PROIECT LA IDENTIFICAREA SISTEMELOR

IDENTIFICAREA UNUI SISTEM BAZAT PE RĂSPUNS LA SINUS

Student: Lascu Narcis-Dumitru Coordonator:

Grupa: 30131 Prof. Dr. Ing. Petru Dobra

CUPRINS

1. Identificarea neparametrică a sistemului fara zero cu ajutorul defazajului…….....................3

2. Identificarea parametrică a sistemului fara zero......................................................................9

2.1 Metoda celor mai mici pătrate extinsă (ARMAX)................................................................9

2.2 Metoda erorii la iesire (OE)………….……………………………………………………11

3. Identificarea parametrică a sistemului cu zero......................................................................13

3.1 Metoda celor mai mici pătrate extinsă (ARMAX)..............................................................14

3.2 Metoda erorii de iesire (OE)................................................................................................16

3.3 Metoda variablielor instrumentale(IV) ................................................................................18

3.4 Aplicarea filtrului FIR..........................................................................................................20

4. Concluzie................................................................................................................................22

1. **Identificarea neparametrică a sistemului prin gasirea defasajului de 90 de grade intre intrare si iesire.**

În MatLAB vom încărca fișierul cu date experimentale primite pentru a identifica un sistem de ordin II utilizând atât metode parametrice cât și neparametrice. Vom studia sistemul de ordin II cu poli complex-conjugați care nu are nici un zero.

A graph of a graph

Description automatically generated with medium confidence

Figura 1. Datele inițiale cu prima ieșire

**Codul MatLAB:**

t=double(Lascu(:,1));

u=double(Lascu(:,2));

y=double(Lascu(:,3));

plot(t,u,t,y);grid on;title('Semnalul de iesire suprapus pe semnalul de intrare');

Vom avea nevoie de indicii care reprezintǎ momentele de timp asociate semnalului de intrare, respectiv semnalului de ieşire la momentul in care defazajul este 90 de grade.

A graph with numbers and lines

Description automatically generated

Figura 2. Indicii

**Codul MatLAB:**

ymax=186;

umax=182;

ymin=194;

umin=190;

Pentru a identifica sistemul prin gasirea defasajului de 90 de grade intre intrare si iesire vom avea nevoie de uramtorii parametri:

- Factorul de proportionalitate K

- Factorul de amortizare ζ

- Pulsatia naturala a oscilatilor

- Defazajul

Pentru a calcula factorul de proporționalitate am folosit formula:

Pentru a calcula factorul de amortizare am folosit formula :

**𝜁**

unde reprezintǎ amplificarea in zona frecventelor medii

iar 𝑦𝑚𝑎𝑥 (186), 𝑦𝑚𝑖𝑛 (194), 𝑢𝑚𝑎𝑥 (182), 𝑢𝑚𝑖𝑛 (190) reprezintă datele exportate de pe grafic. Pentru a calcula pulsația naturală de oscilație am avut nevoie de formula:

[rad/sec]

unde se numește pulsația de rezonanță și are formula :

iar perioada de rezonanta Tn are formula:

**𝑇n = 2 ∗ (t(umin) − t(umax ) )=** **1.6000e-04 [s]**

Pentru a calcula defazajul am folosit formula:

Unde dt este intarzierea pe care semnalul de iesire o are fata de semnalul de intrare, in secunde, pentru care am folosit formula:

dt=t(ymax)-t(umax)

In urma acestor calcule, am verificat daca valorile gasite sunt corecte dupa conditia:

Având toți parametrii necesari, a rezultat funcția de transfer de gradul 2, de forma:

**A number and a line

Description automatically generated with medium confidence**

**Codul MatLAB:**

dt=t(ymax)-t(umax)

Tn=2\*(t(umin)-t(umax))

wn=2\*pi/Tn

ph=dt\*wn

m1=(y(ymax)-y(ymin))/(u(umax)-u(umin))/k

defazaj=rad2deg(ph)

tita=k/m1/2

H=tf(k\*wn^2,[1 2\*tita\*wn wn^2])

Pentru validarea şi simularea modelului din condiții inițiale nenule este necesar modelul de tip spațiul stǎrilor. Pe baza formei canonice observabile dedusǎ din funcția de transfer se obțin matricele:

**A=**

**B=**

**C=**

**D=**

**Codul MatLAB:**

A=[0,1;-wn^2,-2\*tita\*wn];

B=[0;k\*wn^2];

C=[1,0];

D=0;

Pentru a calcula ieșirea simulată am folosit funcția lsim.

