

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

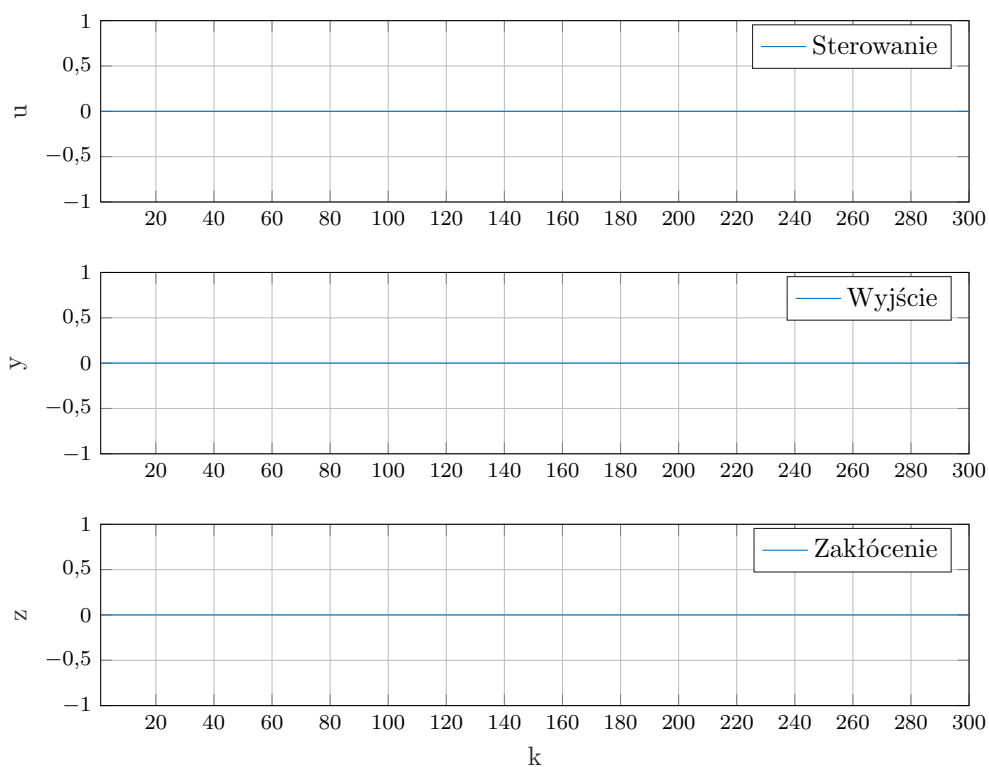
Spis treści

1. Projekt	2
1.1. Poprawność podanego punktu pracy	2
1.2. Wyznaczenie symulacyjne odpowiedzi skokowych	3
1.2.1. Odpowiedź wyjścia na skok wejścia	3
1.2.2. Odpowiedź wyjścia na skok zakłócenia	4
1.2.3. Charakterystyka statyczna $Y(U, Z)$	5
1.3. Wyznaczenie wektorów s i s_z	6
1.3.1. Wyznaczenie wektora s	6
1.3.2. Wykres wektora s	6
1.3.3. Wyznaczenie wektora s_z	7
1.3.4. Wykres wektora s_z	7
1.4. Regulator DMC	8
1.4.1. Program do symulacji algorytmu DMC	8
1.4.2. Dobór parametrów regulatora	10
1.5. Wpływ skokowej zmiany sygnału zakłócenia	11
1.5.1. Dobór parametru D_z	11
1.5.2. Regulacja bez uwzględnienia zakłócenia	11
1.5.3. Regulacja z uwzględnieniem zakłócenia	12
1.5.4. Porównanie wskaźnika jakości	12
1.6. Wpływ ciągłej sinusoidalnej zmiany sygnału zakłócenia	13
1.6.1. Regulacja bez uwzględnienia zakłócenia	13
1.6.2. Regulacja z uwzględnieniem zakłócenia	14
1.7. Odporność algorytmu przy błędach pomiarowych sygnału zakłócenia	15
1.7.1. Regulacja bez uwzględnienia zakłócenia	15
1.7.2. Regulacja z uwzględnieniem zakłócenia	17
2. Laboratorium	19
2.1. Poprawność sterowania i punkt pracy stanowiska	19
2.2. Wyznaczenie odpowiedzi skokowej toru zakłócenie-wyjście	20
2.2.1. Odpowiedzi skokowe obiektu	20
2.2.2. Właściwości statyczne obiektu	21
2.3. Wyznaczenie aproksymowanych wektorów s i s_z	22
2.3.1. Odpowiedź skokowa wyjścia od zakłócenia	22
2.3.2. Aproksymacja odpowiedzi skokowych	22
2.4. Regulator DMC	23
2.4.1. Program do symulacji algorytmu DMC	23
2.4.2. Dobór parametrów regulatora	27
2.5. Wpływ skokowej zmiany sygnału zakłócenia	28
2.5.1. Dobór parametru D_z	28
2.5.2. Regulacja bez uwzględnienia zakłócenia	28
2.5.3. Regulacja z uwzględnieniem zakłócenia	28

