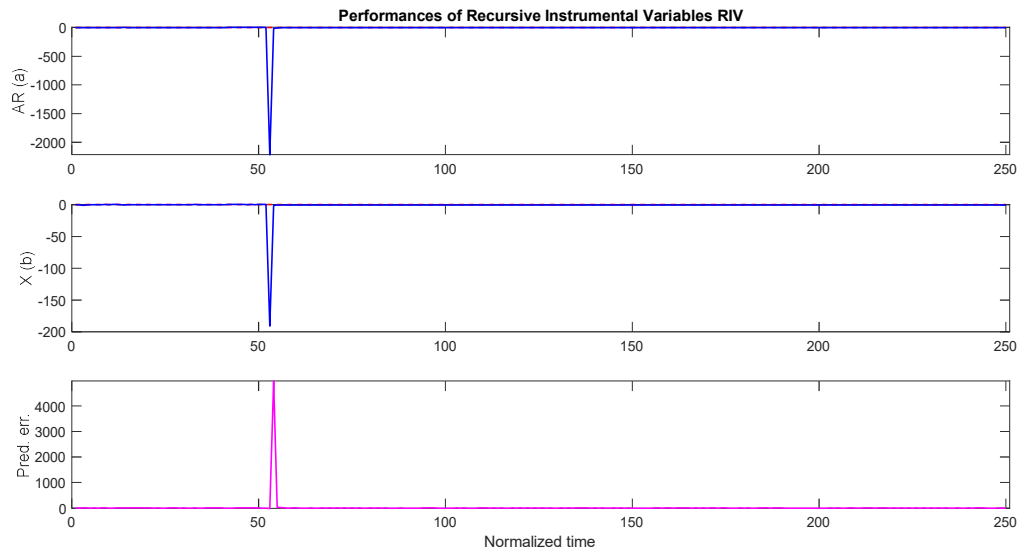


Fig. 14 – MVI - R – param variabili,  $\alpha = 1000$

### Concluzii:

Pentru metodele cu parametri constanti cele 7 grafice sunt la fel pentru orice  $\alpha$ . Acelasi lucru se observa si pentru MCMMP-R cu param variabili.

Pentru MVI-R cu parametrii variabili in schimb, graficele se schimba pentru fiecare valoare a lui  $\alpha$ .

La valori mici ale lui  $\alpha$ , cu cat valoarea lui  $\alpha$  creste mai mult, cu atat graficul se apropie mai mult de referinta. Dar atunci cand  $\alpha$  devine supraunitar graficul atinge un apogeu si ramane aproximativ constant indiferent de cat creste valoarea lui  $\alpha$ .

### PROBLEMA 3 (Identificare recursiva comparativa)

**Rutina arx\_nabla:** Se pleaca de la algoritmul prezentat la ISLAB\_7B si se modifica conform paginii 90 din laborator.

#### Algoritmul recursiv CMMP/VI de tip gradient

- **Date de intrare:**
- a. ordinele modelului de identificare:  $n_a$ ,  $n_b$ ,  $n_c$ ,  $n_d$  și  $n_f$ ;
  - b. câștigul de gradient:  $\gamma \in \mathbb{R}^+$ ;
  - c. o colecție redusă de date intrare-ieșire 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,  $\zeta$ , este identic cu vectorul regresorilor  $\phi$ . Altfel,  $\zeta$  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. Inițializare. 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 inițială 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  $P_0 = \gamma I_{n_\theta}$ .
  3. Pentru  $k \geq 1$ :
    - 3.1. Se evaluează eroarea de predicție:  $\varepsilon[k] = y[k] - \phi^T[k] \hat{\theta}_{k-1}$ .
    - 3.2. Se reactualizează matricea  $P_k$  astfel:  $P_k = P_0$  (gradient ne-normalizat) sau  $P_k = P_0 / \|\phi[k]\|^2$  (gradient normalizat).
    - 3.3. Se reactualizează vectorul parametrilor:  $\hat{\theta}_k = \hat{\theta}_{k-1} + P_k \phi[k] \varepsilon[k]$ .
- **Date de ieșire:** parametri  $\hat{\theta}_k$  ai modelului de identificare la fiecare pas de reactualizare  $k \geq 0$ .



