

# Arx neliniar

Studenti: Cimuca Denisa  
Isărescu Anamaria  
Paștiu Anamaria

Grupa: 30135

**Proiect: Pid 7**





# Cuprins

1. Ipoteză
2. Descrierea metodei
3. Date teoretice
4. Rezultate
5. Concluzie



# Ipoteza

Scopul acestei lucrări este de a determina un algoritm care generează un model ARX neliniar de tip polinomial, pentru parametrii  $n_a$ ,  $n_b$ ,  $m$  configurabili. Modelul realizat trebuie apoi validat comparând modelul cu datele de validare inițiale.



## Descrierea metodei

Pas1: Determinarea matricii de puteri necesară pentru determinarea tuturor combinațiilor ale polinomului neliniar

Pas2: Aflarea matricii de regresori **Phi** și a coeficienților **theta**

Pas3: Predicția modelului

Pas4: Simularea modelului

Pas5: Calcularea MSE și determinarea valorilor optime pentru  $n_a$ ,  $n_b$ ,  $m$



# Date teoretice

Structura ARX neliniar:

$$X(k) = p(y(k-1), \dots, y(k-na), u(k-1), u(k-1), \dots, u(k-nb)),$$

unde  $p$  este polinomul de grad  $m$

-  $na$  și  $nb$  sunt ordinele modelului ARX

Matricea **phi** are toate combinațiile din polinomul  $p$  în funcție de gradul  $m$

Calculăm  $\theta$ :  $\theta = \phi \backslash y_{id}$ ;

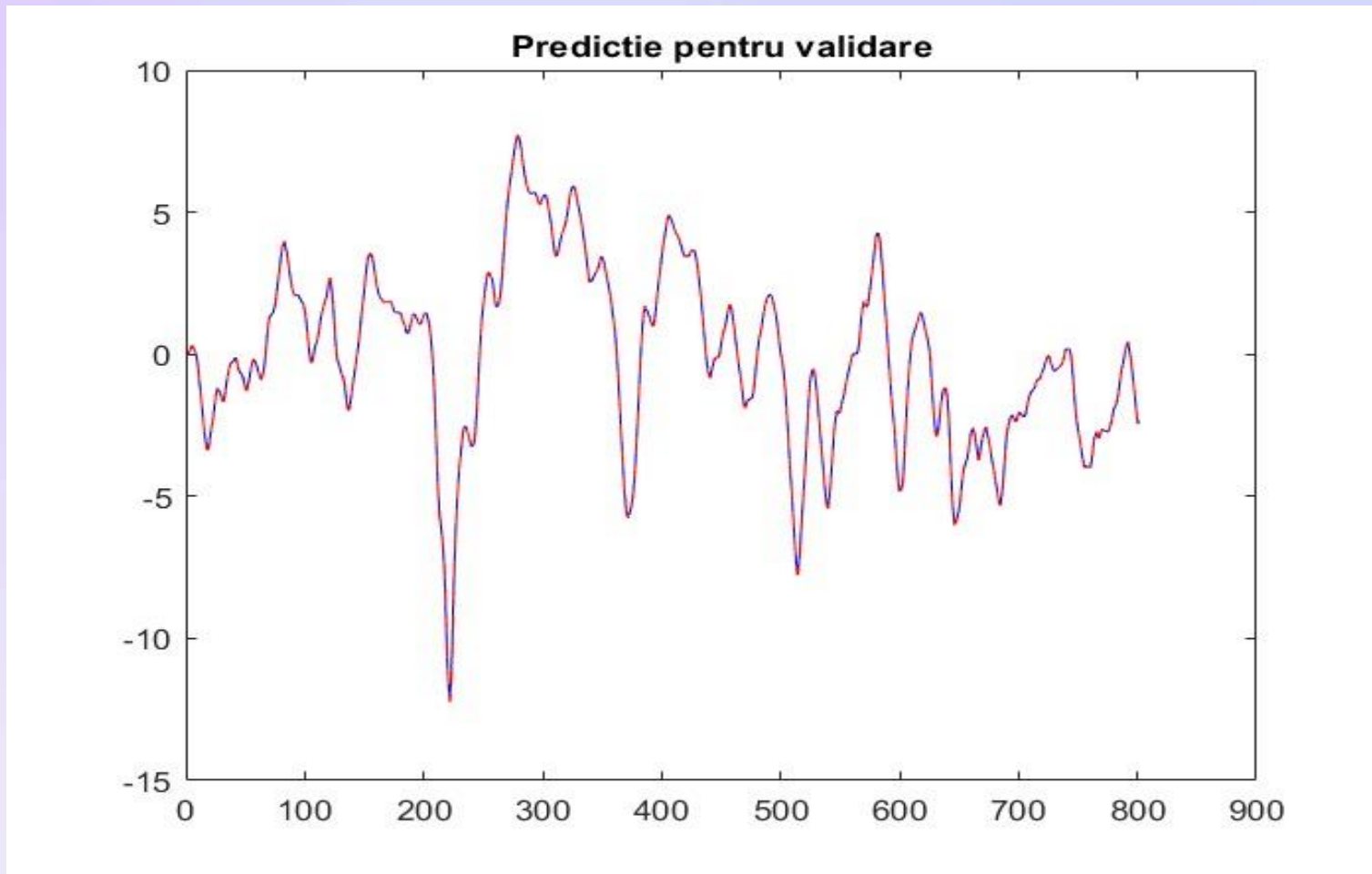
Predicția:  $y_{hat\_val} = PHI\_val * \theta$ ;

