Proiect la Identificarea Sistemelor

Identificarea unei axe actionate cu motor BLDC

Coordonator:
Prof.univ.dr.ing. Petru DOBRA

Student:
Albaş Ciprian
Grupa 30131

08 Ianuarie 2019

Cuprins

Identificarea unei axe actionate cu motor BLDC

1.1 Obtinerea datelor experimentale

1.1.1 Introducere

In Figura 1.1 este prezentat un CNC actionat cu motore BLDC:

Figura 1.1: CNC actionata cu motor BLDC

Sistemul mecanic de pozitionare si sistemul de actionare cu motor BLDC pentru o axa este prezentat in Figura 1.2.:

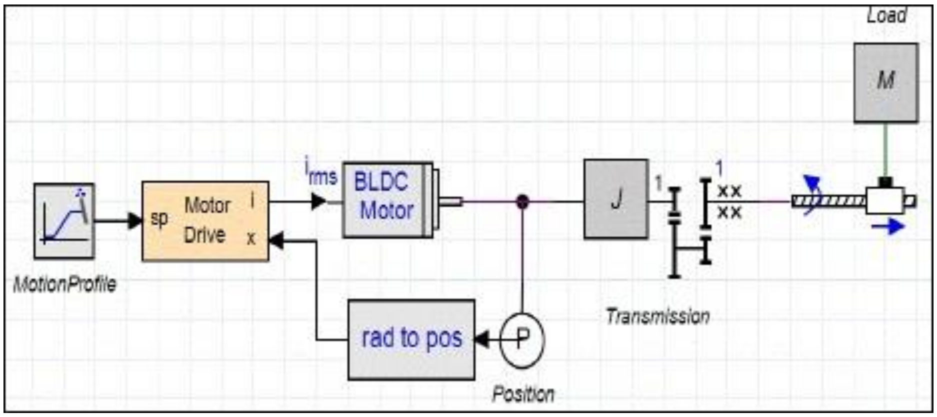


Figura 1.2: Modelul de actionare si pozitionare al unei axe

Motorul este comandat cu ajutorul unui driver de putere comandat in PWM. Viteza unghiulara si pozitia se masoara pe baza semnalelor provenite de la cei trei senzori Hall montati in statorul motorului. Rotorul motorului BLDC are cinci de poli magnetici, iar caracterisctICile electro-mecanice ale motorului sunt prezentate în Figura 1.3.

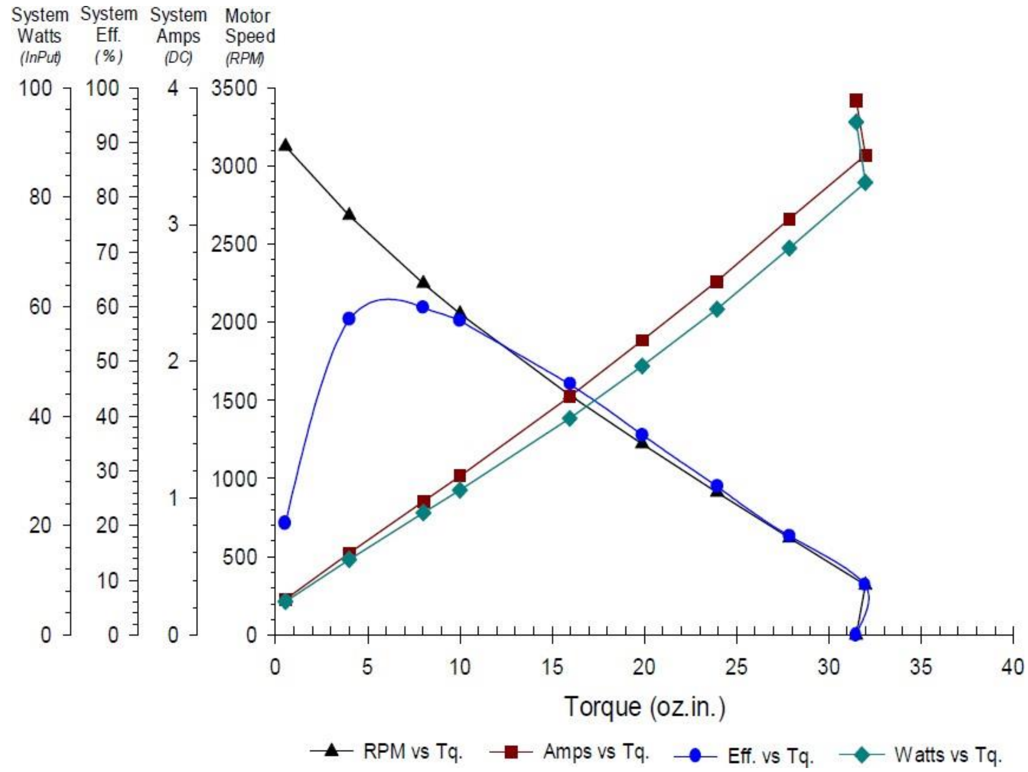


Figure 1.3: Caracteriscticile electro-mecanice ale motorului BLDC

Aparatura utilizata: sursa de alimentare, multimetru, driver de putere, osciloscop, sistem numeric comanda si achizitie a datelor.

1.2 Achizitia datelor de intrare-iesire

Utilizând un sistem numeric de comanda se genereaza semnalele de comanda pentru motorul BLDC (SPAP+SP)¸ si se achizitioneaza datele intrare-iesire în vederea procesarii ulterioare (comanda (factor de umplere), curent(i), viteza unghiulara(ω) si pozitia unghiulara(θ).