%%% ARX\_NABLA

%1 Input

gain = input('Castigul de gradient = ');

[theta,ypred] = rarx(D,[1 1 1],'ng',gain) ; % Se schimba linia aici

% Definire phi (Date iesire, Date intrare)

%Calcul Parametri

phi = [D.y; D.u];

Rn = 0;

rn = 0;

for n = 1:N

    Rn = Rn+(1/N)\*(phi\*phi');

    rn = rn+(1/N)\*(phi\*D.y(n));

end

theta = Rn^(-1)\*rn ;

%2 Initializare

Priv = gain\*eye(length(theta)) ;

%3

for k = 1 :N

    eps(k) = D.y(k) - phi'\*theta;

    Priv = Priv/(norm(phi))^2;

    theta = theta + Priv\*phi\*eps(k);

end



**Rutina arx\_KB:** Se pleaca de la algoritmul prezentat la ISLAB\_7B si se modifica conform paginii 91 din laborator.

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

- Date de intrare:
- ordinele modelului de identificare:  $n_a$ ,  $n_b$ ,  $n_c$ ,  $n_d$  și  $n_f$ ;
  - matricea de răspîndire:  $R_v > 0$ ;
  - dispersia estimată a zgomotului alb de proces:  $\lambda^2$ ;
  - o colecție redusă de date intrare-ieșire 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);
  - un semnal instrumental extern:  $\{f[n]\}_{n \geq 1}$  (eventual).
- Dacă nu a fost specificat nici un semnal instrumental, vectorul variabilelor instrumentale,  $\zeta$ , este identic cu vectorul regresorilor  $\phi$ . Altfel,  $\zeta$  este definit ca în cazul MVI, dar folosind în general semnalul instrumental extern  $f$  în locul intrării  $u$  (în particular, este posibil ca  $f \equiv u$ ).
  - Inițializare. Fie se setează arbitrar vectorul parametrilor  $\hat{\theta}_0$  și matricea  $P_0 = \alpha I_{n_\theta}$  (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 inițială a parametrilor ( $\hat{\theta}_0$ ) folosind o metodă off-line adecvată modelului particular utilizat (din clasa MCMMP-MVI) și se egalează matricea  $P_0$  cu inversa matricii de covarianță  $R_0$  folosită în calculul lui  $\hat{\theta}_0$  (în cazul în care setul de date redus  $\mathcal{D}_{N_0}$  este disponibil).
  - Pentru  $k \geq 1$ :
    - Se evaluează eroarea de predicție:  $\varepsilon[k] = y[k] - \phi^T[k] \hat{\theta}_{k-1}$ .
    - Se evaluează vectorul auxiliar:  $\xi_k = P_{k-1} \zeta[k]$ .
    - Se evaluează cîștigul de sensibilitate:  $\gamma_k = \frac{\xi_k}{\lambda^2 + \phi^T[k] \xi_k}$ .
    - Se reactualizează matricea  $P_k$ , adică:  $P_k = P_{k-1} + R_v - \gamma_k \phi^T[k] P_{k-1}$  (cu evitarea inversării explicite a matricilor).
    - Se reactualizează vectorul parametrilor:  $\hat{\theta}_k = \hat{\theta}_{k-1} + \gamma_k \varepsilon[k]$ .
- Date de ieșire: parametrii  $\hat{\theta}_k$  ai modelului de identificare la fiecare pas de reactualizare  $k \geq 0$ .



%%% ARX\_KB

% Input

lambdaPatrat = 1 ;

[theta,ypred] = rarx(D,[1 1 1],'kf',Rv) ; %Se modifica aici

% Definire phi + Calcul Parametri

phi = [D.y; D.u];

Rn = 0;

rn = 0;

tau = phi;

for n = 1:N

    Rn = Rn+(1/N)\*(phi\*phi');

    rn = rn+(1/N)\*(phi\*D.y(n));

end

theta = Rn^(-1)\*rn ;

%2 Initializare

alpha = 1; %Se alege alpha

Priv = alpha\*eye(length(theta)) ;

Priv = Rn^(-1) ;

Rv = ones(size(Priv)) ;

%3

for k = 1 : N

    eps(k) = D.y(k) - phi'\*theta ;

    tzeta = Priv\*tau ;

    gain = tzeta/(lambdaPatrat + phi'\*tzeta);

