Identyfikacja procesów przemysłowych		
Dominik Wróbel	27 XI 2018 / 04 XII 2018	Wt. 13:45, s. 111
Laboratorium 7		

1 Przebieg ćwiczenia

Laboratorium polega na identyfikacji parametrów modelu serwomechanizmu z silnikiem prądu stałego. Rozważany w zadaniu model jest uproszczonym modelem, który otrzymywany jest z równań mechanicznego i elektrycznego po pominięciu indukcyjności.

1.1. Zadanie 1

Celem zadania jest wyznaczenia parametrów H oraz K, których znaczenie wynika z poniższych wzorów uzyskanych z równania wyjściowego. Wyjściowe uproszczone równanie serwomechanizmu z silnikiem prądu stałego wyraża się wzorem

$$\frac{d\omega}{dt} = \frac{k_m}{JR}(u - k_e\omega - \frac{R}{k_m}f_t(\omega))$$

$k_m -$ stała mechanicza

$J -$ moment bezwładności

$R -$ rezystancja silnika

$k_e -$ stała elektryczna

$f_t -$ funkcja opisująca wypadkowy moment sił oporu

Wielkością mierzoną w laboratorium której pomiary użyte są w obliczeniach jest kąt obrotu wału silnika.

Wprowadzając oznaczenia:

$$x_1 = \varphi - \text{kąt obrotu w radianach}$$
$$x_2 = \omega$$

model przekształcony może zostać do postaci:

$$\dot{x}_1 = x_2,\ x_1(0) = x_{10}$$
$$\dot{x}_2 = K(u - H(x_2)), x_2(0) = x_{20}$$
$$H(\omega) = k_e\omega - \frac{R}{k_m}$$
$$K = \frac{k_m}{JR}$$

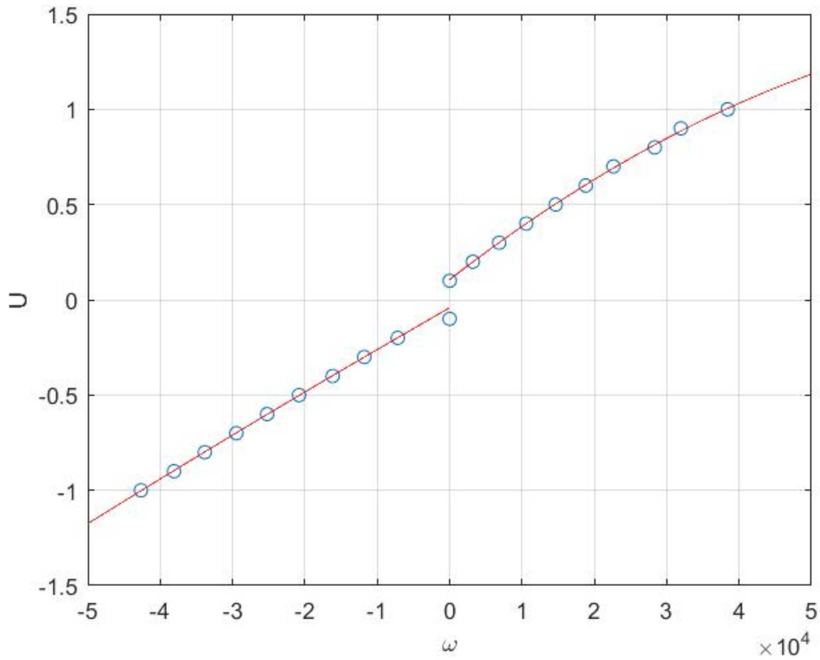
Zadanie rozpoczęto od wyznaczenia wartości funkcji H metodą najmniejszych kwadratów. W pierwszej kolejności na podstawie pomiarów wyznaczono zależność obrotów silnika od sterowania w stanie ustalonym. Na jej podstawie przy pomocy metody najmniejszych kwadratów (*polyfit*) wyznaczono współczynniki wielomianu drugiego stopnia który przybliża funkcję H:

$$H(\omega) = \begin{cases} a^+\omega^2 + b^+\omega + c^+, & \omega > 10^{-2} rad/s \\ a^-\omega^2 + b^-\omega + c^-, & \omega < 10^{-2} rad/s \\ 0, & pozatym \end{cases}$$

Otrzymane wartości współczynników to:

$$a^+ = -1,56e - 10\ b^+ = 2,93e - 05\ c^+ = 0,10$$
$$a^- = -1,86\ b^- = 2,17\ c^- = -0,042$$

Wykres funkcji wraz z naniesionymi wartościami pomiarów przedstawia Rysunek 1.

