

# PROIECT PARTEA 2

#5

COSTIN DENISA NICOLETA

# Introducere in ARX neliniar

Un model ARX neliniar este o extensie a modelului ARX liniar utilizat în identificarea sistemelor dinamice, după urmatoare formula:

ARX-liniar

$$y(k) = -a_1y(k-1) - a_2y(k-2) - \dots - a_n y(k-na) + b_1u(k-1) + b_2u(k-2) + \dots + b_n u(k-nb) + e(k)$$

ARX-neliniar

$$y(k) = g(y(k-1), y(k-2), \dots, y(k-na), u(k-1), u(k-2), \dots, u(k-nb); \theta) + e(k),$$

unde 'g' este funcția care determină parametrii

# Descrierea structurii approximatorului

Codul realizat implementează un model ARX neliniar care

1. Construiește matricea de regresori phi, incluzând termeni liniari și neliniari.
2. Utilizează metoda celor mai mici patrate pentru a estima parametrii theta, ai modelului. ->thetaid=phild\yid\_norm;
3. Simulează și validează modelul folosind date de intrare și ieșire.
4. Calculează eroarea medie pătratică (MSE) pentru a evalua performanța modelului.

```

%Funcție pentru construirea matricei phi cu termeni liniari și neliniari
function phi=MatriceaPHI(y,u,na,nb,m)
    if m>1
        termeni_neliniari=m*(m+1)/2+(m-1)*2; ...calculam numarul de termeni neliniari
    else
        termeni_neliniari=0;
    end
    phi=zeros(length(y),na+nb+termeni_neliniari);

    for k=1:length(y)
        ...Termeni liniari
        for nA=1:na
            if k>nA
                phi(k,nA)=-y(k-nA);
            end
        end
        for nB=1:nb
            if k>nB
                phi(k,na+nB)=u(k-nB);
            end
        end
        ...Termeni neliniari
        if m>1
            coloana=na+nb+1;
            for gr=2:m
                if k>gr-1
                    phi(k,coloana)=y(k-1)^gr; ...y(k-1)^gr
                    coloana=coloana+1;
                    phi(k,coloana)=u(k-1)^gr; ...u(k-1)^gr
                    coloana=coloana+1;
                    phi(k,coloana)=(y(k-1)*u(k-1))^(gr-1); ...combinatie intre u si y
                    coloana=coloana+1;
                end
                if k>gr
                    phi(k,coloana)=y(k-gr)*u(k-gr); ...y(k-gr)*u(k-gr)
                    coloana=coloana+1;
                    phi(k,coloana)=y(k-gr)*u(k-1); ...y(k-gr)*u(k-1)
                    coloana=coloana+1;
                end
            end
        end
    end
end

```

Termeni Liniari:

$-y(k-1), \dots, -y(k-na)$  - autoregresori ai ieșirii.  
 $u(k-1), \dots, u(k-nb)$  - regresori ai intrării.

Termeni Neliniari (dacă  $m > 1$ ):

Puteri de ordin superior ale valorilor anterioare ale ieșirii:  
 $y(k-1)^2, \dots, y(k-1)^m$ .

Puteri de ordin superior ale valorilor anterioare ale intrării:  
 $u(k-1)^2, \dots, u(k-1)^m$ .

Combinări neliniare între ieșire și intrare:  $y(k-1) \cdot u(k-1)$  și

# Caracteristici esentiale ale solutiei noastre

Am ales sa cream o functie care genereaza parametrii 'theta' pentru a facilita intelegerarea codului si a utilitatii acestei functii de catre orice utilizator.

Am calculat MSE urile pentru a vedea cat de aproape suntem de rezultatele dorite.

