

Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych
Instytut Automatyki i Informatyki Stosowanej

Projektowanie układów sterowania
(projekt grupowy)

Sprawozdanie z projektu i ćwiczenia laboratoryjnego
nr 2, zadanie nr 7

Autorzy:
Grochowina Mateusz
Winnicki Konrad
Zgorzelski Jan

Warszawa, 4 kwietnia 2019

Spis treści

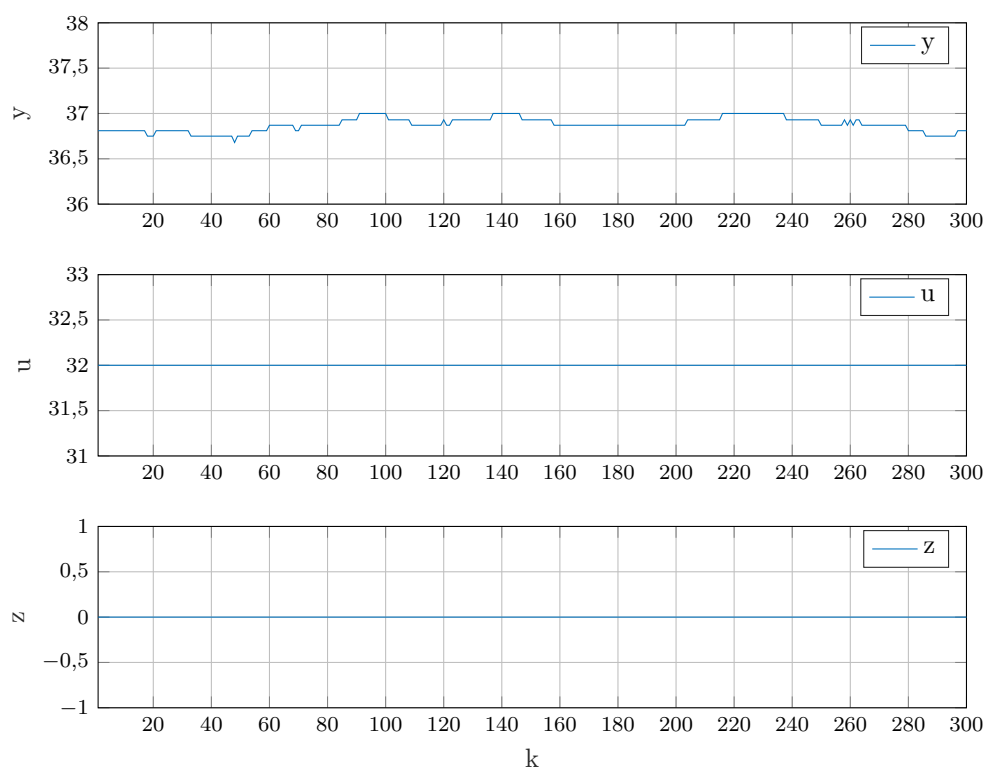
1. Projekt	2
2. Laboratorium	3
2.1. Poprawność sterowania i punkt pracy stanowiska	3
2.2. Wyznaczenie odpowiedzi skokowej toru zakłócenie-wyjście	4
2.2.1. Odpowiedzi skokowe obiektu	4
2.2.2. Właściwości statyczne obiektu	5
2.3. Wyznaczenie aproksymowanych wektorów s i s_z	7
2.3.1. Odpowiedź skokowa wyjścia od zakłócenia	7
2.3.2. Aproksymacja odpowiedzi skokowych	7
2.4. Regulator DMC	8
2.4.1. Program do symulacji algorytmu DMC	8
2.4.2. Dobór parametrów regulatora	12
2.5. Wpływ skokowej zmiany sygnału zakłócenia	13
2.5.1. Dobór parametru D_z	13
2.5.2. Regulacja bez uwzględnienia zakłócenia	13
2.5.3. Regulacja z uwzględnieniem zakłócenia	13

1. Projekt

2. Laboratorium

2.1. Poprawność sterowania i punkt pracy stanowiska

Przeprowadzono symulację odpowiedzi procesu dla punktu pracy. Ustalone zostały stałe sygnały sterujące $G1 = 32$ i $Z = 0$.