ysim=lsim(A,B,C,D,u,t,[y(1), (y(2)-y(1))/(t(2)-t(1))]);

Pentru validarea sistemului vom calcula eroarea medie pătratica

J = 0.0713

Și vom calcula eroarea medie pătratică normalizată

Empn = 0.1423

A graph with a line of a sound wave

Description automatically generated with medium confidence

Figura 3. Suprapunerea dintre semnalul masurat si semnalul calculat

**Codul MatLAB:**

sys=ss(A,B,C,D);

ysim=lsim(sys,u,t,[y(1),(y(2)-y(1))/(t(2)-t(1))]);

figure;plot(t,[y,ysim]); grid on;title('Semnalul identificat prin metoda defasajului suprapus peste semnalul masurat')

J=norm(y-ysim)/sqrt(length(y))

Empn=norm(y-ysim)/norm(y-mean(y))

**2.Identificarea parametrica a sistemului fara zero**

Pornind de la structura generală de identificare care folosește minimizarea erorii de predicție, am impus 4 metode particulare: metoda celor mai mici pătrate recursive (implementată în rutina ARX), metoda celor mai mici pătrate extinsă (implementată în rutina ARMAX), metoda variabilelor instrumentale (implementată în rutina IV4 ), și metoda erorii de ieșire (implementată în rutina OE). În cadrul acestui proiect, rezultatele cele mai bune au fost obținute prin metodele ARMAX și IV

**2.1 Metoda celor mai mici pătrate extinsă (ARMAX)**

Schema bloc a acestui model este:

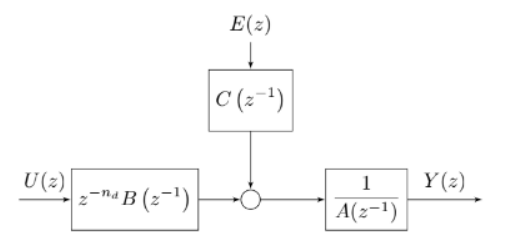


Figura 4. Structura corespunzătoare metodei ARMAX

Modelul discret de tip proces + perturbație este:

(𝑧 −1)(𝑧) = 𝑧 −𝑛𝑑𝐵(𝑧 −1)𝑈(𝑧) + 𝐶(𝑧 −1)𝐸(𝑧),

unde:

(𝑧 −1) = 1 + 𝑎1𝑧 −1 + 𝑎2𝑧 −2 + ⋯ + 𝑎𝑛𝐴 𝑧 −𝑛𝐴

(𝑧 −1) = 𝑏1 + 𝑏2𝑧 −1 + ⋯ + 𝑏𝑛𝐵 𝑧 −𝑛𝐵+1

(𝑧 −1) = 1 + 𝑐1𝑧 −1 + 𝑐2𝑧 −2 + ⋯ + 𝑐𝑛𝐶 𝑧 –𝑛C

Pentru identificarea modelului vom avea:

m\_armax=armax(data\_vd,[2,1,5,1])

unde primele 3 cifre reprezintă gradul polinoamelor A, B și C, iar ultima cifră este numărul tacților de întârziere.

Funcția de transfer obținută în discret este:

Hd\_armax= A close up of numbers

Description automatically generated

Apoi, folosind funcțiile RESID și COMPARE, putem observa că testul de validare este trecut.

A screenshot of a graph

Description automatically generated

Figura 5. Testul de autocorelatie respectiv gradul de suprapunere

**Codul MatLAB:**

Tr=(t(131)-t(120))/2;

data\_id = iddata(y,u,Tr);

m\_armax = armax(data\_id,[2,1,2,1])

%gradul de suprapunere

grid on;figure; compare(data\_id,m\_armax); shg;title('Gradul de suprapunere folosind metoda ARMAX')

% validarea statistica

grid on;figure; resid(data\_id,m\_armax,'corr',5);title('Validare prin autocorelatie')

Hd\_armax = tf(m\_armax.B,m\_armax.A,Tr,'variable','z^-1')

**2.2 Metoda erorii de iesire (OE)**

Schema bloc a acestui model este:

**A diagram of a mathematical equation

Description automatically generated**

Figura 6: Structura corespunzătoare metodei OE

Modelul discret de tip proces + perturbație este:

A black and white math equation

Description automatically generated

Unde:



Pentru identificarea modelului vom avea:

m\_oe = oe(data\_id,[1,2,1])

Unde primele 2 cifre reprezintă gradul polinoamelor B si F , iar ultima cifră este numărul tacților de întârziere. Funcția de transfer obținută în discret este:

Hd\_oe= A number and math symbols

Description automatically generated with medium confidence

Apoi, folosind funcțiile RESID și COMPARE, putem observa că testul de validare este trecut.