Pentru a centra datele in jurul valorii zero am folosit 'detrend', astfel inbunatatim stabilitatea numerica a algoritmului de regresie.

```
Interval uid: [0, 26.0459]
Interval yid: [1.2689, 4.0564]
Interval uid_norm: [-2.6233, 2.1919]
Interval yid_norm: [-2.705, 1.3408]
```

```

% Calcularea MSE-urilor
MSEid=calculeaza_mse(yid,ypredictieId);
MSEval=calculeaza_mse(yval,ypredictieVal);
MSEsimid=calculeaza_mse(yid_norm,ysimulareId);
MSEsimval=calculeaza_mse(yval_norm,ysimulareVal);
% Plotare rezultate mse
figure;
subplot(211);
plot(1:na_max,MSEid,'-o'); grid on;
title('MSE pentru identificare');
xlabel('na'); ylabel('MSE');
subplot(212);
plot(1:na_max,MSEval,'-o'); grid on;
title('MSE pentru validare');
xlabel('na'); ylabel('MSE');

figure;
subplot(211);
plot(1:na_max, MSEsimid,'-o'); grid on;
title('MSE pentru simulare identificare');
xlabel('na'); ylabel('MSE');
subplot(212);
plot(1:na_max,MSEsimval,'-o'); grid on;
title('MSE pentru simulare validare');
xlabel('na'); ylabel('MSE');

function MSE=calculeaza_mse(real,predictie)
    e=real-predictie;
    MSE=mean(e.^2);
end

```

# MSE

MSE măsoară diferența dintre valorile reale și cele estimate de model.

MSEid: Eroarea pentru datele de identificare prezise.

MSEval: Eroarea pentru datele de validare prezise.

MSEsimid: Eroare pentru simulare pe datele de identificare.

MSEsimval: Eroare pentru simulare pe datele de validare.

Valori mici ale MSE indică o performanță bună a modelului.