1.2.1 Desfasurarea experimentului

1. Se alimenteaza ansamblul driver + motor BLDC cu = 24 V .
2. Se efectueaza urmatorul experiment:

A.1 Se genereaza un semnal de tip SPAB având caracteristicile corelate cu dinamica ansablului „motor BLDC + axa“;

A.2 Se vizualizeaza ¸si se masoara sincron intrarea, si iesirile, obtinând datele experimentale: [t_k,u_k, ω_k, Θ_k] k=1, 2, ...

1.3 Procesarea datelor experimentale

Vizualizarea datelor experimentale utilizând mediul Matlab.

Se vor determina functiile de transfer ale ansablului „motor BLDC + axa“ utilizând metodele de identificare parametrica (MCMMPR, MCMMPE, VI, MEP, etc.).

1.3.1 Validarea modelului

Validarea modelului determinat se face pe baza erorii de predictie reziduale¸ si pe baza decorelarii dintre observatii¸ si eroarea de predictie.

De asemenea se va compara raspunsul experimental cu raspunsul modelului la intrarea cu care a fost obtinut raspunsul experimental. Se calculeaza eroarea medie patratica normalizata (ε_{MPN}) ¸si eroarea de urmarire (FIT):

$$\epsilon_{MPN} = \frac{\|\mathbf{y}-\mathbf{y}^M\|}{\|\mathbf{y}-\bar{\mathbf{y}}\|} \times 100,$$
$$FIT = (1-\epsilon_{MPN}) \times 100$$

unde y este vectorul masuratorilor, y^M raspunsul modelului ¸si $\bar{y}$ este valoarea medie a vectorului masuratorilor.

Achizitia datelor experimentale

Pentru realizarea acestui proiect am importat datele din matricea “albas.mat” in Matlab apoi am achizitionat datele din aceasta matrice. Am atribuit variabilelor t,u,w,y valorile din tabelul importat si am afisat intrarea u(factorul de umplere PWM), viteza unghiulara w(rad/sec) si pozitia unghiulara y(impulsuri). (Figura 1)

Observam ca iesirea Y e un vector linie cu 3 elemente: intrarea (semnal pseudo aleator binar - SPAB), viteza unghiulara si pozitia unghiulara pentru axa pe care se deplaseaza motorul.

Am pus timpul intr-o variabila “t”: t=double(d.X.Data').

Urmatorul pas il reprezinta indentificarea intrarii, pozitiei si a vitezei, si introducerea fiecareia in variabile:

```
u=double (d.Y(1,3).Data'); %intrare
w=double (d.Y(1,2).Data'); %viteza
y=double (d.Y(1,1).Data'); %pozitie
```

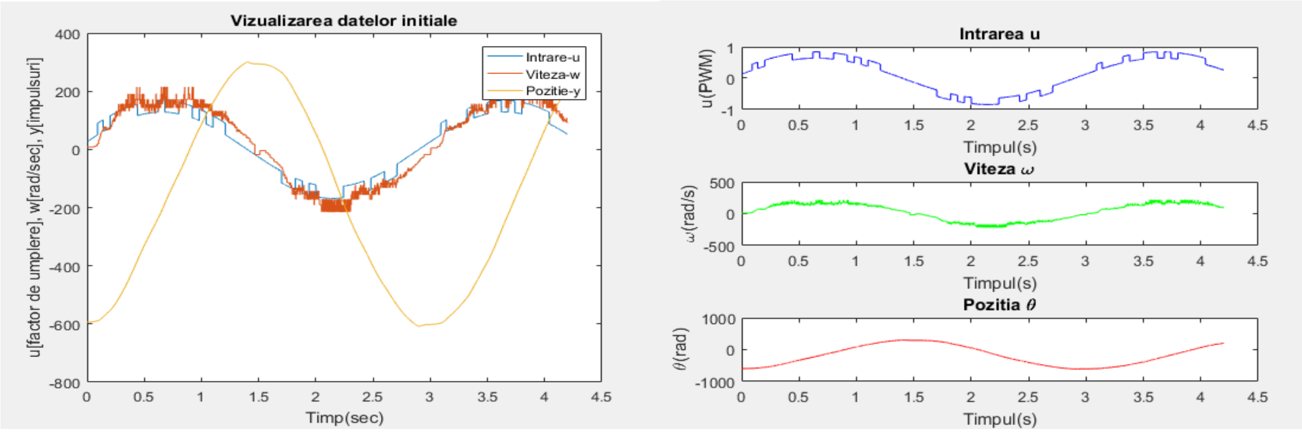


Figura 1

Urmatorul pas este de a identifica si a valida modelul. Pentru aceasta trebuie sa luam doua intervale, primul interval fiind pentru identificarea modelului, iar al doilea pentru validarea lui.

Pe semnalul w alegem 4 puncte pentru a imparti graficul in doua pentru a elimina datele de la oprire. (Figura 2)

Aceste puncte le-am exportat in Workspace cu numele de “valori”, iar apoi am apelat comanda “valori.DataIndex” pentru a le afisa:

x1=314; x2=3606; x3=4004; x4=7343;

Pe primul sens de mers vom face identificarea, iar pe al doilea sens validarea. (Figura 3)