Rys. 2.1. Punkt pracy obiektu

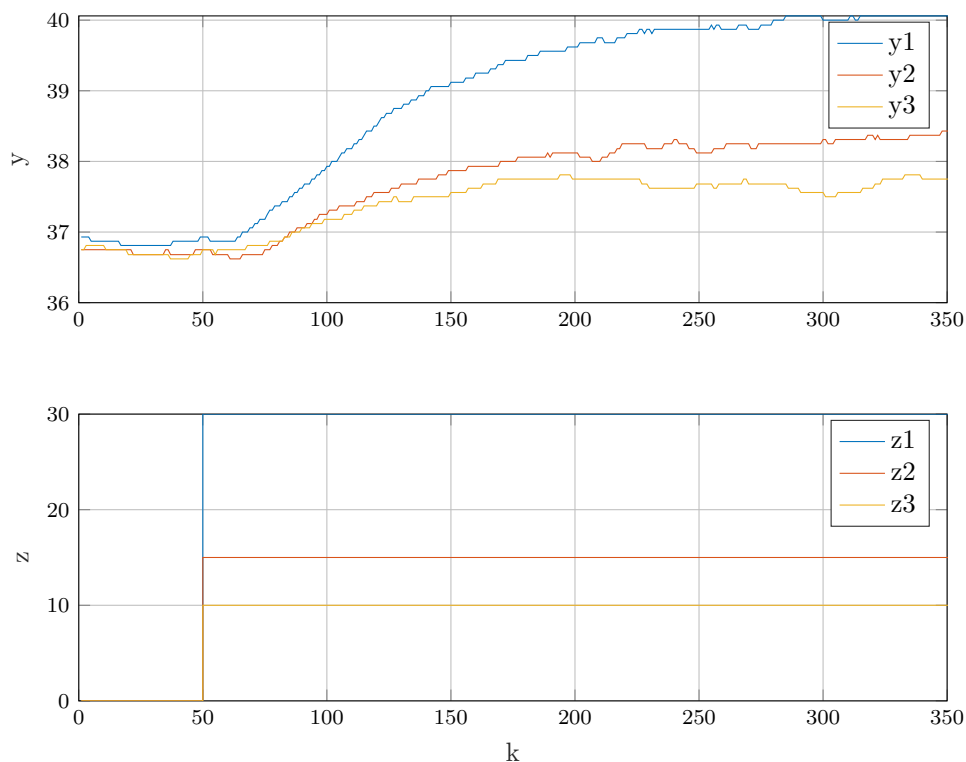
Ustalona wartość wyjścia obiektu wynosi $T1 = 35,4^{\circ}C$.

2.2. Wyznaczenie odpowiedzi skokowej toru zakłócenie-wyjście

Odpowiedzi skokowe toru zakłócenie-wyjście zostały wyznaczone symulacyjnie dla trzech zmian sygnału zakłócenia.