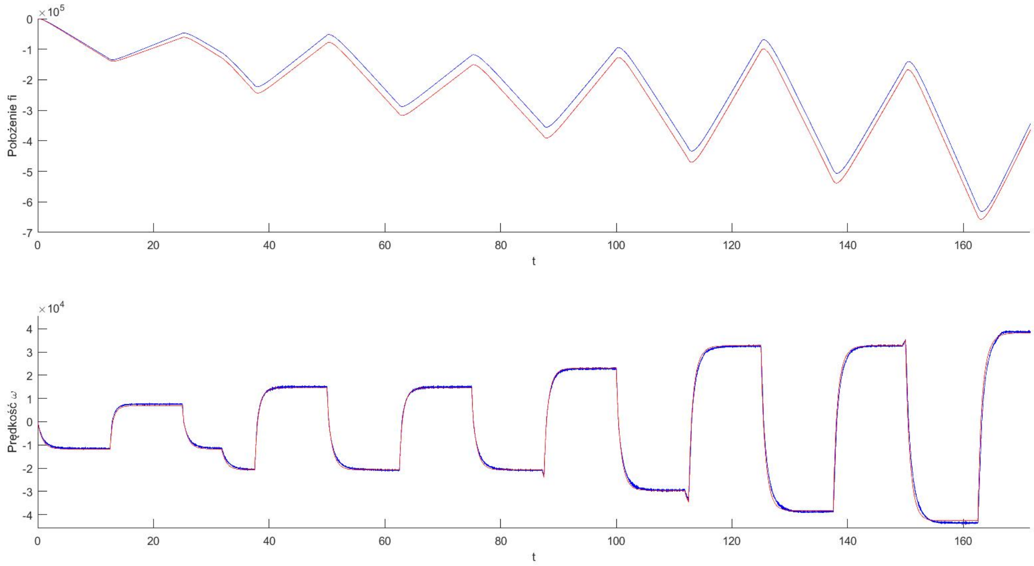


Rysunek 1: Funkcja $H(\omega)$ wraz z naniesionymi punktami pomiarowymi

Kolejnym krokiem w rozwiązaniu było wyznaczenie parametru K. Parametr ten wyznaczono minimalizując wskaźnik jakości:

$$Q(K, x_{10}, x_{20}) = \frac{1}{2N} \sum_{k=1}^{N} (x_1(kT_0) - y(kT_0))^2$$

Minimalizację przeprowadzono korzystając z funkcji matlaba *lsqnonlin* oraz całkowania metodą stałokrokową Rungego-Kutty. Porównanie wartości prędkości i położenia dla danych z modelu i pomiarów przedstawia Rysunek 2. Uzyskana wartość parametru K = 5.45e+04.

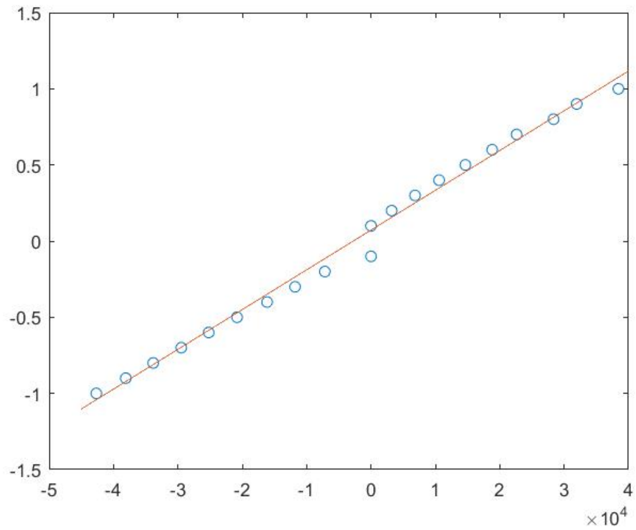


Rysunek 2: Porównanie modelu oraz pomiarów dla prędkości i położenia. Kolor czerwony - model.