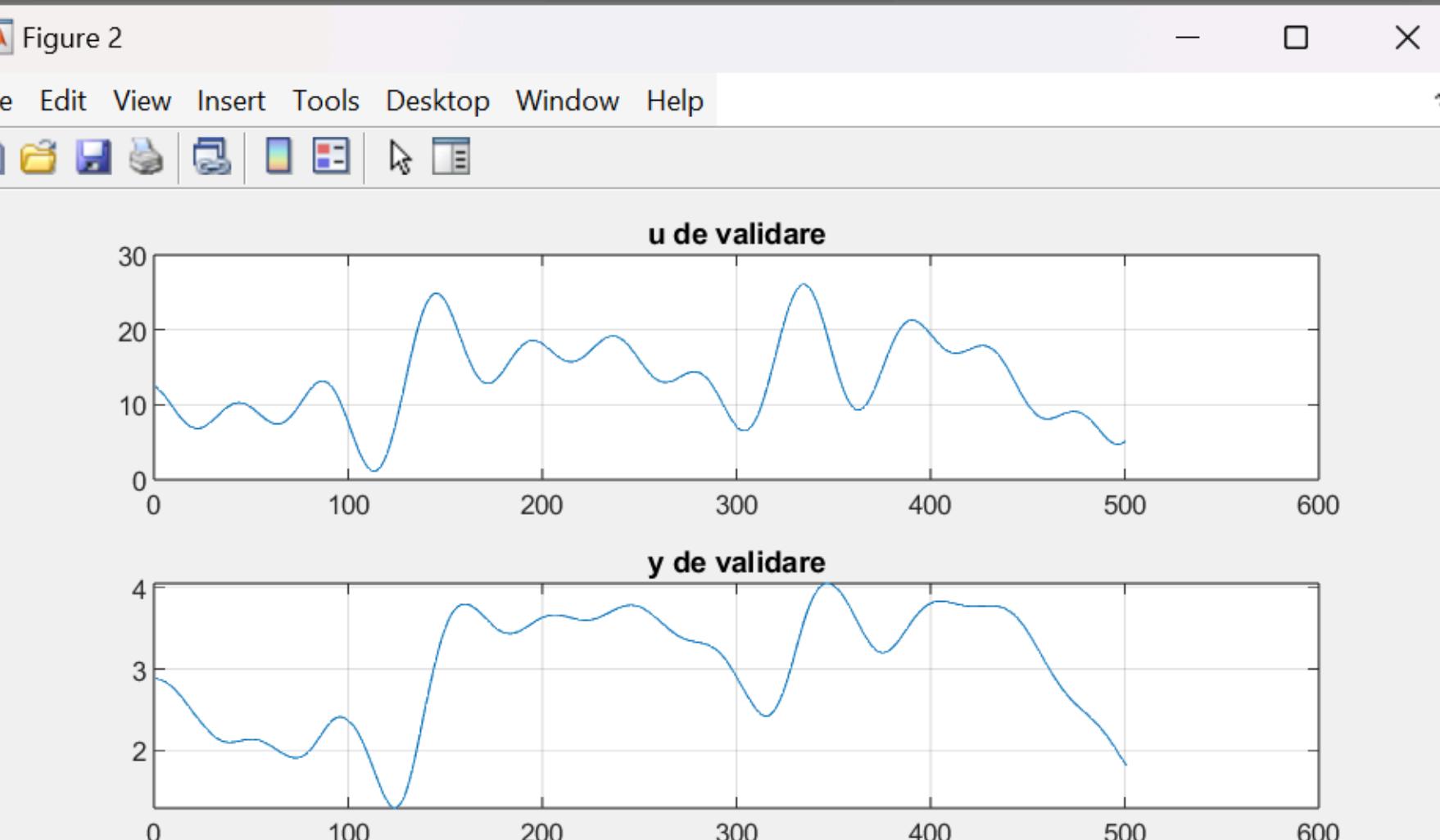
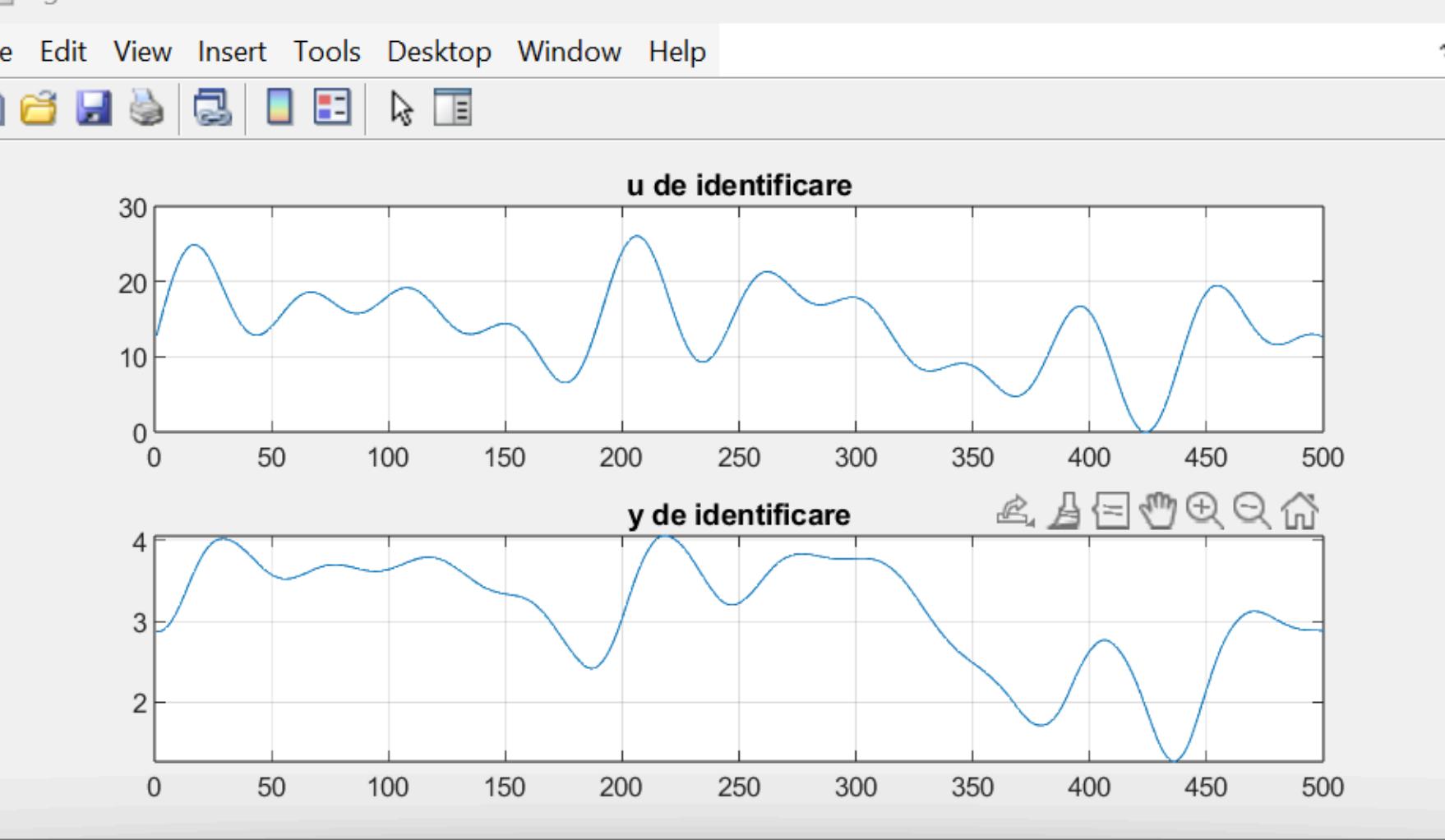
na=nb=4;  
m=2;

```

Mse identificare:0.0001353 0.00013519 0.00013494 0.00013493 0.00013487
Mse validare:4.6376e-05 4.5521e-05 4.4872e-05 4.4977e-05 4.6648e-05
Mse pe simulare identificare:0.0035606 0.0035115 0.0039223 0.0039197 0.0039321
Mse pe simulare validare:0.0072121 0.0075649 0.007203 0.0071993 0.0072372