1. Projekt

1.1. Poprawność podanego punktu pracy

Przeprowadzono symulację odpowiedzi procesu dla punktu pracy. Ustalono stałe sterowanie $U_{pp} = 0$ oraz stałe zakłócenie $Z_{pp} = 0$.



Rys. 1.1. Punkt pracy obiektu symulacji

Wynik:

Uzyskany punkt pracy wyjścia $Y_{pp} = 0$.

Wniosek:

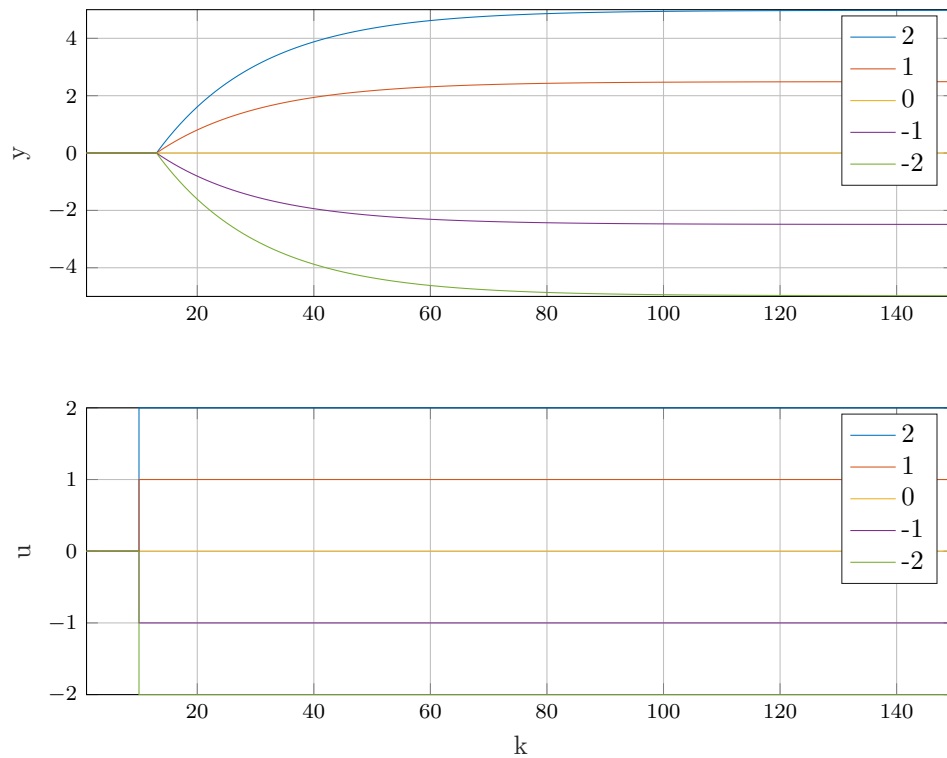
Stała wartość wyjścia oznacza poprawność danych punktu pracy.