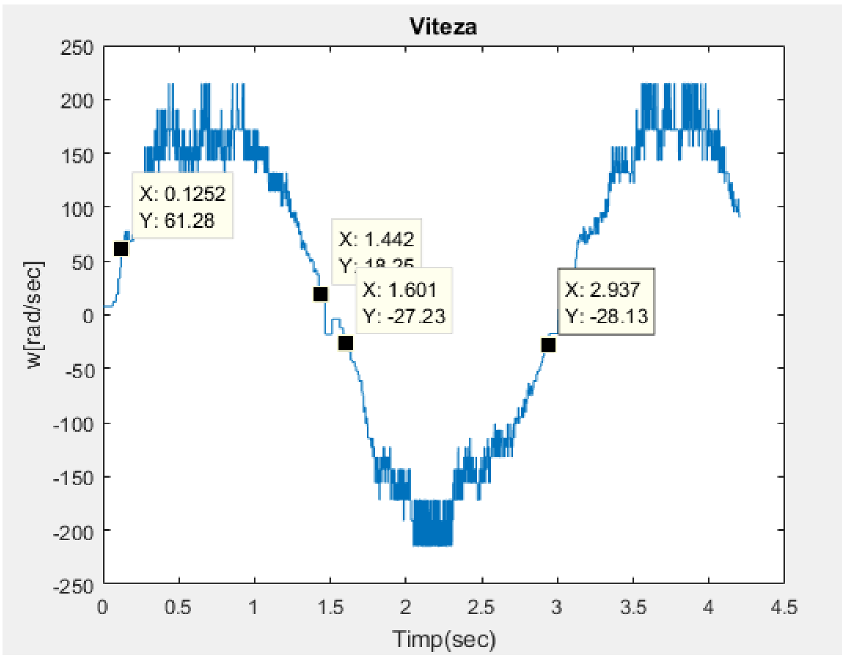


Figura 2

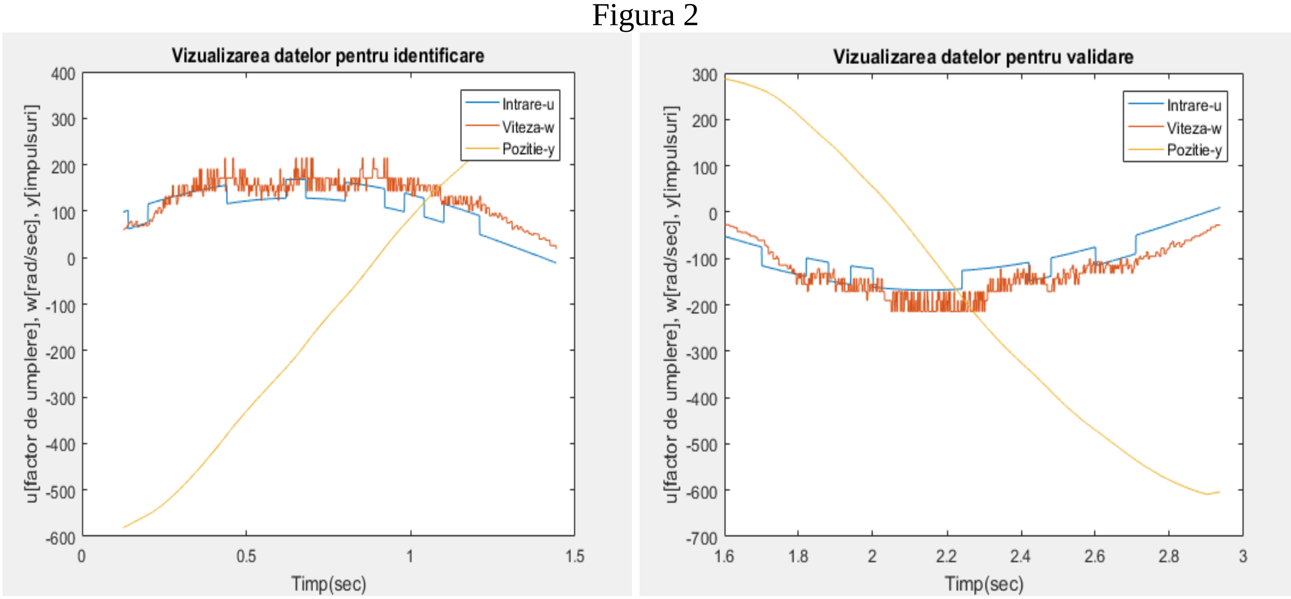


Figura 3

Calculam perioada de esantionare `Te=t(2)-t(1),` astfel `Te = 4.0000e-04` adica 0.4 milisecunde

Identificarea si validarea pentru viteza unghiulara

Datele pentru identificare sunt:d_id_w=iddata(w(i1),u(i1),Te)
Datele pentru validare sunt: d_vd_w=iddata(w(i2),u(i2),Te)

Apelam la metodele de identificare arx, armax, oe si iv4. Aleg sa folosesc metodele arx si iv4.

Modelul ARX (Metoda celor mai mici patrate recursiva)

Structura: $A(q-1)y(t)=q^{-d}B(q-1)u(t)+e(t)$ oricare ar fi t natural si e(t) fiind eroarea.
Pentru identificarea modelului vom avea:

Model=arx(data,[nA nB nk])

unde n_A este ordinul polinomului A , n_B ordinul polinomului B ,n_k este timpul mort.