2.2.1. Odpowiedzi skokowe obiektu

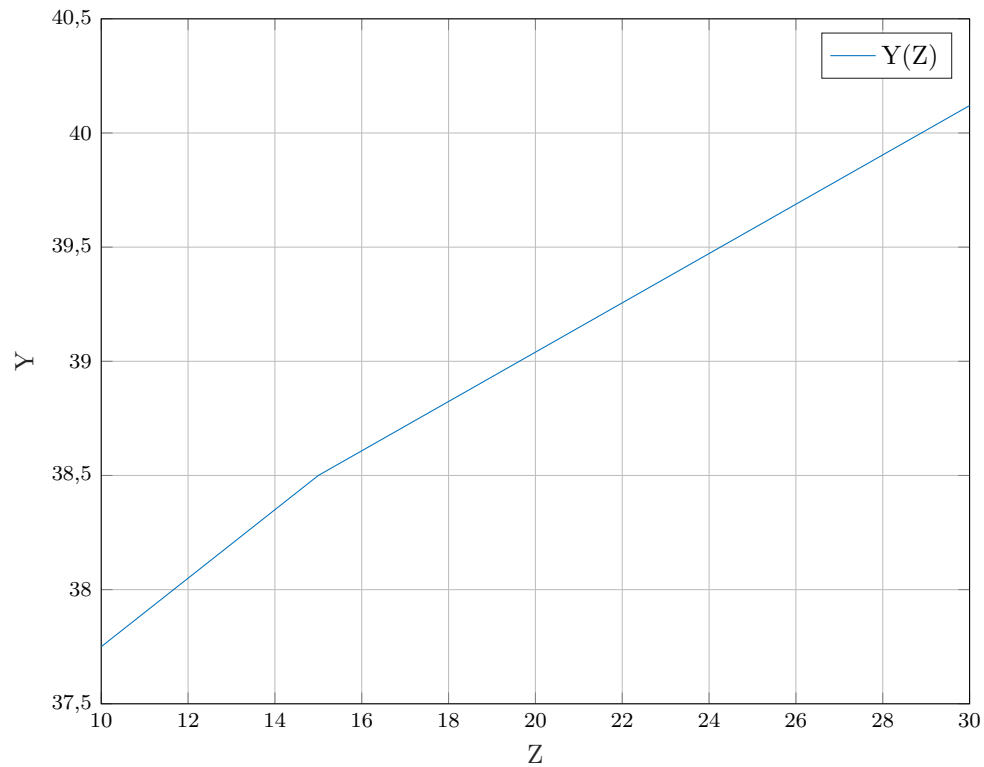
Do uzyskania odpowiedzi skokowych dla tego toru ustawiono sygnał wejściowy na stałą wartość $U=32$, przy ustabilizowanej temperaturze $T_1 = 35,4^{\circ}\text{C}$ przeprowadzone zostały skoki sterowania z $Z = 0$ na 10, 15 oraz 30.



Rys. 2.2. Odpowiedzi skokowe obiektu od zakłócenia

2.2.2. Właściwości statyczne obiektu

Dzięki uzyskanym odpowiedziom skokowym otrzymano charakterystykę statyczną zakłóceń.



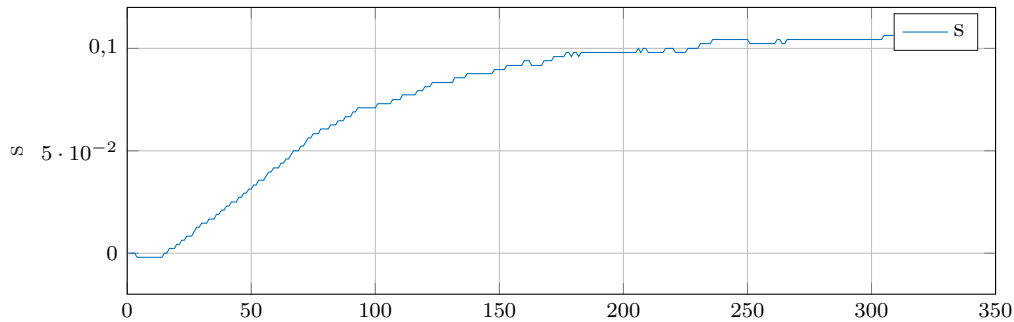
Rys. 2.3. Charakterystyka statyczna wyjścia od zakłócenia

Na podstawie wykresu charakterystyki statycznej można założyć, że właściwości statyczne procesu są z dobrym przybliżeniem liniowe.

Wzmocnienie statyczne procesu określone zostało na $K = 0,1046$.