1.2. Wyznaczenie symulacyjne odpowiedzi skokowych

Odpowiedzi skokowe torów wejście-wyjście i zakłócenie-wyjście zostały wyznaczone symulacyjnie dla pięciu zmian sygnału sterującego oraz pięciu zmian zakłócenia .

1.2.1. Odpowiedź wyjścia na skok wejścia

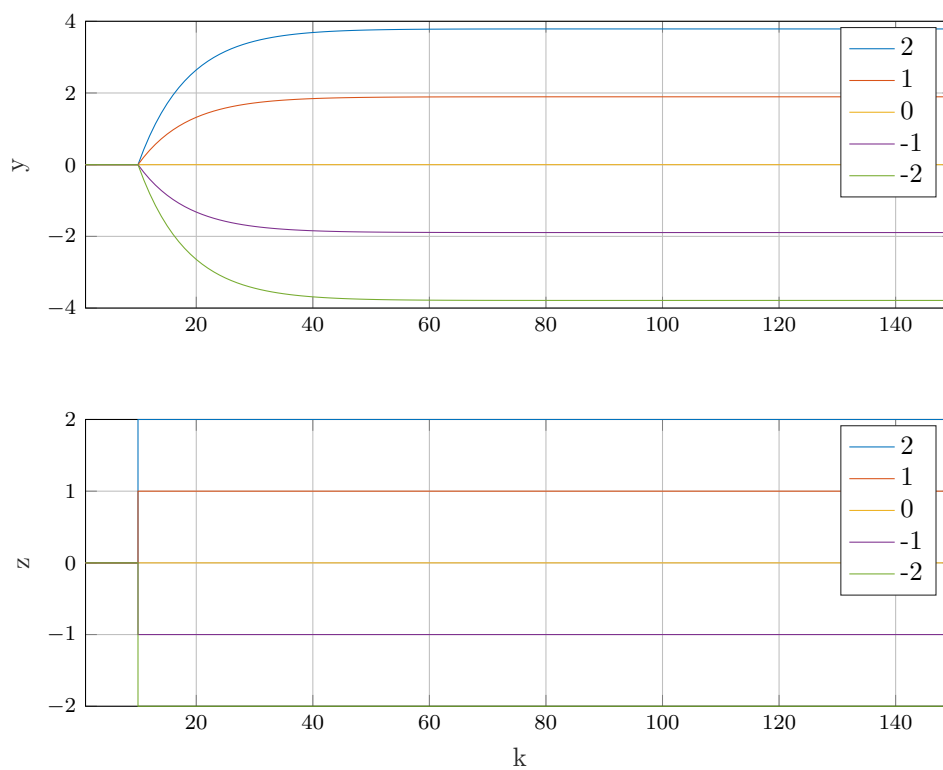
Do uzyskania odpowiedzi skokowych dla tego toru ustawiono zakłócenie na stałą wartość $Z_{pp} = 0$ oraz przeprowadzona została seria skoków sterowania U