A screenshot of a graph

Description automatically generated

Figura 7. Testul de intercorelatie respectiv gradul de suprapunere

**Codul MatLAB:**

m\_oe=oe(data\_id,[1 2 1]);

grid on;figure; resid(m\_oe,data\_id);title('Validare prin intercorelatie')

grid on;figure; compare(m\_oe,data\_id);title('Gradul de suprapunere folosind metoda OE')

Hd\_oe=tf(m\_oe.B,m\_oe.F,Tr,'variable','z^-1')

**3. Identificarea parametrică a sistemului cu zero**

**A screen shot of a sound wave

Description automatically generated**

Figura 8. Graficul semnalului de iesire masurat suprapus pe semnalul de intrare masurat dupa introducerea unui zero in plus

**Codul MatLAB:**

t=Lascu(:,1);

u=Lascu(:,2);

y=Lascu(:,4);

plot(t,u,t,y);title('Semnalul de iesire suprapus pe semnalul de intrare dupa ce am introdus un zero in plus');grid on;

**3.1 Metoda celor mai mici pătrate extinsă (ARMAX)**

Schema bloc a acestui model este:

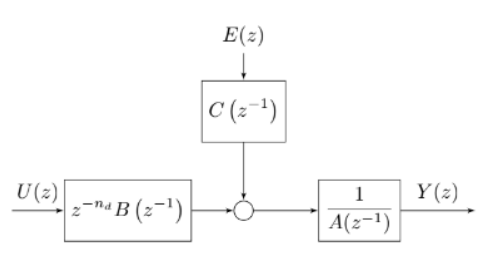


Figura 9. Structura corespunzătoare metodei ARMAX

Modelul discret de tip proces + perturbație este:

𝐴(𝑧 −1)𝑌(𝑧) = 𝑧 −𝑛𝑑𝐵(𝑧 −1)𝑈(𝑧) + 𝐶(𝑧 −1)𝐸(𝑧),

unde:

𝐴(𝑧 −1) = 1 + 𝑎1𝑧 −1 + 𝑎2𝑧 −2 + ⋯ + 𝑎𝑛𝐴 𝑧 −𝑛𝐴

𝐵(𝑧 −1) = 𝑏1 + 𝑏2𝑧 −1 + ⋯ + 𝑏𝑛𝐵 𝑧 −𝑛𝐵+1

𝐶(𝑧 −1) = 1 + 𝑐1𝑧 −1 + 𝑐2𝑧 −2 + ⋯ + 𝑐𝑛𝐶 𝑧 –𝑛C

Pentru identificarea modelului vom avea:

m\_armax=armax(data\_id,[2,2,4,1])

unde primele 3 cifre reprezintă gradul polinoamelor A, B și C, iar ultima cifră este numărul tacților de întârziere. Funcția de transfer obținută în discret este:

Hd\_armax= A number and a line of numbers

Description automatically generated with medium confidence

Apoi, folosind funcțiile RESID și COMPARE, putem observa că testul de validare este trecut.

A screenshot of a graph

Description automatically generated

Figura 10. Testul de autocorelatie respectiv gradul de suprapunere al semnalului folosind metoda ARMAX

**Codul MatLAB:**

data\_id = iddata(y,u,Tr);

m\_armax = armax(data\_id,[2,2,4,1])

%gradul de suprapunere

grid on;figure; compare(data\_id,m\_armax); shg;title('Gradul de suprapunere folosind metoda ARMAX')

% validarea statistica

grid on;figure; resid(data\_id,m\_armax,'corr',9);title('Validare prin autocorelatie folosind metoda ARMAX')

Hd\_armax = tf(m\_armax.B,m\_armax.A,Tr,'variable','z^-1')

**3.2 Metoda erorii de iesire (OE)**

Schema bloc a acestui model este:

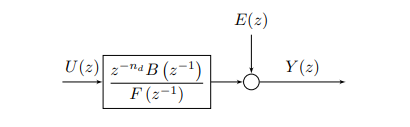
****

Figura 11. Structura corespunzătoare metodei OE

Modelul discret de tip proces + perturbație este:

A black and white math equation

Description automatically generated

Unde:

A math equation with numbers and symbols

Description automatically generated

Pentru identificarea modelului vom avea:

m\_oe=oe(data\_vd,[2,2,1])

unde primele 2 cifre reprezintă gradul polinoamelor F si B , iar ultima cifră este numărul tacților de întârziere. Funcția de transfer obținută în discret este:

Hd\_oe= A number and numbers with a line

Description automatically generated with medium confidence

Apoi, folosind funcțiile RESID și COMPARE, putem observa că testul de validare nu este trecut.