```

Figure 1



# Afisarea datelor initiale

```

figure
subplot(211);
plot(uid); grid on;
title('u de identificare');
subplot(212);
plot(yid); grid on;
title('y de identificare');

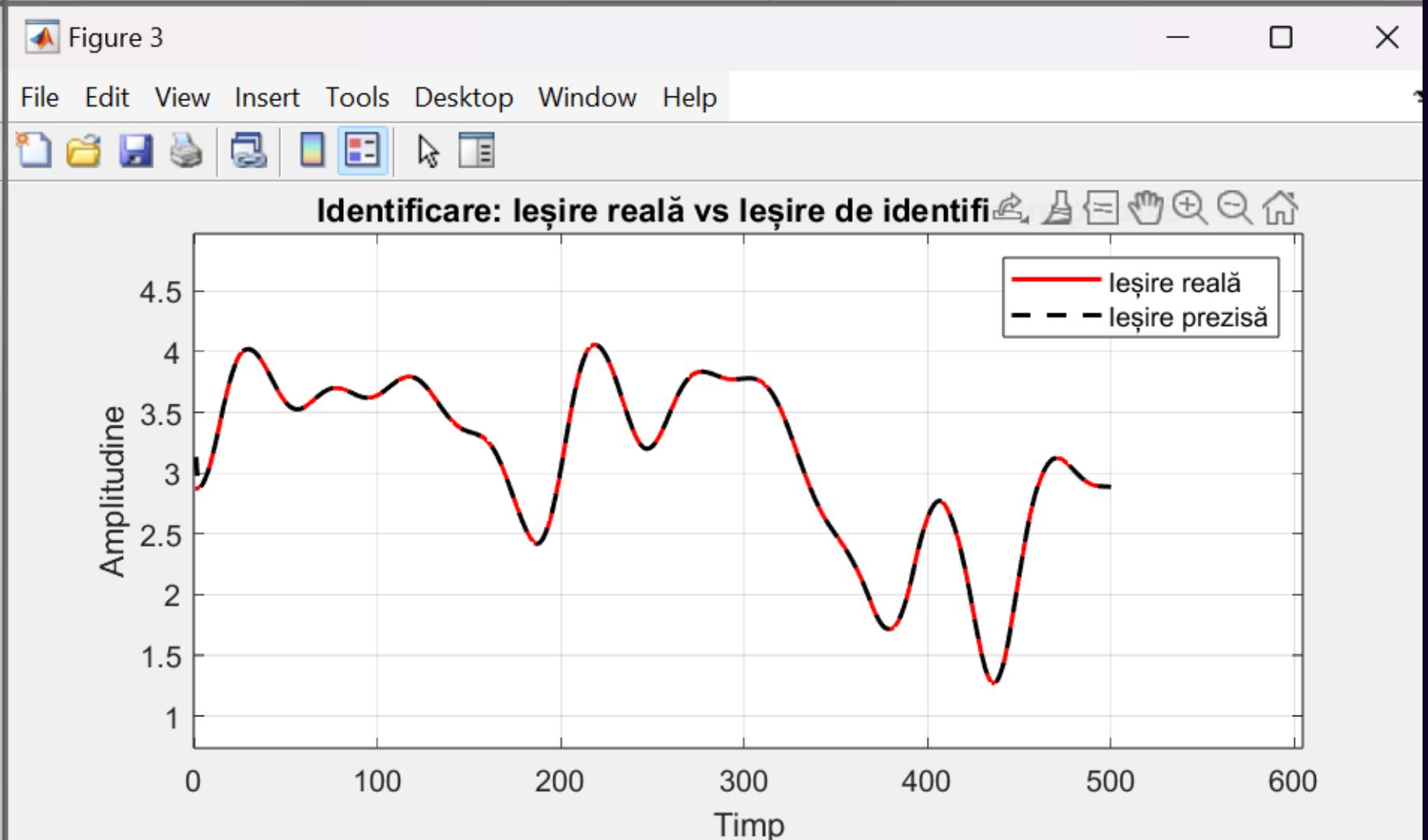
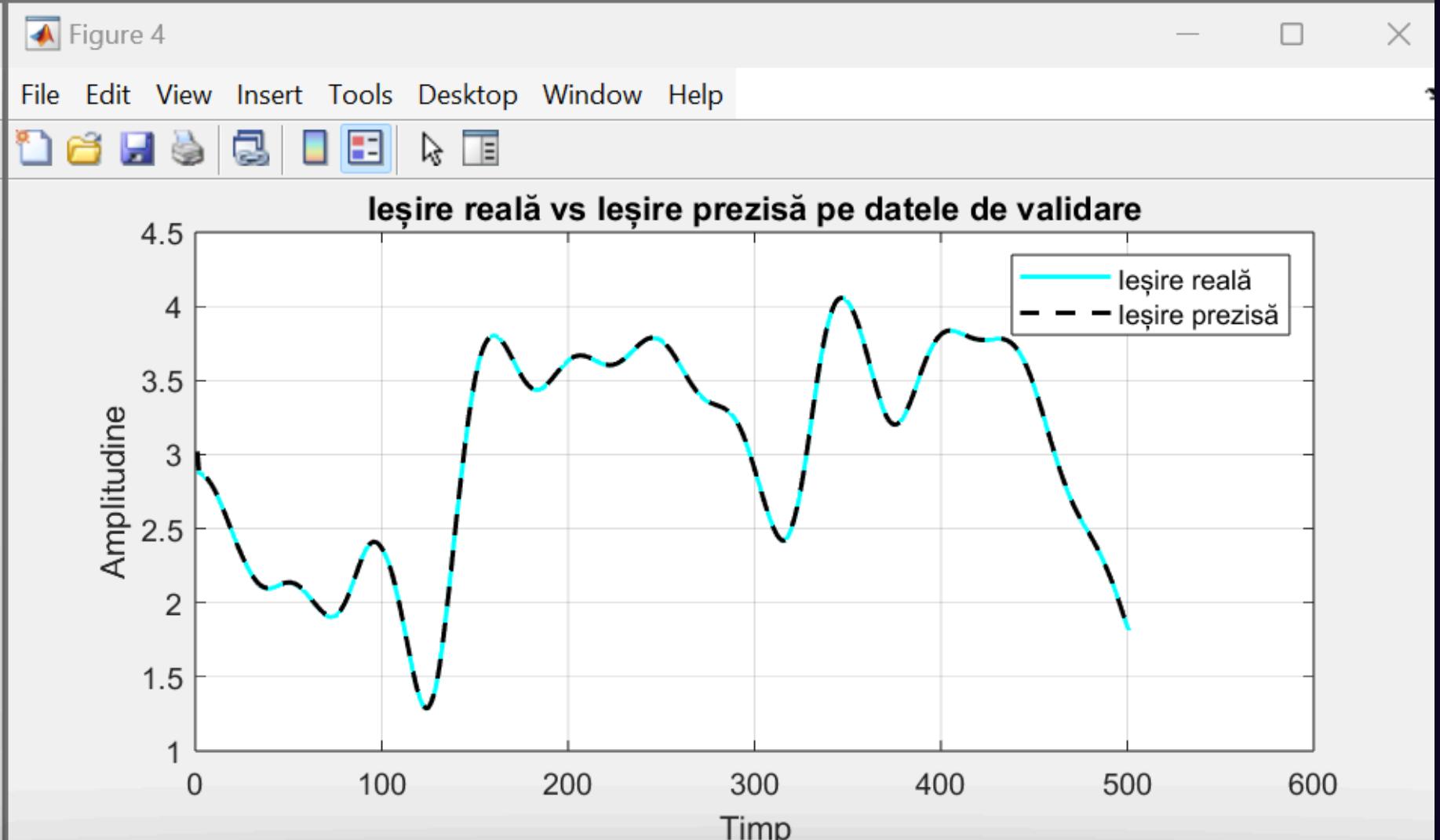
figure
subplot(211);
plot(uval); grid on;
title('u de validare');
subplot(212);
plot(yval); grid on;
title('y de validare');

%%
disp(['Interval uid:[', num2str(min(uid)), ',', num2str(max(uid)), ']']);
disp(['Interval yid:[', num2str(min(yid)), ',', num2str(max(yid)), ']']);

uid_norm =detrend(uid);
uval_norm =detrend(uval);
yid_norm= detrend(yid);
yval_norm=detrend(yval);

disp(['Interval uid_norm:[',num2str(min(uid_norm)), ',',num2str(max(uid_norm)), ']']); ... se verificare interval
disp(['Interval yid_norm:[',num2str(min(yid_norm)), ',',num2str(max(yid_norm)), ']']);