- prima data cream modelul: `m_w_arx=arx(d_id_w,[1 1 1]);`
- validam modelul: `resid(m_w_arx,d_vd_w,15);` (Figura 4)
- comparam iesirea modelului cu iesirea masurata: `compare(d_vd_w,m_w_arx);` (Figura 5)

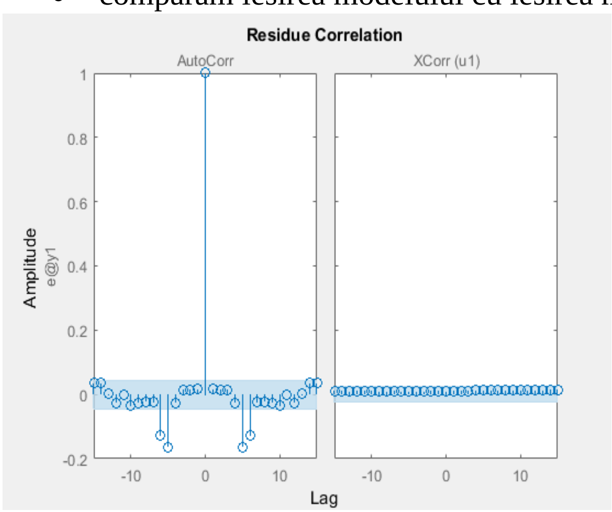


Figura 4

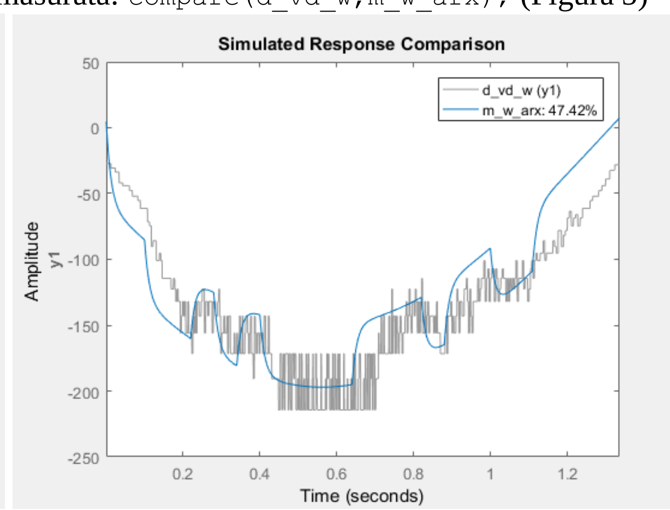


Figura 5

Din Figura 4 primul grafic reprezinta functia de autocorelatie, iar al doilea grafic reprezinta functia de intercorelatie. Banda albastra este nivelul in care trebuie sa se incadreze valorile pentru a fi validat modelul. In acest caz se functia de autocorelatie este trecuta.

Aflam functia de transfer a modelului, functia este redata in discret si trebuie sa o transformam in continuu:

Hzw1=tf(m_w_arx.B,m_w_arx.A,Te,'variable','z^-1')
Hsw1=d2c(Hzw1,'zoh')

```
              7.473 z^-1                        18990
Hzw1=  -----------------           Hsw1 =   ------------
            1 - 0.9681 z^-1                    s + 81.02
```

Modelul IV4 (Metoda variabilelor instrumentale)

Structura: $A(q{-1})y(t)=q^{-d}B(q^{-1})u(t)+w'(t)$ oricare ar fi t natural unde: $w'(t)=A(q^{-1})\,w(t)$.
Aici, $w'(t)$ este o perturbatie oarecare, independenta de $u(t)$, de medie nula si varianta finita

Model=iv4(data,[nA nB nk])

unde n_A este ordinul polinomului A , n_B ordinul polinomului B ,n_k este timpul mort.

- cream modelul: m_w_iv4=iv4(d_id_w,[1,1,1]);
- validam modelul: resid(m_w_iv4,d_vd_w) (Figura 6)
- comparam iesirea modelului cu iesirea masurata: compare(d_vd_w,m_w_iv4) (Figura 7)

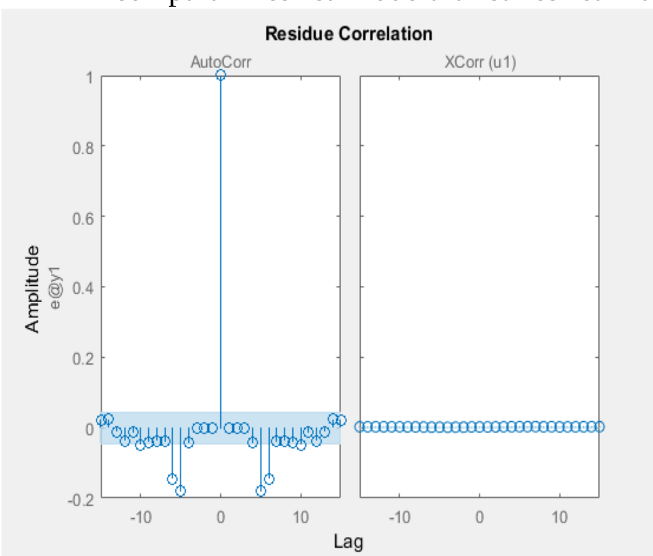


Figura 6

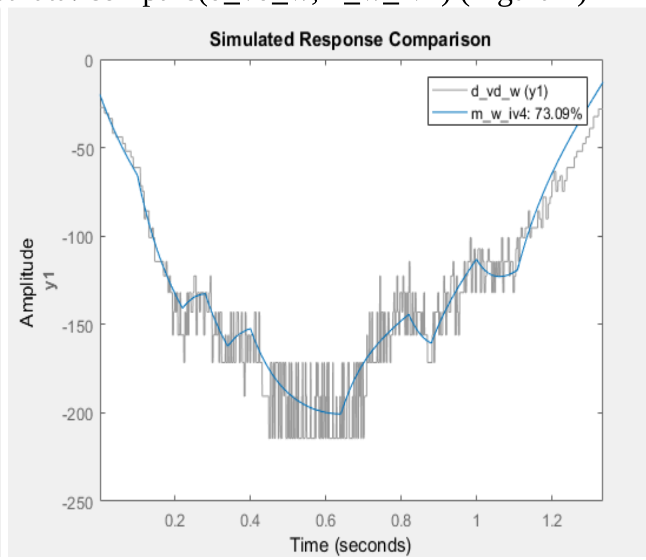


Figura 7

Din Figura 6 primul grafic reprezinta functia de autocorelatie, iar al doilea grafic reprezinta functia de intercorelatie. Banda albastra este nivelul in care trebuie sa se incadreze valorile pentru a fi validat modelul. In acest caz se functia de intercorelatie este trecuta.