Rys. 1.2. Odpowiedzi skokowe obiektu od sterowania

1.2.2. Odpowiedź wyjścia na skok zakłócenia

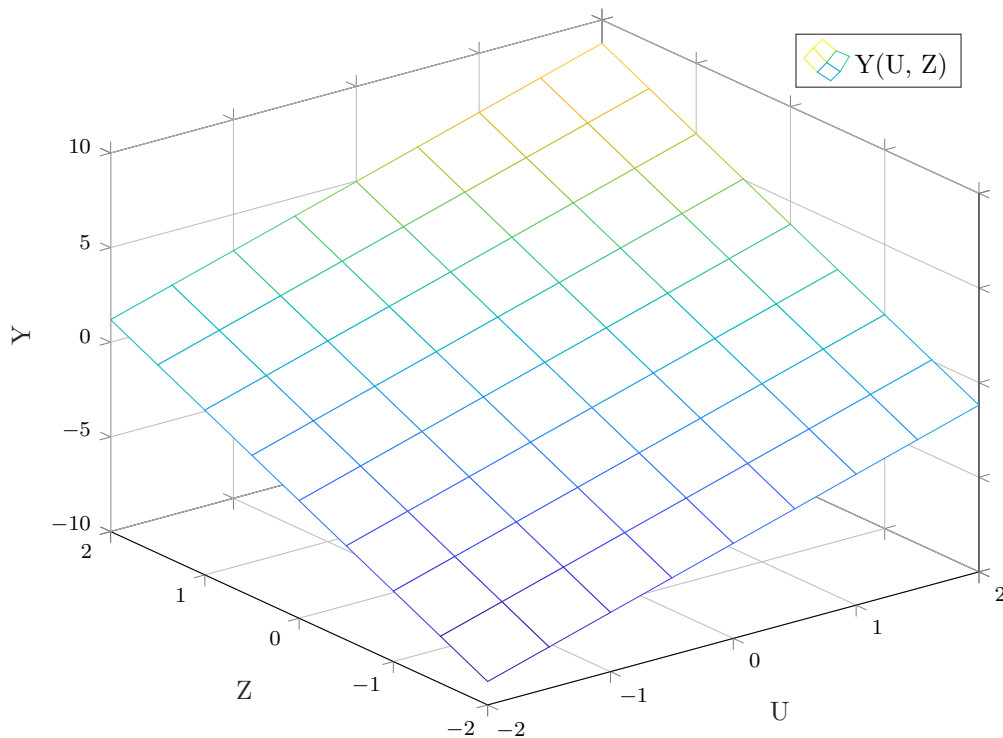
Do uzyskania odpowiedzi skokowych dla tego toru ustawiono sygnał wejściowy na stałą wartość $U_{pp} = 0$ oraz przeprowadzona została seria skoków zakłócenia Z



Rys. 1.3. Odpowiedzi skokowe obiektu od zakłócenia

1.2.3. Charakterystyka statyczna $Y(U, Z)$

Wykonując zestaw symulacji dla każdej kombinacji wartości sygnałów U i Z z ich zakresu wyznaczono charakterystykę statyczną $Y(U, Z)$



Rys. 1.4. Charakterystyka statyczna obiektu $y(u, z)$

Na podstawie wykresu charakterystyki statycznej można ustalić, że właściwości statyczne procesu są liniowe.

Wzmocnienie statyczne toru wejście-wyjście wynosi $K_u = 2,4893$.

Wzmocnienie statyczne toru zakłócenie-wyjście wynosi $K_z = 1,8933$.