Simularea: pentru calcularea ieșirilor simulate s-au folosit ieșirile anterioare simulate și intrările anterioare

```
suma=0;
for a=1:length(power)
p=1;
for i=1:(na+nb)
aux=X_simulare(k,i).^power(a,i);
p=aux*p;
end
suma=suma+p*theta(a);
end
ysimulare(k)=suma;
```

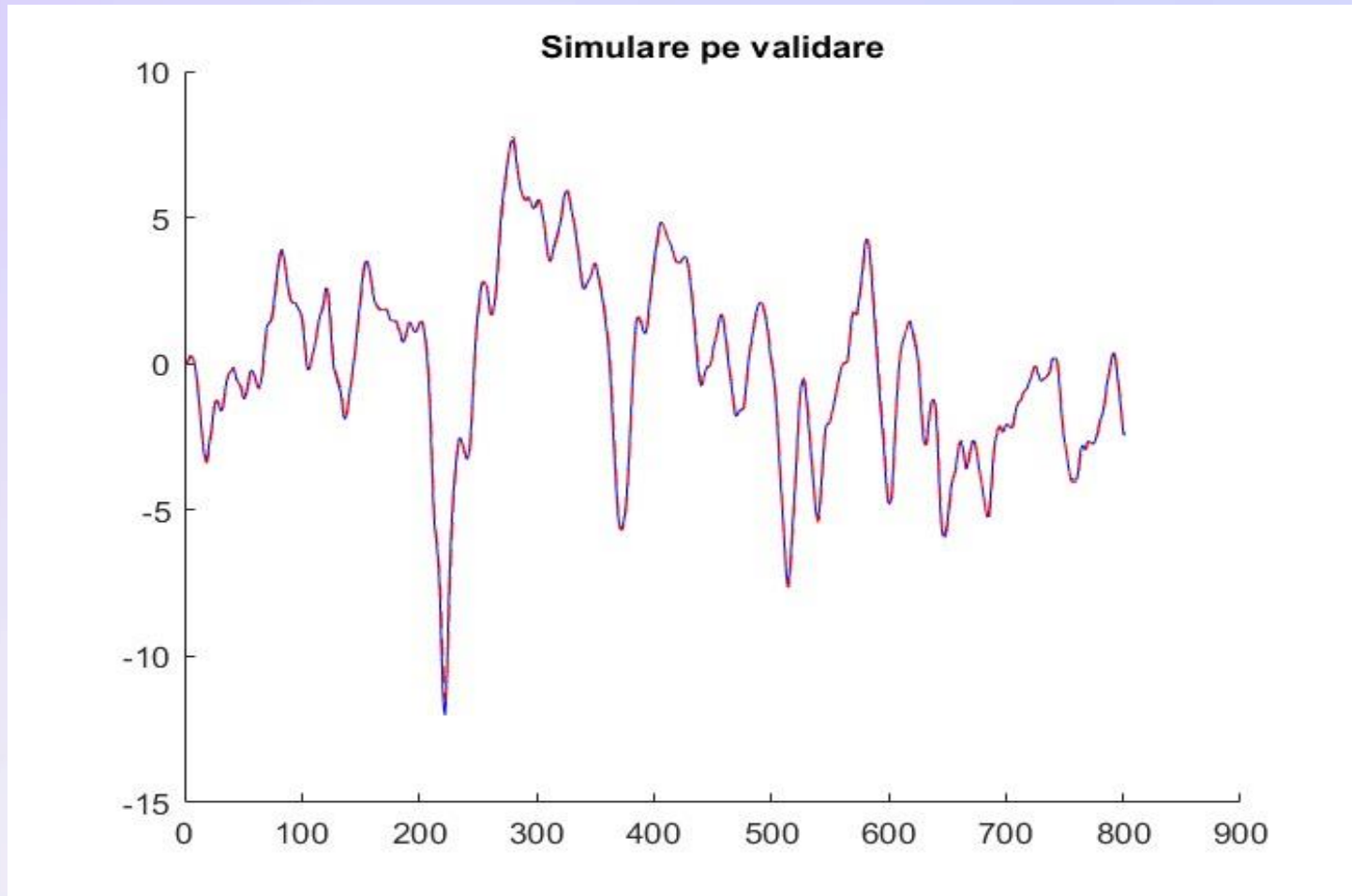
# Rezultate

A: Predictia



# Rezultate

## B: Simularea



# MSE

	m	na	nb	Mse_pred	Mse_sim
1	1	1	1	0.0172	0.2763
2	1	1	2	0.0171	0.2770
3	1	1	3	0.0171	0.2777
4	2	1	1	0.0067	0.0554
5	2	1	2	0.0065	0.0555
6	2	1	3	0.0064	0.0568
7	3	1	1	0.0045	0.0380
8	3	1	2	0.0043	0.0388
9	3	1	3	0.0043	0.0394



# Concluzie

În urma mai multor încercări a rezultat

- gradul optim  $m = 3$ , valorile optime  $n_a = 1$  și  $n_b = 3$  pentru predicție

- gradul optim  $m = 3$ , valorile optime  $n_a = 1$  și  $n_b = 1$  pentru simulare

Astfel pe baza acestor valori am determinat modelul ARX neliniar pentru care erorile medii pătratice sunt minime atât pentru predicție cât și pentru simulare.

**MULȚUMIM  
PENTRU  
ATENȚIE!**




# Anexă

```
load('iddata-07.mat');
u_id = id.InputData;
y_id = id.OutputData;
u_val = val.InputData;
y_val = val.OutputData;

figure
plot(u_id);
title("Intrarea de
identificare");
figure
plot(y_id);
title("Iesirea de
identificare");
figure
plot(u_val);
title("Intrarea de
validare");
figure
plot(y_val);
title("Iesirea de validare");
```

```
MSE_predictie = [];
MSE_simulare = [];
mat_parametri = [];
for m = 1:3
    for na = 1:1
        for nb = 1:3
            n = na + nb;
            N = length(u_id);
            x1_id = [];
            x2_id = [];
            for k = 1:N
                for i = 1:na
                    if k - i <= 0
                        x1_id(k,i) = 0;
                    else
                        x1_id(k,i) = y_id(k - i);
                    end
                end
                for i = 1:nb
                    if k - i <= 0
                        x2_id(k,i) = 0;
                    else
                        x2_id(k,i) = u_id(k - i);
                    end
                end
            end
        end
    end
end
```



```

x1_val = [];
x2_val = [];
for k = 1:length(u_val)
    for i = 1:na
        if k - i <= 0
            x1_val(k,i) = 0;
        else
            x1_val(k,i) = y_val(k - i);
        end
    end
    for i = 1:nb
        if k - i <= 0
            x2_val(k,i) = 0;
        else
            x2_val(k,i) = u_val(k - i);
        end
    end
end

X_id = cat(2,x1_id,x2_id);
X_val = cat(2,x1_val,x2_val);