Aflam functia de transfer a modelului, functia este redata in discret si trebuie sa o transformam in continuu:

```
Hzw2=tf(m_w_iv4.B,m_w_iv4.A,Te,'variable','z^-1')
Hsw2=d2c(Hzw2,'zoh')
```

Functiile de transfer in discret si continuu:

```
            1.348 z^-1                         3380
Hzw2=    ----------------          Hsw2=    -----------
          1 - 0.9944 z^-1                    s + 13.96
```

Identificarea si validarea pentru pozitia unghiulara

Datele pentru identificare sunt: `d_id_y=iddata(y(i1),w(i1),Te);`
Datele pentru validare sunt: `d_vd_y=iddata(y(i2),w(i2),Te);`

Apelam la metodele de identificare arx, armax, oe si iv4. Aleg sa folosesc metodele arx si iv4.

Modelul ARX (Metoda celor mai mici patrate recursiva)

Structura: $A(q^{-1})y(t)=q^{-d}B(q^{-1})u(t)+e(t)$ oricare ar fi t natural si e(t) fiind eroarea.
Pentru identificarea modelului vom avea:

Model=arx(data,[nA nB nk])

unde n_A este ordinul polinomului A , n_B ordinul polinomului B ,n_k este timpul mort.

- cream modelul: m_y_arx=arx(d_id_y,[1,1,0])
- validam modelul: resid(m_y_arx,d_vd_y); (Figura 8)
- comparam iesirea modelului cu iesirea masurata: compare(d_vd_y,m_y_arx) (Figura 9)

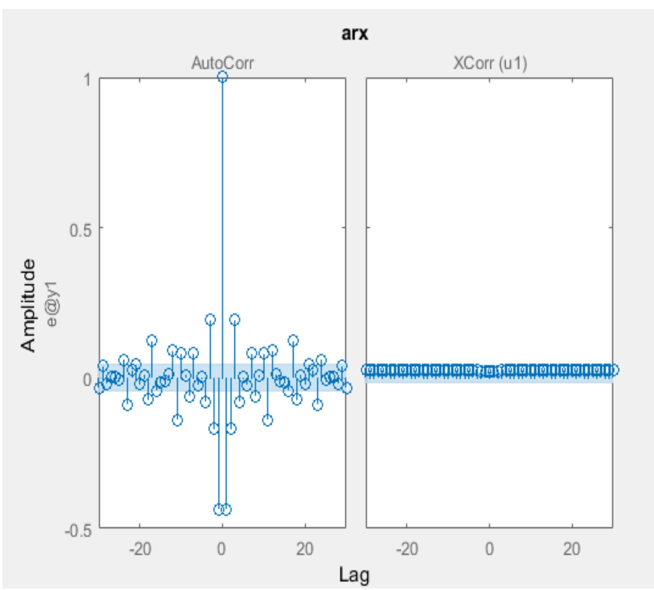


Figura 8

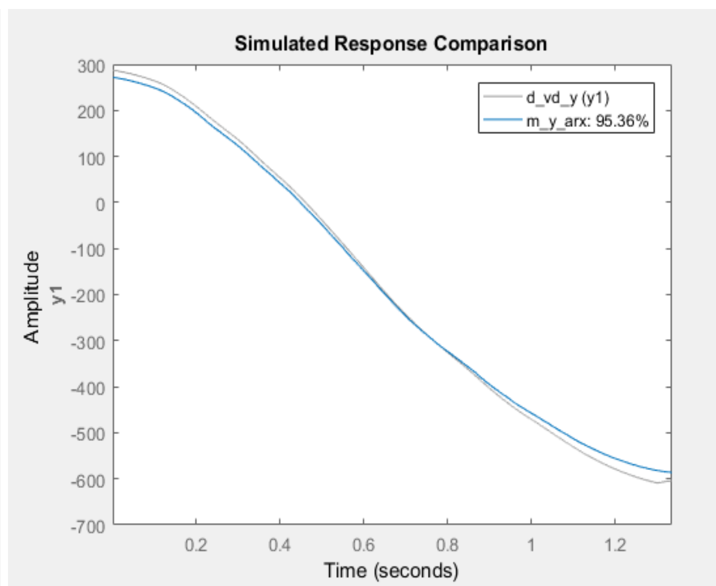


Figura 9

Observam ca functia de autocorelatie este trecuta.
Functiile de transfer in discret si continuu:

$$Hzy1 = \frac{0.001995}{1 - z^{-1}} \qquad Hsy1 = \frac{0.001995s + 4.988}{s + 0.06836}$$

In acestcaz 0.001995 si 0.06836 apar din cauza erorilor de calcul si le vom considera zero, astfel functia va fi :

$$Hsy1 = \frac{4.988}{s}$$ astfel obtinem un integrator.

Modelul IV4 (Metoda variabilelor instrumentale)

Structura: $A(q{-1})y(t)=q^{-d}B(q^{-1})u(t)+ w'(t)$ oricare ar fi t natural unde: $w'(t)= A(q^{-1})\,w(t)$.
Aici, $w'(t)$ este o perturbatie oarecare, independenta de $u(t)$, de medie nula si varianta finita

Model=iv4(data,[nA nB nk])

unde n_A este ordinul polinomului A , n_B ordinul polinomului B ,n_k este timpul mort.