2.3. Wyznaczenie aproksymowanych wektorów s i s_z

2.3.1. Odpowiedź skokowa wyjścia od zakłócenia



Rys. 2.4. Odpowiedź skokowa obiektu od zakłócenia

2.3.2. Aproksymacja odpowiedzi skokowych

Aproksymacja została wykonana jako człon inercyjny drugiego rzędu z opóźnieniem. Opisany jest on następującą transmitancją:

$$G(s) = \frac{K}{(sT_1 + 1)(sT_2 + 1)} e^{-T_d s}$$

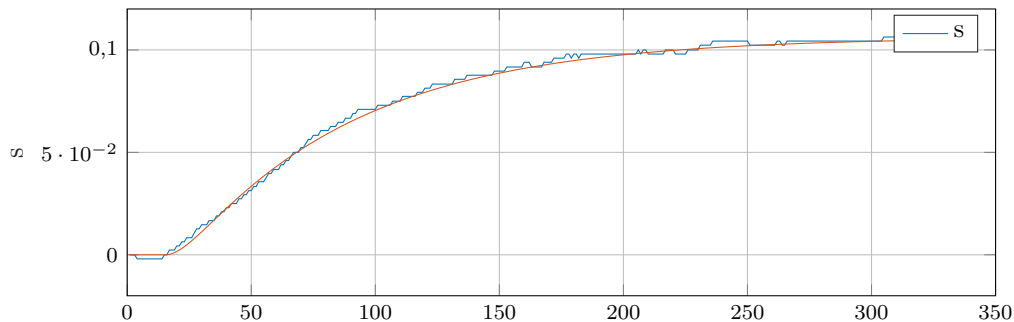
Powyższa transmitancja po przekształceniu do dziedziny czasu dyskretnego i przejściu na postać równania różnicowego:

$$y[k] = b_1 u[k - T_D - 1] + b_2 u[k - T_D - 2] + a_1 y[k - 1] + a_2 y[k - 2]$$

Na podstawie danych pozyskanych ze stanowiska laboratoryjnego dobrano parametry T_1 , T_2 , K , T_d , tak aby błąd dopasowania, rozumiany jako suma kwadratów kolejnych uchybów sterowania, był jak najmniejszy. Należy jednak pamiętać, że wielkość T_d może przyjmować tylko wartości całkowite (ze względu na zastosowany czas dyskretny). W celu doboru parametrów modelu wykorzystano funkcję optymalizacyjną *fmincon* programu MATLAB, jako parametr optymalizacji wybrano błąd dopasowania (rozumiany jako suma kwadratów błędów dla kolejnych elementów odpowiedzi skokowych), ponieważ im mniejszy błąd dopasowania tym lepsza aproksymacja.

Otrzymane parametry aproksymacji odpowiedzi skokowej od zakłócenia to:

$$T_1 = 7,59 \quad T_2 = 70,4 \quad K = 0,11 \quad T_d = 14,0$$



Rys. 2.5. Aproksymowana odpowiedź skokowa od zakłócenia

2.4. Regulator DMC

2.4.1. Program do symulacji algorytmu DMC

Inicjacja parametrów

```
%zadanie 4 – Skrypt relizujący algorytm DMC regulatora  
%nastawy regulatora DMC
```

```
addpath ( 'F:\SerialCommunication' ) ; % add a path  
initSerialControl COM5 % initialise com port
```

```
%wczytanie aproksymowanych odpowiedzi skokowych  
load( '..\zad3\s_skok_s_32_55_approx.mat' );  
load( '..\zad3\s_skok_z_0_30_approx.mat' );  
% gorne ograniczenie sterowania  
load( 'u_max.mat' );  
D = length(s); % horyzont dynamiki  
Dz = length(s_z); % horyzont dynamiki  
N = 200; %horyzont predykcji  
Nu = 10; %horyzont sterowania  
lambda = 0.8;
```

```
%warunki poczatkowe  
kk = 2000; %koniec symulacji  
start = 300; %start symulacji  
start_z = 1000; %start symulacji  
u=ones(1, kk)*39; %sygnal sterujacy  
E = 0;  
zakl = 1; %czy sa zaklocenia 1-T/0-N  
licz_z = 0; %czy mamy uwzgledniac 1-T/0-N
```

```
%parametry symulacji  
u_delta = zeros(1, D-1);  
z_delta = zeros(1, Dz);  
y_zad=ones(1, kk+D)*35.4;  
y_zad(1, D+50:kk+D)=37.8  
y = ones(1, kk)*35.4; % wektor wyjsc obiektu  
z = zeros(1, kk);
```

```
if zakl == 1  
    z(start_z+D:kk) = 30;  
end
```