    Priv = Priv + Rv - gain\*phi'\*Priv ;

    theta = theta + gain\*eps(k) ;

**ISLAB\_7D:** Se proiecteaza asemanator cu ISLAB\_7B, dar se adauga toate variantele:

```
disp('Lista celor 6 variante: ');
disp('1 = MCMMP-R cu parametri variabili');
disp('2 = MVI-R cu parametri variabili');
disp('3 = MMEP-R cu parametri variabili');
disp('4 = MRPL-R cu parametri variabili');
disp('5 = CMMP/VI de tip gradient');
disp('6 = CMMP/VI cu filtrare de tip Kalman-Bucy');

optiune = input('Alegere varianta: ') ;
```

Se foloseste switch si case pt toate variantele.

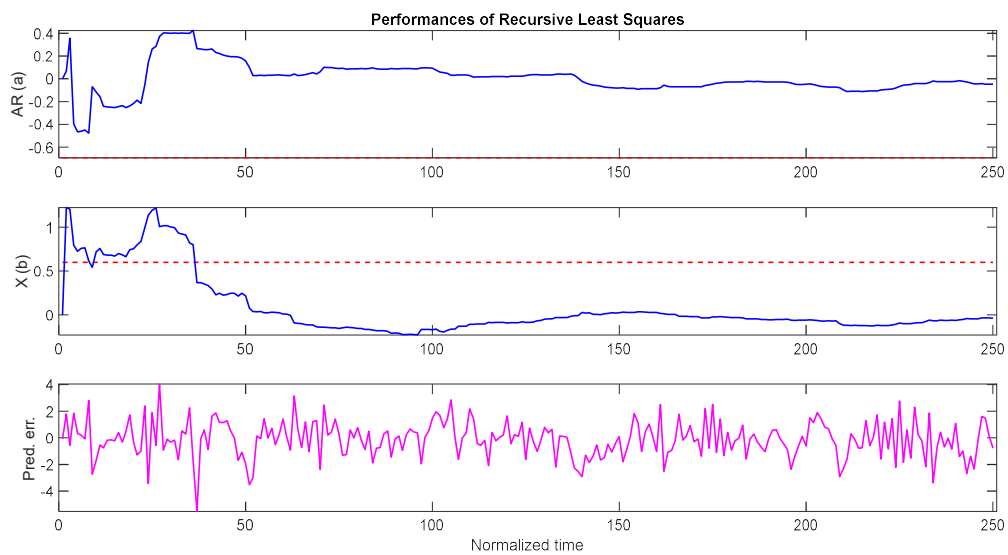


Fig. 15 – MCMMP – R cu param variabili

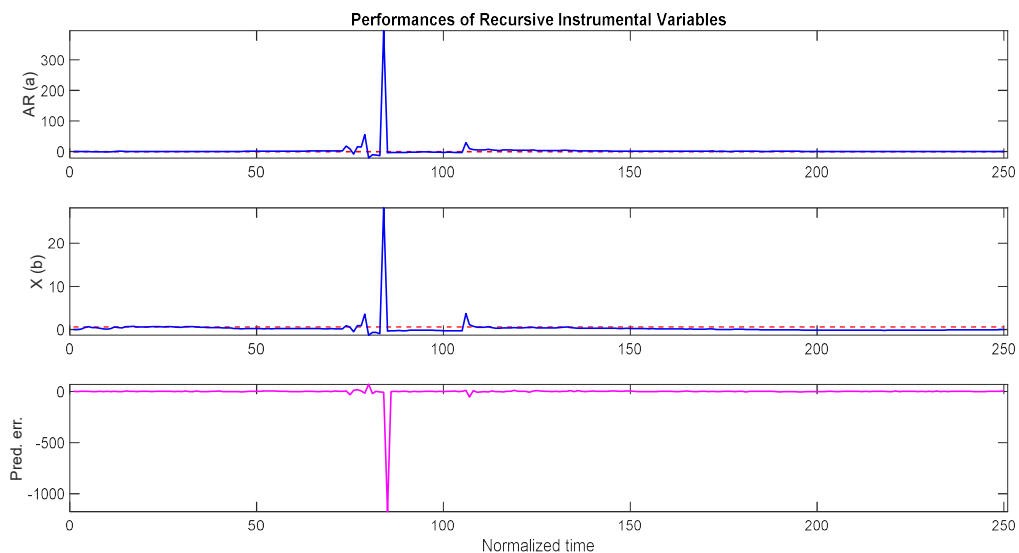


Fig. 16 – MVI – R cu param variabili

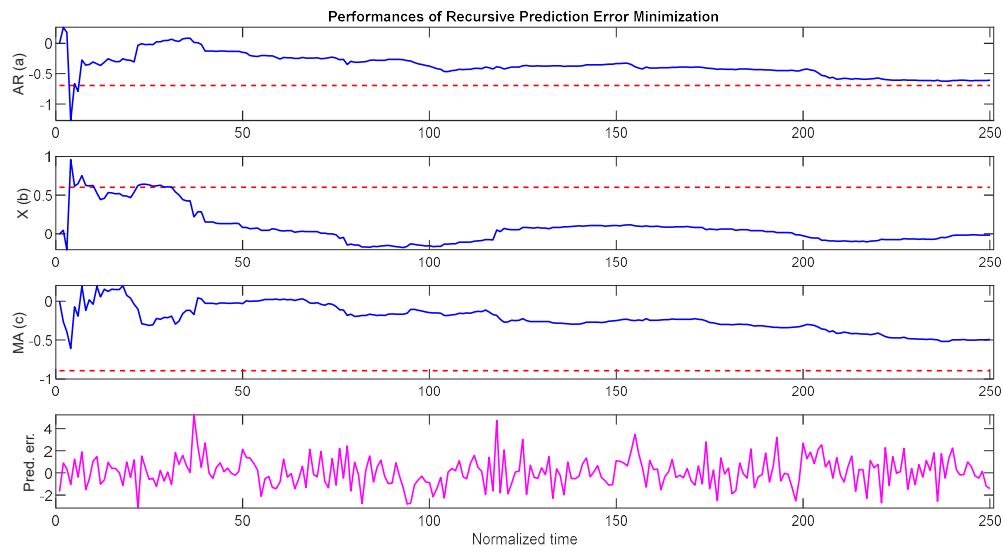


Fig. 17 – MMEP – R cu param variabili

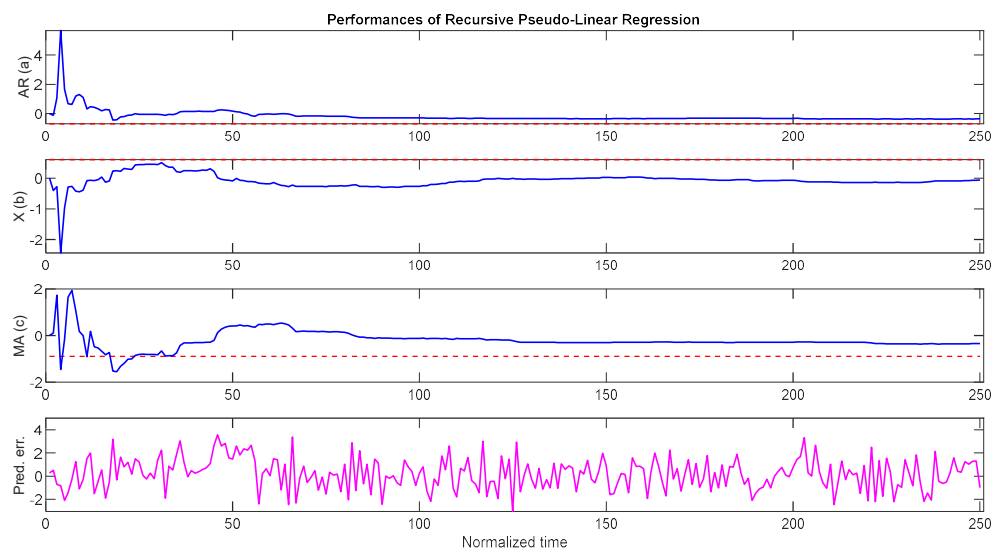


Fig. 18 – MRPL – R cu param variabili

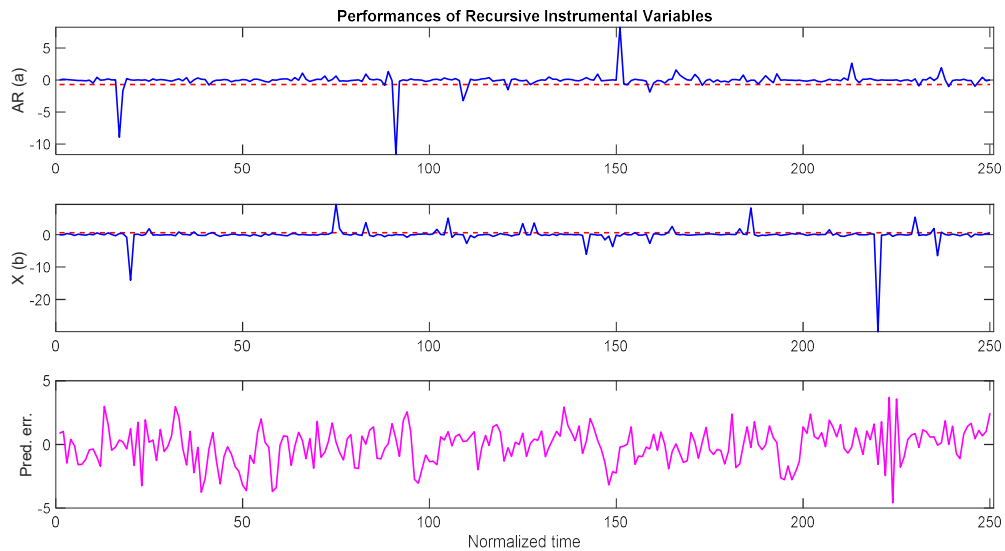