- cream modelul: m_y_iv4=iv4(d_id_y,[1,1,0])
- validam modelul: resid(m_y_iv4,d_vd_y); (Figura 10)
- comparam iesirea modelului cu iesirea masurata: compare(d_vd_y,m_y_iv4) (Figura 11)

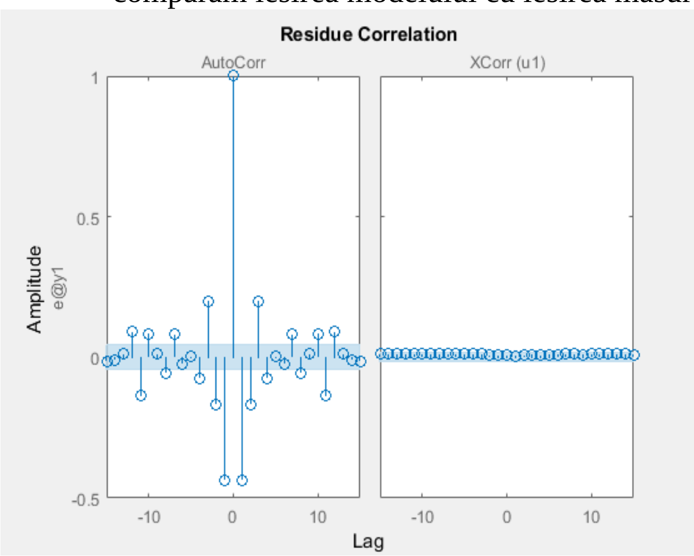


Figura 10

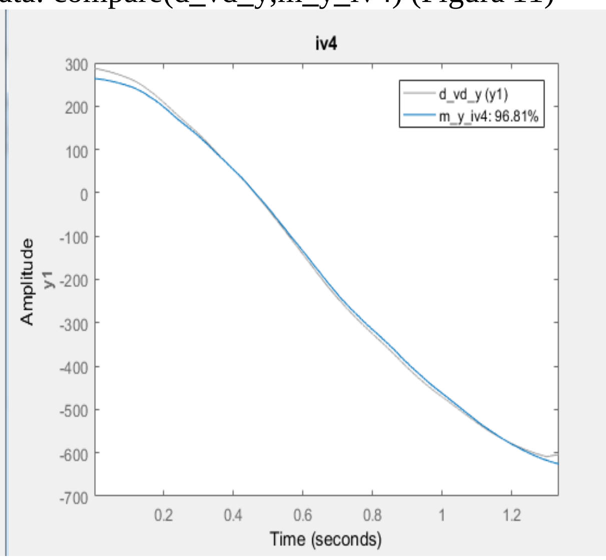


Figura 11

Observam ca functia de intercorelatie este trecuta
Functiile de transfer in discret si continuu:

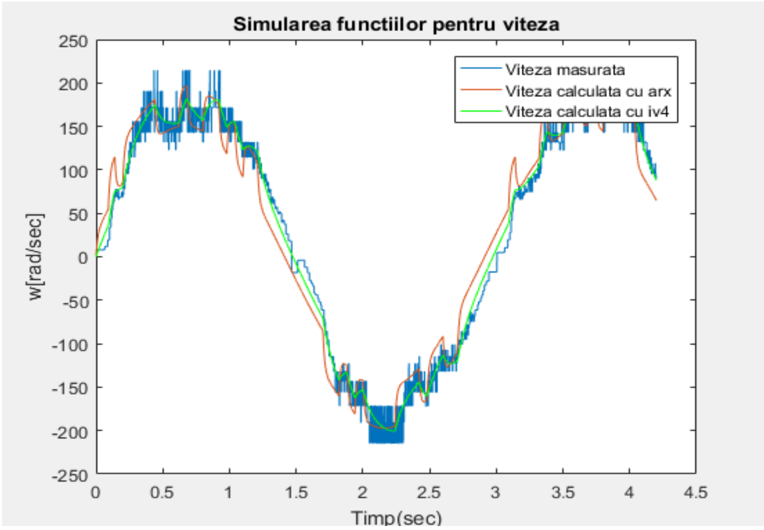
$$Hzy2= \frac{0.001955}{1 - z^{-1}} \qquad Hsy2 = \frac{4.888}{s - 0.1319}$$

In acestcaz - 0.1319 apare din cauza erorilor de calcul si le vom considera zero, astfel functia va fi : astfel obtinem un integrator.

$$Hsy2= \frac{4.888}{s}$$

Simularea modelelor

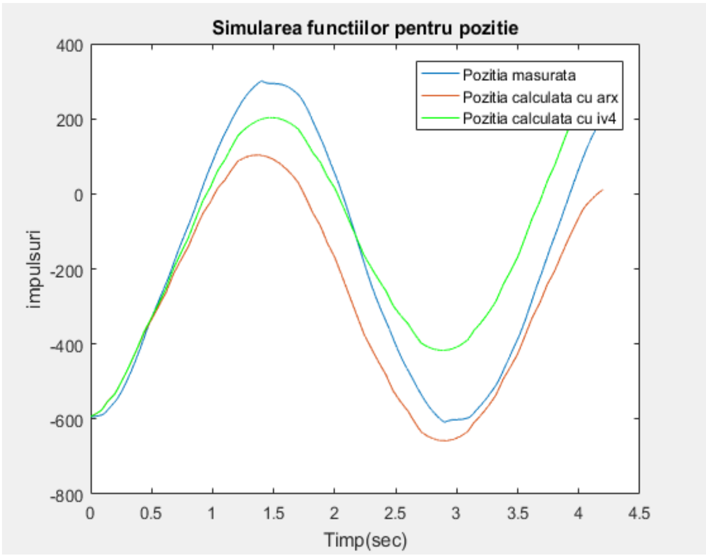
Functiile de transfer pentru viteza masurata si caculata cu metodele arx si iv4 sunt afisate in graficul din Figura 12.