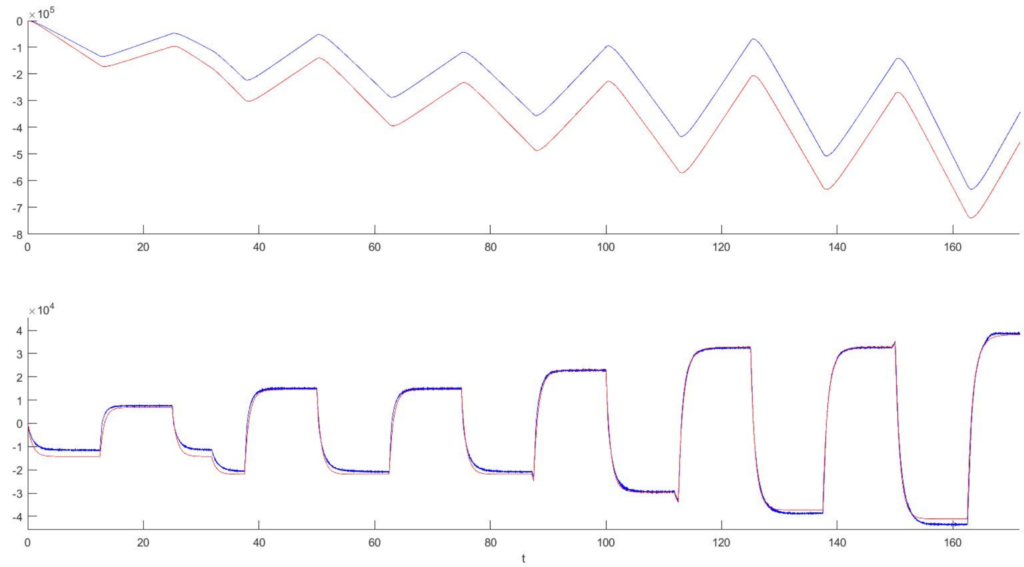
W celu porównania, wszystkie wykonane czynności powtórzono dla modelu liniowego:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -\frac{1}{T}x_2 + \frac{k}{T}u$$

Otrzymano wartości parametrów T = 2.91e-04 oraz k = 13.85.



Rysunek 3: Funkcja $H(\omega)$ wraz z naniesionymi punktami pomiarowymi - model liniowy



Rysunek 4: Porównanie modelu liniowego oraz pomiarów dla prędkości i położenia. Kolor czerwony - model.

1.1.1. Kod programu

Kod programu prezentuje listing poniżej.

```
close all;
clear all;

load('data_01.mat')

timeInterval = 0.01;

for n=1:20

    % reading data
    un = data(n).u;
    yn = data(n).y;

    % Y Axis - U %
    % last 20 samples
    un_20 = un(end-19:end);
    % mean of last 20 samples
    un_20Mean = mean(un_20);
    % add mean of last 20 samples to vector
    uLast20(n) = un_20Mean;

    % X Axis - Velocity %

    % diff of samples
    yn_diff = diff(yn);
```

```
    % calculating velocity
    %yn_velocity = yn_diff / timeInterval;
    yn_velocity = yn_diff./timeInterval;
    % last 20 samples
    yn_velocity = yn_velocity(end-20:end);

    % mean velocity
    meanVelocity(n) = mean(yn_velocity);

end

figure();
plot( meanVelocity, uLast20, 'o');


% Calculating a+, b+, c+ coef
pPlus = polyfit(meanVelocity(12:20),uLast20(12:20),2)
% Calculating a-, b-, c- coef
pMinus = polyfit(meanVelocity(2:10),uLast20(2:10),2)

xPlus = 0:0.1:50000;
xMinus = -50000:0.1:0;

yPlusResult = pPlus(1)*xPlus.^2 + pPlus(2)*xPlus + pPlus(3);

yMinusResult = pMinus(1)*xMinus.^2 + pMinus(2)*xMinus + pMinus(3);

hold on;
grid on;
xlabel('\omega');
ylabel('U');
plot(xPlus, yPlusResult, 'r');
plot(xMinus, yMinusResult, 'r');

LB=-inf;
UB=inf;
X0=1;
xopt1=lsqnonlin('cel',X0,LB,UB,[],data(22).u(1:5000),
data(22).t(1:5000),(data(22).y(1:5000)));
tf=data(22).t(end);
[t,x]=rk42([0;0],data(22).u,tf,K1);
```