Obliczanie macierzy DMC

```

% Macierz predykcji
Mp = zeros(N,D-1);
for i = 1:N
    for j = 1:D-1
        if i+j <= D
            Mp(i,j) = s(i+j)-s(j);
        else
            Mp(i,j) = s(D)-s(j);
        end
    end
end

% Macierz M
M = zeros(N,Nu);
for i = 1:N
    for j = 1:Nu
        if (i >= j)
            M(i,j) = s(i-j+1);
        end
    end
end

% Macierz Mzp
if zakl == 1 && licz_z == 1
    Mzp = zeros(N,Dz);
    for l = 1:Dz
        Mzp(1,l) = s_z(l);
    end
    for i = 2:N
        for j = 1:Dz-1
            if i+j <= Dz
                Mzp(i,j) = s_z(i+j-1)-s_z(j);
            else
                Mzp(i,j) = s_z(Dz)-s_z(j);
            end
        end
    end
end

% Obliczanie parametrow regulatora
mac_lam = lambda*eye(Nu);
psi = eye(N);
K = ((M*psi*M+mac_lam)^(-1))*M'*psi;

```

Pętla główna regualtora

```

%% ustalenie początkowych sterowan, W1=50, G1=U
sendControls ([ 1 , 2 , 3 , 4 , 5 , 6] , [ 50 , 0 , 0 , 0 , u(1) , 0]) ;

% Regulacja
for k = D:kk
    %% obtaining measurements
    measurements = readMeasurements (1:7) ; % read measurements
    y(k)=measurements(1); % powiększamy wektor y o element Y

    % przesunięcie wektora delta u
    for n = D-1:-1:2
        u_delta(n) = u_delta(n-1);
    end

    % jesli istnieja zaklocenia i uwzgledniamy je
    % przesunięcie wektora delta z
    if zakl == 1 && licz_z == 1
        for n = Dz:-1:2
            z_delta(n) = z_delta(n-1);
        end
        z_delta(1) = z(k) - z(k-1);
    end

    uchyb = y_zad(k) - y(k);
    %wyznaczenie zmiany sterowania
    if zakl == 1 && licz_z == 1
        u_delta(1) = sum(K(1,:))*uchyb - K(1,:)*(Mp*u_delta' + Mzp*z_delta');
    else
        u_delta(1) = sum(K(1,:))*uchyb - K(1,:)*(Mp*u_delta');
    end
    u(k) = u(k-1) + u_delta(1);

    % ograniczenia sterowania
    if u(k)>u_max
        u(k)=u_max;
    end
    if u(k)<0.0
        u(k) = 0.0;
    end

    u_delta(1) = u(k) - u(k-1);

    E = E + (y_zad(k) - y(k))^2;

```

Generacja wykresów w czasie rzeczywistym

```
%wykres wyjscia i wartosci zadanej
figure(1);
clf(1);
hold on;
title('y');
grid on;
xlabel('time');
ylabel('value');
plot(y(1:k)); % wyswietlamy y w czasie
plot(y_zad(1:k));
legend('y', 'y_z_a_d')

%wykres sterowania
figure(2);
clf(2);
hold on;
title('u');
grid on;
xlabel('time');
ylabel('value');
plot(u(1:k)); % wyswietlamy u w czasie

%wykres zaklocenia
figure(3);
clf(3);
hold on;
title('z');
grid on;
xlabel('time');
ylabel('value');
plot(z(1:k)); % wyswietlamy u w czasie
%legend('u')

drawnow

%% sending new values of control signals
sendControlsToG1AndDisturbance(u(k), z(k));
%% synchronising with the control process
waitForNewIteration(); % wait for new iteration
end
```

2.4.2. Dobór parametrów regulatora

Dobre parametry regulatora DMC:

- $D = 684$,
- $N = 200$,
- $N_u = 10$,
- $\lambda = 0.8$,
- $D_z = 316$.