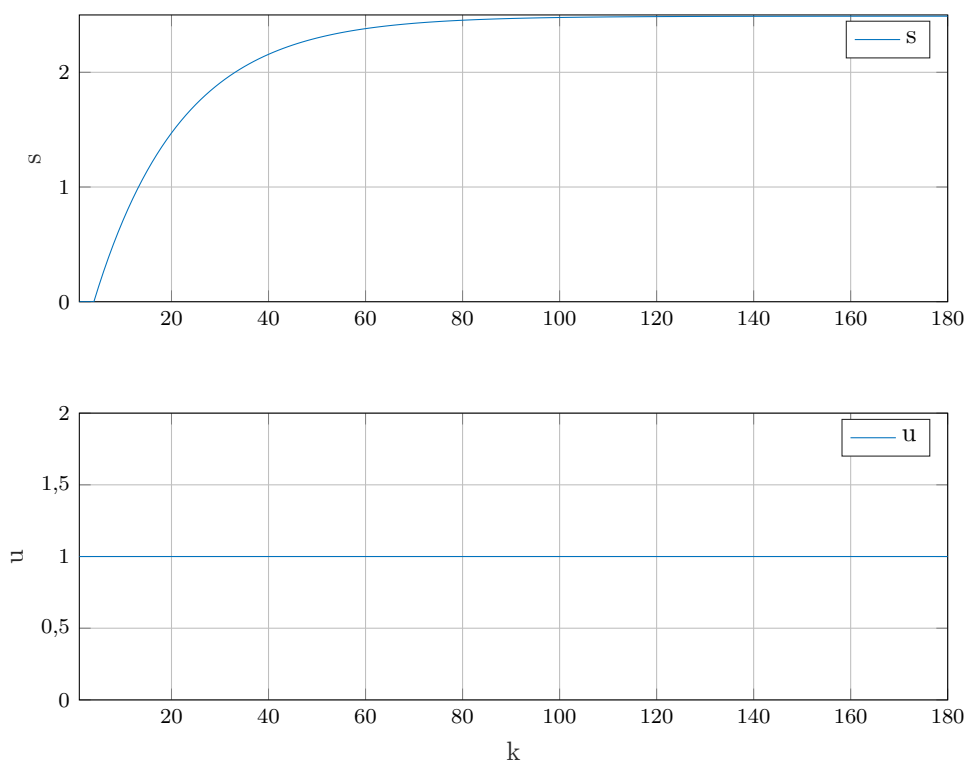
1.3. Wyznaczenie wektorów s i s_z

1.3.1. Wyznaczenie wektora s

Uzyskaną odpowiedź procesu na zmianę sygnału sterującego z punktu pracy $U_{pp} = 32$ na $U_{max} = 55$ przekształcono w następujący sposób:

- Ograniczono (przycięto) czas zmiany sterowania u oraz wyjścia y od chwili skoku do ustabilizowania,
- Wykres sterowania u przesunięty został o wartość początkową U_{pp} w dół,
- Wykres wyjścia y przesunięty został o wartość początkową Y_{pp} w dół,
- Wykres sterowania u i wyjścia y podzielono przez wysokość skoku sterowania u .