Listing 1: Zadanie 1

1.2. Zadanie 2

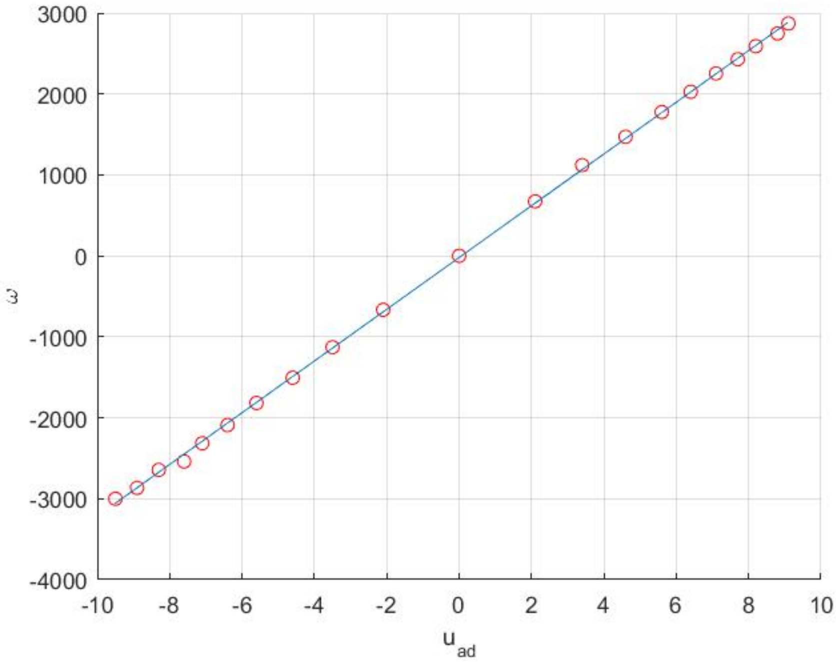
W zadaniu tym identyfikowanym obiektem jest silnik prądu stałego napędzający śmigło. Prędkość obrotowa śmigła jest mierzona za pomocą tachoprądnicy o stałej $\frac{0.52V}{1000obr.}$. Napięcie jest wzmacniane i mierzone przy pomocy przetwornika A/D. Model silnika i śmigła ma postać:

$$\dot{x} = K(u - H(x)),\ x(0) = x_0$$

Zależność pomiędzy prędkością obrotową śmigła, a wyjściem przetwornika A/D ma postać:

$$\omega = au + b$$

W pierwszej kolejności wyznaczono parametry a i b prostej. Prędkość otrzymano korzystając ze zmierzonego napięcia oraz stałej tachoprądnicy. Następnie skorzystano w metody najmniejszych kwadratów w celu wyznaczenia parametrów a oraz b. Otrzymano wartości a = 319,59 oraz b = -20,08. Prostą wraz z pomiarami prezentuje Rysunek 5.



Rysunek 5: Prosta $\omega = au + b$

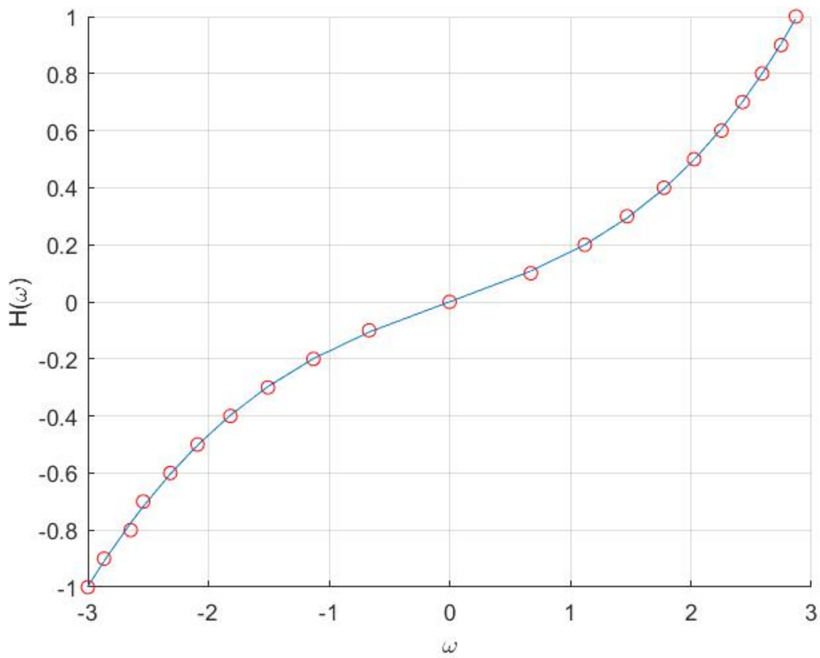
Kolejną czynnością było wyznaczenie współczynników funkcji $H(\omega)$. W tym zadaniu funkcja ta jest modelowana wielomianem 3 stopnia:

$$H(\omega) = az^3 + bz^2 + cz + d$$
$$z = 10^{-3}\omega$$

W tym celu ponownie skorzystano z metody najmniejszych kwadratów oraz wcześniej wyznaczonej prędkości. Otrzymano następujące wartości współczynników:

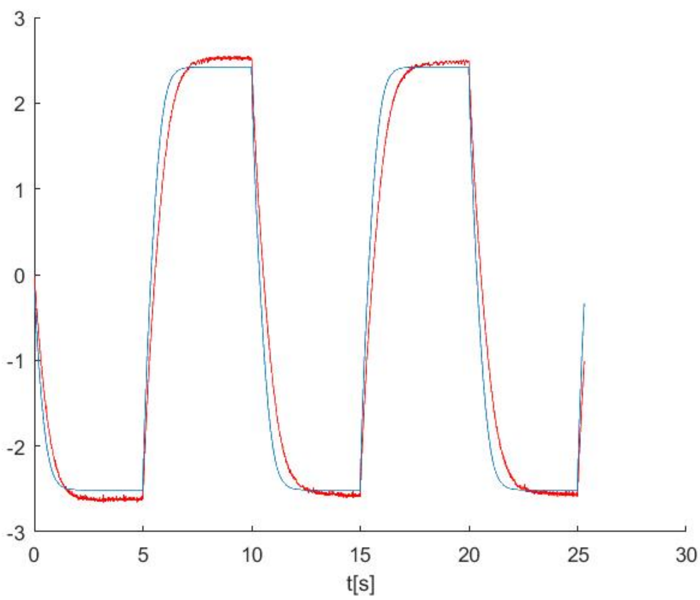
$$a = 0,022$$
$$b = 0,00027$$
$$c = 0,149$$
$$d = 0,00018$$

Otrzymaną funkcję przedstawia Rysunek 6.



Rysunek 6: Prosta $H(\omega)$

Następnie wyznaczono parametr K metodą opisaną w Zadaniu 1. Otrzymano wartość parametru k równą 7,064. Porównanie modelu z pomiarami przedstawia Rysunek 7.



Rysunek 7: Prosta $Pomiar - kolorczerwony, Model - kolorniebieski.$

1.2.1. Kod programu

Kod programu prezentuje listing poniżej.

```
clear all;
close all;

load('data_01_ident.mat')

% --- READING DATA ---- %
u = [-1.560 -1.490 -1.375 -1.321 -1.204 -1.087 -0.945 -0.783
-0.587 -0.347 0 0.350 0.583 0.765 0.924 1.054 1.172 1.263 1.347 1.429 1.493];
u_ad = [-9.5 -8.9 -8.3 -7.6 -7.1 -6.4 -5.6 -4.6 -3.5 -2.1 0 2.1
3.4 4.6 5.6 6.4 7.1 7.7 8.2 8.8 9.1];


% --- CALCULATING a and b --- %
scale = 0.52 / 1000;
velocity = u ./ scale;
abCoefs = polyfit(u_ad,velocity,1);
% --- CALCULATING f(u_ad) = Omega --- %
Hw = control;
figure();
hold on;
grid;
xlabel('u_{ad}');
ylabel('\omega')
plot(u_ad,velocity,'or');
plot(u_ad, polyval(abCoefs, u_ad),'-');
hold off;


% --- CALCULATING H(w) = f(w) --- %
% --- CALCULATING a, b, c and d H(w) --- %
wH = velocity;
wH = 10^-3 .* wH;
abcdCoefs = polyfit(wH,control,4);

figure()
hold on;
grid;
xlabel('\omega');
ylabel('H(\omega)');
plot(wH, control, 'or');
plot(wH, polyval(abcdCoefs, wH),'-');
hold off;


load('data_02_ident.mat')


figure()
wID=polyval(velocity,u_ad)/1000;
hold on;

plot(t,u,'g');
```

```
plot(t,wID,'r');


LB = [-100 -100];
UB = [100 100];
x0 = 3;
K = 7;
Kopt = lsqnonlin('cel', [x0 K]', LB, UB, [], u, t, wID, abcdCoefs);
```

Listing 2: Zadanie 2

2 Wnioski końcowe

W badanym zagadnieniu zgodnie z oczekiwaniami sprawdziły się modele wyższego rzędu. Ważnym i ciekawym aspektem rozważanego zagadnienia było modelowanie funkcji na dwóch rozłącznych przedziałach. Takie podejście daje przewagę nad rozważaniem jednego przedziału w przypadku gdy funkcja jest silnie nieliniowa w danym przedziale. Między innymi z tego też powodu funkcja liniowa dała gorsze rezultaty w Zadaniu 1.