Cod: Figura 12

```
wc_arx=lsim(Hsw1,u,t);
wc_iv=lsim(Hsw2,u,t);
plot(t,w,t,wc_arx,t,wc_iv,'g');
```

Functiile de transfer pentru pozitia masurata si caculata cu metodele arx si iv4 sunt afisate in graficul din Figura 13



Cod Figura 13

```
yc_arx=lsim(Hsy1,u*200,t);
yc_iv=lsim(Hsy2,u*200,t);
plot(t,y,t,yc_arx+y(1),t,yc_iv+y(1),'g');
```

Modelul in spatiul starilor : Eroarea medie patratica respectiv fit

Calcularea vitezei si pozitiei utilizand spatial starilor cu comanda n4sid, rezultand graficul din Figura 14.

```
data= iddata([w, y], u, Te);
sys=n4sid(data, 'best')
figure
resid(sys,data);
compare(data, sys);
```

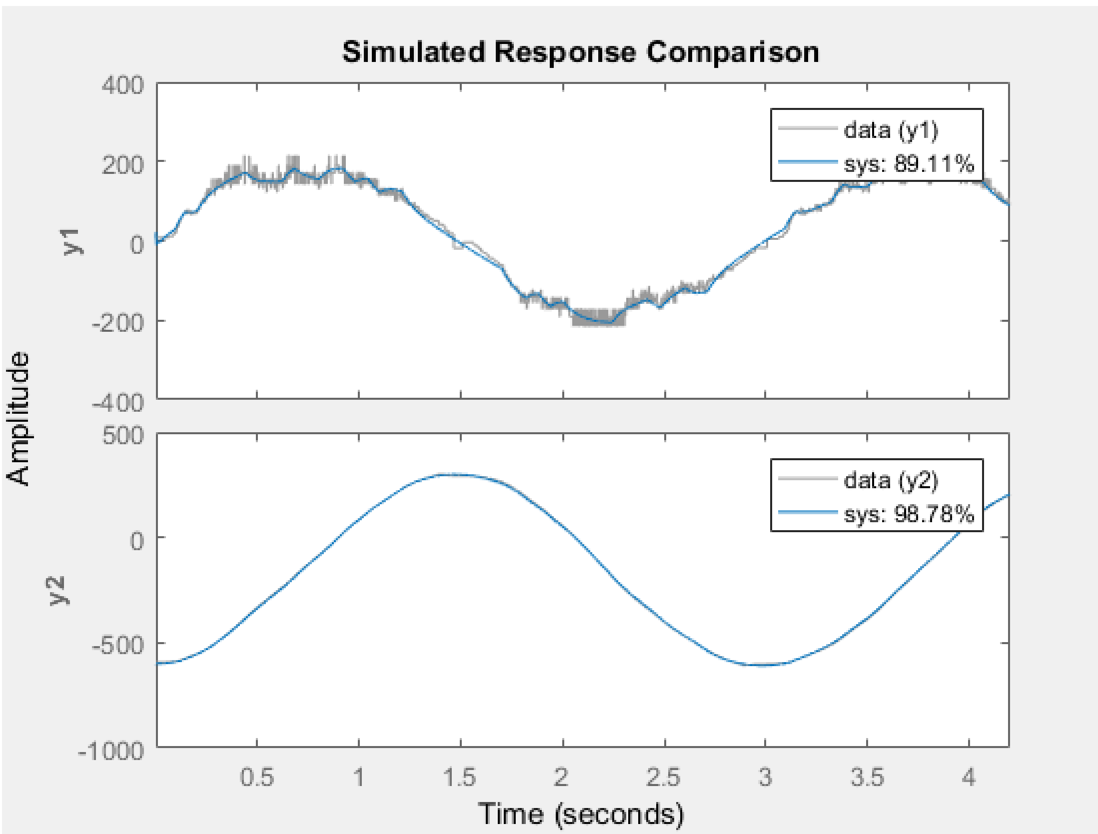


Figura 14

In cele invatate la curs se stie ca Empn a vitezei respectiv a pozitiei este complementul fit-ului obtinut de spatiul starilor astfel ne rezulta:

Fit_viteza:89.11%
Empn_viteza=100-89.11
=10. 89[%]
=0.1

Fit_viteza:98.79%
Empn_pozitie=100-98.78
=1.22[%]
=0.01

Cod Matlab

```
%importare date
date=albas;
t=double(date.X.Data')
u=double(date.Y(1,3).Data')
w=double(date.Y(1,2).Data')
y=double(date.Y(1,1).Data')

%vizualizarea datelor initiale
plot(t,[u*200,w,y]); %intrarea este scalata cu 200
title('Vizualizarea datelor initiale')
xlabel('Timp(sec)')
ylabel('u[factor de umplere], w[rad/sec], y[impulsuri]')
legend('Intrare-u','Viteza-w','Pozitie-y')
figure

%vizualizarea intrarii,vitezei si pozitiei in acelas grafic
subplot(311)
plot(t,u,'b')
title('Intrarea u')
ylabel('u(PWM)');
xlabel('Timpul(s)');

subplot(312)
plot(t,w,'g')
title('Viteza \omega')
ylabel('\omega(rad/s)');
xlabel('Timpul(s)');


subplot(313)
plot(t,y,'r')
title('Pozitia \theta')
ylabel('\theta(rad)');
xlabel('Timpul(s)');
figure

%graficul vitezei de pe care vom alege cele 4 puncte de care avem nevoie
plot(t,w);
title('Viteza');
xlabel('Timp(sec)');
ylabel('w[rad/sec]');

%cele 4 puncte + graficele pentru identificare si validare
x1=314;
x2=3606;
x3=4004;
x4=7343;
i1=[x1:x2]'; %pt identificare
i2=[x3:x4]'; %pt validare
```

```
%Vizualizarea datelor pentru identificare
plot(t(i1),[u(i1)*200,w(i1),y(i1)]);
title('Vizualizarea datelor pentru identificare')
xlabel('Timp(sec)')
ylabel('u[factor de umplere], w[rad/sec], y[impulsuri]')
legend('Intrare-u','Viteza-w','Pozitie-y')
figure
%Vizualizarea datelor pentru validare
plot(t(i2),[u(i2)*200,w(i2),y(i2)]);
title('Vizualizarea datelor pentru validare')
xlabel('Timp(sec)')
ylabel('u[factor de umplere], w[rad/sec], y[impulsuri]')
legend('Intrare-u','Viteza-w','Pozitie-y')
figure

%perioada de esantionare
Te=t(2)-t(1);% 4.0000e-04 perioada de esantionare 0.4 milisecunde

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Identificarea si validarea pentru viteza unghiulara(w) aleg ARX si IV4

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

   % datele pentru identificare si validare
   d_id_w=iddata(w(i1),u(i1),Te);  % date identificare viteza
   d_vd_w=iddata(w(i2),u(i2),Te);  % date validare viteza

%%%   ARX  %%%

m_w_arx=arx(d_id_w,[1 1 1]); % model viteza=date identificare 1-grB 1-grA 1-
timp mort
resid(m_w_arx,d_vd_w,15);% model viteza , validare , nr de puncte;
rezid -> autocorelatie si intercorelatie
figure
compare(d_vd_w,m_w_arx); % compar datele de validare cu modelul %% 47.42% %%
figure
%%% functia de transfer in z si s (continuu si discret)
Hzw1=tf(m_w_arx.B,m_w_arx.A,Te,'variable','z^-1') ;
Hsw1=d2c(Hzw1,'zoh');

%%%   IV4  %%%

m_w_iv4=iv4(d_id_w,[1 1 1]); % model viteza=date identificare 1-grB 1-grA 1-
timp mort
resid(m_w_iv4,d_vd_w,15); model viteza , validare , nr de puncte;
rezid -> autocorelatie si intercorelatie
figure
compare(d_vd_w,m_w_iv4); compar datele de validare cu modelul %% 73.09% %%
figure
%%% functia de transfer in z si s (continuu si discret)
Hzw2=tf(m_w_iv4.B,m_w_iv4.A,Te,'variable','z^-1') ;
Hsw2=d2c(Hzw2,'zoh');
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Identificarea si validarea pentru pozitia unghiulara(y) aleg ARX si IV4

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%datele pentru identificare si validare
   d_id_y=iddata(y(i1),w(i1),Te);  % date identificare viteza
   d_vd_y=iddata(y(i2),w(i2),Te);  % date validare viteza

  %%%  ARX  %%%

m_y_arx=arx(d_id_y,[1 1 0]); % model pozitie=date identificare 1-grB 1-grA 1-timp mort
resid(m_y_arx,d_vd_y,'corr',30);% model pozitie , validare , nr de puncte; rezid -> autocorelatie si intercorelatie
figure
compare(d_vd_y,m_y_arx); % compar datele de validare cu modelul %% 95.36% %%
figure
 %%% functia de transfer in z si s (continuu si discret)
Hzy1=tf(m_y_arx.B,m_y_arx.A,Te,'variable','z^-1')
Hsy1=d2c(Hzy1,'zoh')

%%%  IV4  %%%

m_y_iv4=iv4(d_id_y,[1 1 1]); model pozitie=date identificare 1-grB 1-grA 1-timp mort
resid(m_y_iv4,d_vd_y,15); );% model pozitie , validare , nr de puncte; rezid -> autocorelatie si intercorelatie
figure
compare(d_vd_y,m_y_iv4); compar datele de validare cu modelul %% 95.36% %%
figure
title('iv4');
figure
 %%% functia de transfer in z si s (continuu si discret)
Hzy2=tf(m_y_iv4.B,m_y_iv4.A,Te,'variable','z^-1')
Hsy2=d2c(Hzy2,'zoh')


%simulare w
wc_arx=lsim(Hsw1,u,t); % w calculat cu arx
wc_iv4=lsim(Hsw2,u,t); % w calculat cu iv4
plot(t,w,t,wc_arx,t,wc_iv4,'g');
title('Simularea functiilor pentru viteza'); xlabel('Timp(sec)')
ylabel('w[rad/sec]')
legend('Viteza masurata','Viteza calculata cu arx','Viteza calculata cu iv4');
```

```
%simulare y
yc_arx=lsim(Hsy1,u*200,t); % y calculat cu arx
yc_iv4=lsim(Hsy2,u*200,t); % y calculat cu iv4
plot(t,y,t,yc_arx+y(1),t,yc_iv4+y(1),'g');
title('Simularea functiilor pentru pozitie'); xlabel('Timp(sec)')
ylabel('impulsuri')
legend('Pozitia masurata','Pozitia calculata cu arx','Pozitia calculata cu
iv4');

%Calcularea vitezei si pozitiei utilizand spatiul starilor
data= iddata([w, y], u, Te);
sys=n4sid(data, 'best')
figure
resid(sys,data);
compare(data, sys);
```