Fig. 19 – CMMP/VI de tip gradient cu castigul = 1

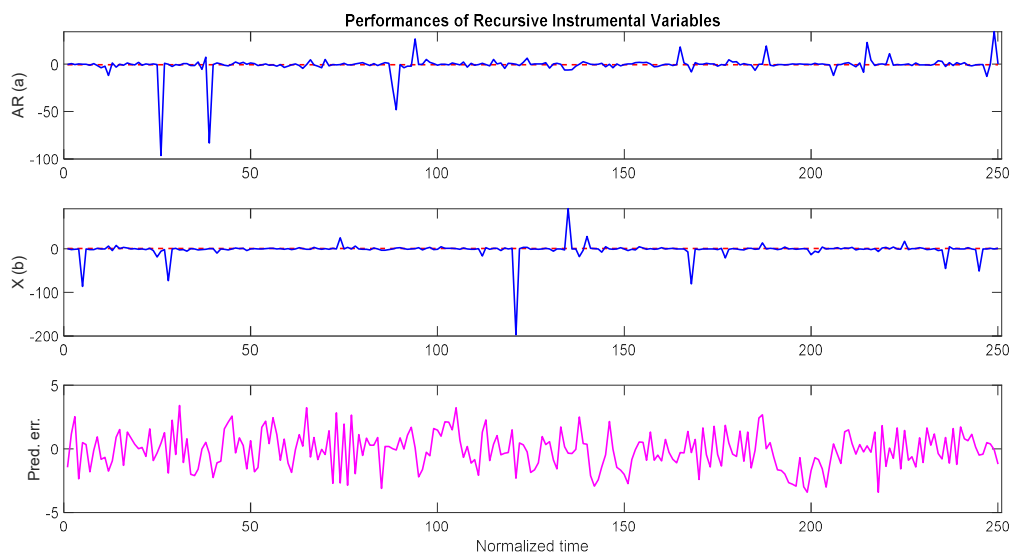


Fig. 20 – CMMP/VI cu filtrare de tip Kalman-Bucy cu  $\alpha = 1$

### Concluzii:

Se observa cum MCMMP are cea mai inconsistenta urmarire.

Cea mai buna urmarire o are CMMP/VI de tip gradient.

Si CMMP/VI de tip KB are o consistenta buna.

Asadar, opinia mea este ca ordinea solutiilor (de la cea mai buna la cea mai slaba) este:

CMMP/VI-gradient > CMMP/VI-KB > MVI-R > MRPL-R > MMEP-R > MCMMP-R.