```


```

vars = cell(n,1);
[vars{1:n}] = ndgrid(0:m);
power = reshape(cat(n+1, vars{:})), [], n);
power = power(sum(power,2) <= m & sum(power,2)
> 0, :);
power(end + 1,:) = 0;

PHI_id = [];
PHI_val = [];
b = 1;
for a = 1:N
    phi_id = [];
    %for pentru puteriile indicilor
    for k = 1:length(power)
        for i = 1:n
            aux = (X_id(a,i)^(power(k,i))) * b;
            b = aux;
        end
        phi_id(end+1) = aux;
        b = 1;
    end
    PHI_id = [PHI_id;phi_id];
end

```





```

c=1;
for a = 1:length(u_val)
    phi_val = [];
    %for pentru puteriile indicilor
    for k = 1:length(power)
        for i = 1:(na+nb)
            aux =
(X_val(a,i)^(power(k,i)))*c;
            c = aux;
        end
        phi_val(end + 1) = aux;
        c = 1;
    end
    PHI_val = [PHI_val; phi_val];
end
theta = PHI_id\y_id;
y_hat_id = PHI_id*theta;
y_hat_val = PHI_val*theta;

mse_predictie = sum((y_hat_val-
y_val).^2)/length(y_val);
MSE_predictie(end+1) = mse_predictie;


ysimulare = zeros(1,length(u_val));
x1_simulare = [];
x2_simulare = [];


```

```

for k = 1:length(u_val)
    for i = 1:na
        if k - i <= 0
            x1_simulare(k,i) = 0;
        else
            x1_simulare(k,i) =
ysimulare(k-i);
        end
    end
    for i = 1:nb
        if k - i <= 0
            x2_simulare(k,i) = 0;
        else
            x2_simulare(k,i) = u_val(k
- i);
        end
    end
    X_simulare =
cat(2,x1_simulare,x2_simulare);
    suma = 0;
    for a = 1:length(power)
        p = 1;
        for i = 1:(na+nb)
            aux =
X_simulare(k,i).^power(a,i);
            p = aux*p;
        end
    end

```





```

suma = suma + p * theta(a);
    end
    ysimulare(k) = suma;
end

mat_parametri(end+1,:) = [m na nb];
mse_simulare = sum((ysimulare-
y_val').^2)/length(y_val);
MSE_simulare(end+1) = mse_simulare;
    end
end
end

%MSE uri
[mse_pred_min,index_pred_min] =
min(MSE_predictie)
disp("mat_parametri pentru mse predictie
minim [m na nb]")
mat_parametri(index_pred_min,:)
[mse_sim_min,index_sim_min] =
min(MSE_simulare)
disp("mat_parametri pentru mse simulare
minim [m na nb]")
mat_parametri(index_sim_min,:)

```

```

%Afisari
%Predictie
figure
plot(y_id,'b');
hold on
plot(y_hat_id,'--r');
title("Predictie pentru identificare");
figure
plot(y_val,'b');
hold on
plot(y_hat_val,'--r');
title("Predictie pentru validare");
%Simulare
figure
hold on
plot(y_val,'b');
plot(ysimulare,'--r');
title("Simulare pe validare");

```