2.5. Wpływ skokowej zmiany sygnału zakłócenia

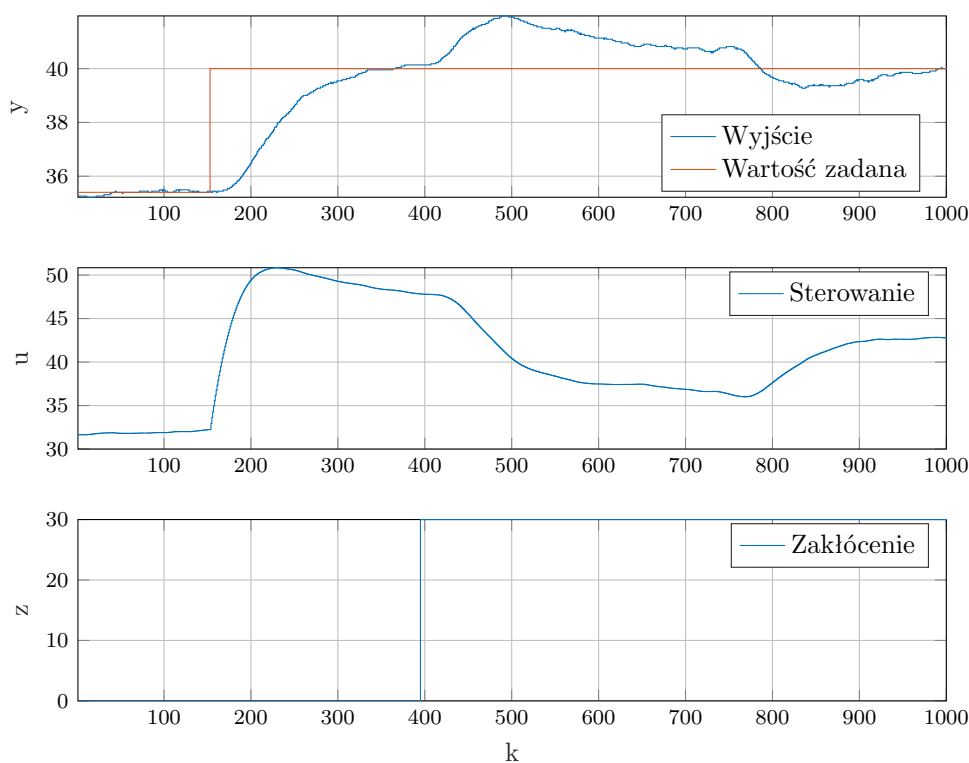
2.5.1. Dobór parametru D_z

Parametr D_z jest to liczba próbek, dla której następuje stabilizacja odpowiedzi skokowych toru zakłóceń, dobrano go na podstawie analizy odpowiedzi skoku zakłócenia Z z 0 na 30.

D_z wynosi 316

2.5.2. Regulacja bez uwzględnienia zakłócenia

Po osiągnięciu przez proces wartości zadanej wyjścia następuje zmiana sygnału zakłócenia z wartości 0 na 30.



Rys. 2.6. Reeregulacja bez uwzględnienia zakłócenia

2.5.3. Regulacja z uwzględnieniem zakłócenia

Uwzględnienie zakłócenia w algorytmie regulatora pozwala na polepszenie jakości regulacji poprzez pomiar i uwzględnienie sygnału zakłócającego w procesie regulacji.

Uwzględnienie mierzalnego zakłócenia w algorytmie regulacji jest bardzo dobrym rozwiązaniem, odsprężanie zakłócenia powoduje kompensację uchybu regulacji, a co za tym idzie wskaźnik jakości jest lepszy, a sama regulacja uznana jest za lepszą.