```



# Calcularea si Afisarea predictiei datelor

%Parametrii modelului

na=5;

nb=5;

m=3;

phiId=MatriceaPHI(yid\_norm,uid\_norm,na,nb,m);

thetaId=phiId\yid\_norm;

ypredictieId=phiId\*thetaId;

phiVal=MatriceaPHI(yval\_norm,uval\_norm,na,nb,m);

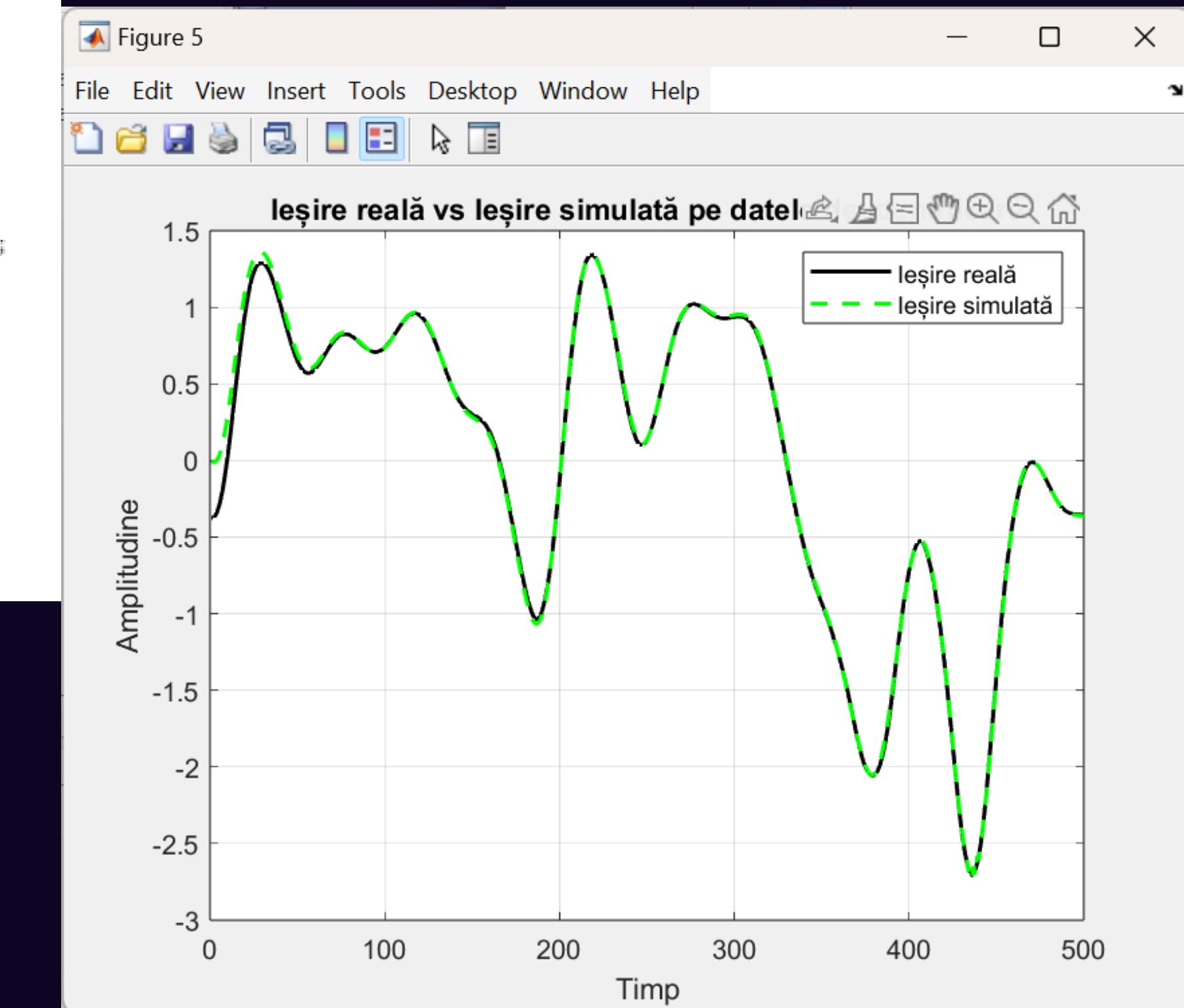
ypredictieVal=phiVal\*thetaId;

```

%Generam simularea pentru identificare
ysimulareId=zeros(length(uid_norm),1);
for k=1:length(uid_norm)
    for i=1:na
        if k>i
            ysimulareId(k)=ysimulareId(k)-thetaid(i)*ysimulareId(k-i);
        end
    end
    for j=1:nb
        if k>j
            ysimulareId(k)=ysimulareId(k)+thetaid(na+j)*uid_norm(k-j);
        end
    end
end
if m>1
    coloana=na+nb+1;
    for gr=2:m
        if k>gr-1
            ysimulareId(k)=ysimulareId(k)+thetaid(coloana)*ysimulareId(k-1)^gr;
            coloana=coloana+1;
            ysimulareId(k)=ysimulareId(k)+thetaid(coloana)*uid_norm(k-1)^gr;
            coloana=coloana+1;
            ysimulareId(k)=ysimulareId(k)+thetaid(coloana)*(ysimulareId(k-1)*uid_norm(k-1))^(gr-1);
            coloana=coloana+1;
        end
        if k>gr
            ysimulareId(k)=ysimulareId(k)+thetaid(coloana)*ysimulareId(k-gr)*uid_norm(k-gr);
            coloana=coloana+1;
            ysimulareId(k)=ysimulareId(k)+thetaid(coloana)*ysimulareId(k-gr)*uid_norm(k-1);
            coloana=coloana+1;
        end
    end
end
end