A screenshot of a graph

Description automatically generated

Figura 12. Testul de intercorelatie respective gradul de suprapunere folosind metoda OE

**Codul MatLAB:**

% identificare OE

m\_oe=oe(data\_id,[2 2 1]);

grid on;figure; resid(m\_oe,data\_id,9);title('Validare prin intercorelatie folosind metoda OE')

grid on;figure; compare(m\_oe,data\_id);title('Gradul de suprapunere folosind metoda OE')

Hd\_oe=tf(m\_oe.B,m\_oe.F,Tr,'variable','z^-1')

**3.3 Metoda variabilelor instrumentale (IV)**

Schema bloc a acestui model este:

**A diagram of a mathematical equation

Description automatically generated**

Figura 13. Structura corespunzătoare metodei IV

Modelul discret de tip proces + perturbație este:

A black and white text

Description automatically generated

Unde:

A math symbols with black and white text

Description automatically generated with medium confidence

Pentru identificarea modelului vom avea:

m\_iv = iv4(data\_id,[2,2,1])

Unde primele 2 cifre reprezintă gradul polinoamelor A si B , iar ultima cifră este numărul tacților de întârziere. Funcția de transfer obținută în discret este:

Hd\_iv= A math equation with numbers and lines

Description automatically generated with medium confidence

Apoi, folosind funcțiile RESID și COMPARE, putem observa că testul de validare nu este trecut.

A close-up of a graph

Description automatically generated

Figura 14. Testul de intercorelatie respectiv gradul de suprapunere folosind metoda IV

**Codul MatLAB:**

% identificare IV

m\_iv = iv4(data\_id,[2,2,1]);

grid on;figure; resid(m\_iv,data\_id);title('Validare prin intercorelatie folosind metoda IV')

grid on;figure; compare(m\_iv,data\_id);title('Gradul de suprapunere folosind metoda IV')

Hd\_iv=tf(m\_iv.B,m\_iv.A,Tr,'variable','z^-1')

**3.4 Filtrare folosind metodele pem si n4sid**

Dupa ce ambele teste de validare prin intercorelatie folosind atat metoda OE cat si IV nu au fost trecute am incercat sa aplic un filtru de tip FIR (Finite Impulse Response), adica am aproximat primii 2N parametrii Markov si am construit matricea Henkel folosindu-ma de functia n4sid din MatLab.

Dupa crearea unui model te tipul n4sid, mi-am cautat momentele de timp in care se termina zona frecventelor joase respectiv zona frecventelor medii, si am analizat semnalul doar in zona frecventelor medii.

Mi-am construit un nou set de date de validare care sa nu contina zona frecventelor joase respective a celor inalte, iar pe baza noului set de date si a modelului n4sid mi-am construit un nou model care sa minamlizeze eroarea de predictie folosind functia pem.

Dupa care modelul pe baza modelului de tip pem si a noului set de date am aplicat atat testul de intercorelatie cat si gradul de suprapunere.

A screen shot of a graph

Description automatically generated

Figura 15. Aproximarea gradului modelului folosind functia n4sid

Apoi, folosind funcțiile RESID și COMPARE, putem observa că testul de validare este trecut.

**A screenshot of a computer

Description automatically generated**

Figura 16. Testul de intercorelatie respectiv gradul de suprapunere

**Codul MatLAB:**

m\_n4sid=n4sid(data\_id,1:10)

idx=[30:330]

data2=iddata(y(idx,1),u(idx,1),Tr)

m\_pem2=pem(data2,m\_n4sid)

figure; compare(data2,m\_pem2);title('Gradul de suprapunere dupa ce am filtrat semnalul folosind metoda n4sid respectiv pem prezentata la curs');

figure; resid(data2,m\_pem2);title('Validare prin intercorelatie')

CONCLUZIE

În cadrul acestui proiect, am identificat un sistem bazat pe răspuns la sinus fara zero, folosind atât metode neparametrice, prin cautarea defazajului de 90 de grade, cât și metode parametrice, folosind metode precum metoda celor mai mici pătrate extinse (ARMAX) și metoda erorii de iesire (OE). De asemenea, am identificat si un sistem bazat pe raspuns la sinus cu zero, folosind metode parametrice, cele mai eficiente fiind metoda celor mai mici pătrate extinse (ARMAX) si prin aplicarea unui filtru de tip FIR.