```

## Simularea datelor de identificare

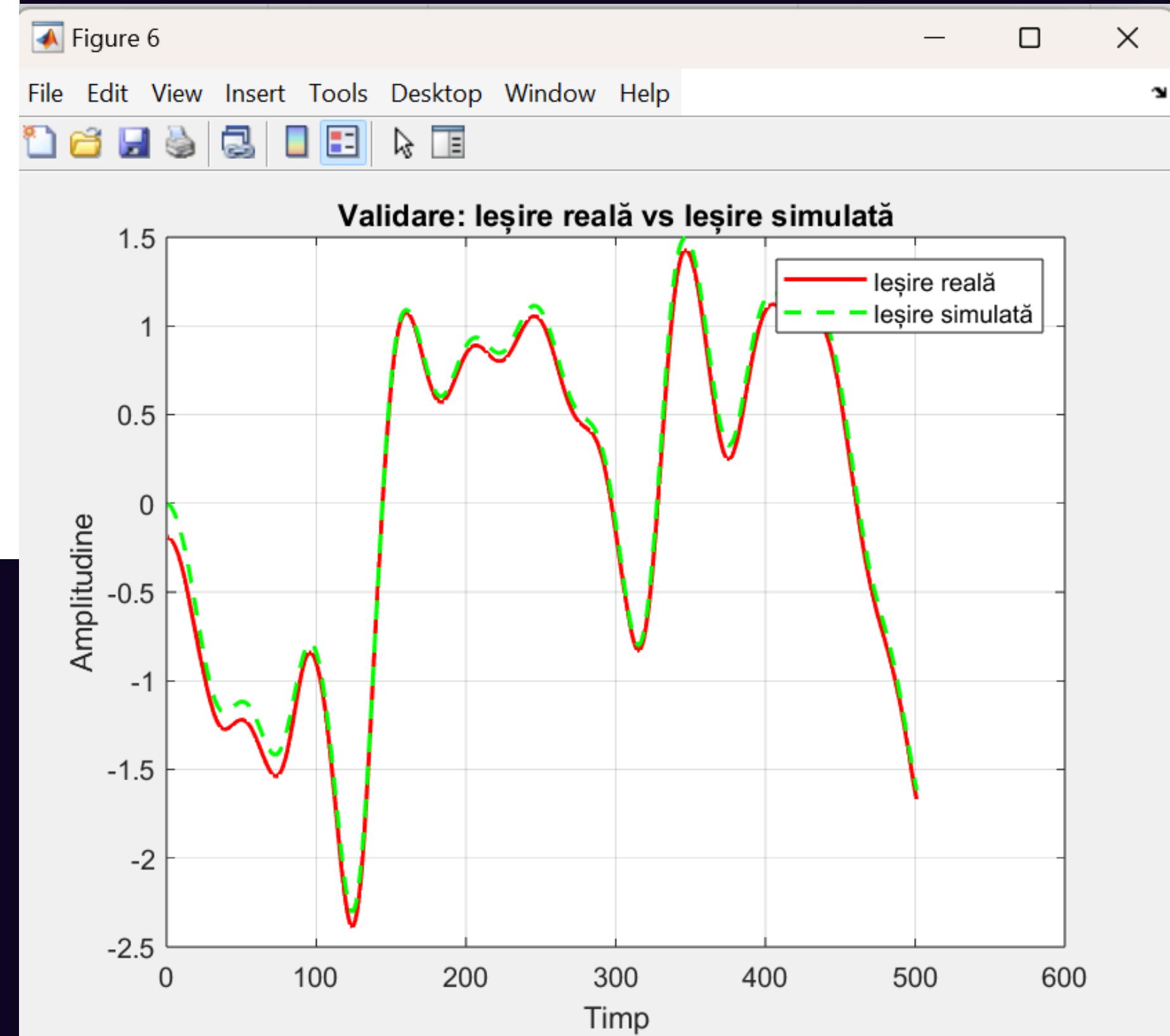


```

%Facem simularea pentru validare
ysimulareVal=zeros(length(uval_norm),1);
for k=1:length(uval_norm)
    for i=1:na
        if k>i
            ysimulareVal(k)=ysimulareVal(k)-thetaid(i)*ysimulareVal(k-i);
        end
    end
    for j=1:nb
        if k>j
            ysimulareVal(k)=ysimulareVal(k)+thetaid(nb+j)*uval_norm(k-j);
        end
    end
end
if m>1
    coloana=na+nb+1;
    for gr=2:m
        if k>gr-1
            ysimulareVal(k)=ysimulareVal(k)+thetaid(coloana)*ysimulareVal(k-1)^gr;
            coloana=coloana+1;
            ysimulareVal(k)=ysimulareVal(k)+thetaid(coloana)*uval_norm(k-1)^gr;
            coloana=coloana+1;
            ysimulareVal(k)=ysimulareVal(k)+thetaid(coloana)*(ysimulareVal(k-1)*uval_norm(k-1))^(gr-1);
            coloana=coloana+1;
        end
        if k>gr
            ysimulareVal(k)=ysimulareVal(k)+thetaid(coloana)*ysimulareVal(k-gr)*uval_norm(k-gr);
            coloana=coloana+1;
            ysimulareVal(k)=ysimulareVal(k)+thetaid(coloana)*ysimulareVal(k-gr)*uval_norm(k-1);
            coloana = coloana+1;
        end
    end
end
end

```

## Simularea datelor de validare



## Valorile MSE-Rezultate in functie de m

m	1	2	3	4	5	6
MSEid	0.000317837	0.000285604	0,000284297	0,000284071	0,000283192	0,000282769
MSEva	0,00120987	0,000879805	0,000852087	0,000847454	0,000844099	0,000838844
MSEsimId	0,0161411	0,0114598	0,0122318	0,0130368	0,0149344	0,0181693
MSEsimVal	0,0510381	0,0146776	0,015616	0,0166292	0,0178053	0,0210216

Pentru MSEid, cea mai potrivita valoare a polinomului este m=6

Pentru MSEval, cea mai potrivita valoare a polinomului este m=6

Pentru MSEsimId, cea mai potrivita valoare a polinomului este m=2

Pentru MSEsimVal, cea mai potrivita valoare a polinomului este m=2

# Rezultate MSE in functie de na

na	1	2	3	4	5
MSEsimId	0,00356057	0,00351149	0,00392226	0,00391969	0,00393207
MSEsimVal	0,00721212	0,00756489	0,00720295	0,00719926	0,00723722
MSEid	0,000135304	0,000135192	0,000134938	0,000134935	0,000134871
MSEval	0,000463757	0,000455212	0,00048725	0,000449766	0,000466478