1.3.2. Wykres wektora s



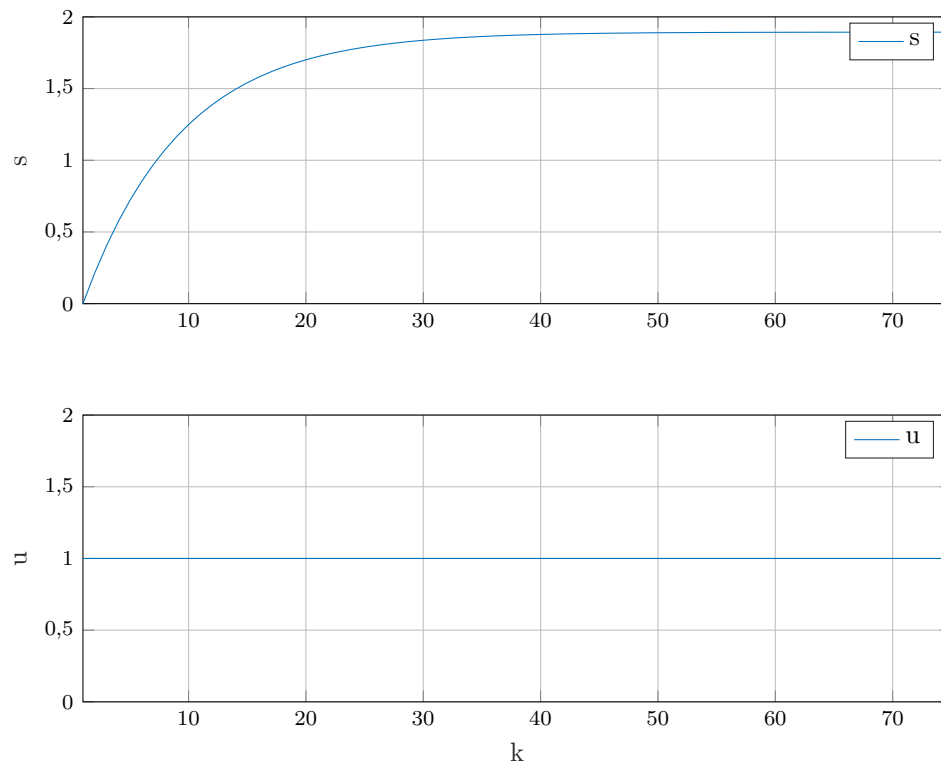
Rys. 1.5. Wektor s

1.3.3. Wyznaczenie wektora s_z

Uzyskaną odpowiedź procesu na zmianę sygnału sterującego z punktu pracy $Z_{pp} = 0$ na $Z_{max} = 30$ przekształcono w następujący sposób:

- Ograniczono (przycięto) czas zmiany sterowania z oraz wyjścia y od chwili skoku do ustabilizowania,
- Wykres sterowania z przesunięty został o wartość początkową Z_{pp} w dół,
- Wykres wyjścia y przesunięty został o wartość początkową Y_{pp} w dół,
- Wykres sterowania z i wyjścia y podzielono przez wysokość skoku sterowania z .

1.3.4. Wykres wektora s_z



Rys. 1.6. Wektor s_z

1.4. Regulator DMC

Algorytm DMC (Dynamic Matrix Control) algorytm regulacji predykcyjnej. Do predykcji wykorzystuje się model procesu w postaci odpowiedzi skokowych. W algorytmie DMC dynamika obiektu regulacji modelowana jest dyskretnymi odpowiedziami skokowymi, które opisują reakcję wyjścia na skok jednostkowy sygnału sterującego.

1.4.1. Program do symulacji algorytmu DMC

Inicjacja parametrów

```
%Skrypt relizujący algorytm DMC regulatora z zakłoceniem
%nastawy regulatora DMC
D = 180;%horyzont dynamiki
Dz = 75;%horyzont dynamiki zakłocen
N = 180;%horyzont predykcji
Nu = 4;%horyzont sterowania
lambda = 9;

%warunki początkowe
kk = 300; %koniec symulacji
start = 50; %start symulacji
start_z = 150; %start symulacji
u = zeros(1, kk); %sygnał sterujący
E = 0;
zakl = 1; %czy są zakłocenia 1-T/0-N
licz_z = 0; %czy mamy uwzględnić 1-T/0-N
load('s_u.mat')
load('s_z.mat')
s = s_u;
%parametry symulacji
u_delta = zeros(1, D-1);
z_delta = zeros(1, Dz);
y_zad = zeros(kk, 1);
y_mod = zeros(kk, 1);
y_zad(start:end) = 10
z = zeros(kk, 1);
if zakl == 1
    z(start_z:kk) = 1;
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 parametrów regulatora
mac_lam = lambda*eye(Nu);
psi = eye(N);
K = ((M'*psi*M+mac_lam)^(-1))*M'*psi;

% Symulacja
```

Pętla główna regulatora

```
% Symulacja
for k = 20:kk
    % Równanie różnicowe z punktu drugiego
    y_mod(k)=symulacja_obiektu7y(u(k-4),u(k-5),z(k-1),z(k-2),y_mod(k-1),y_mod(k-2))

    % Regulator
    for n = D-1:-1:2
        u_delta(n) = u_delta(n-1);
    end

    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_mod(k);
    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);

    E = E + (y_zad(k) - y_mod(k))^2;
end
```

1.4.2. Dobór parametrów regulatora

Dobre parametry regulatora DMC:

- $D = 180$,
- $N = 180$,
- $N_u = 4$,
- $\lambda = 9$,
- $D_z = 75$.

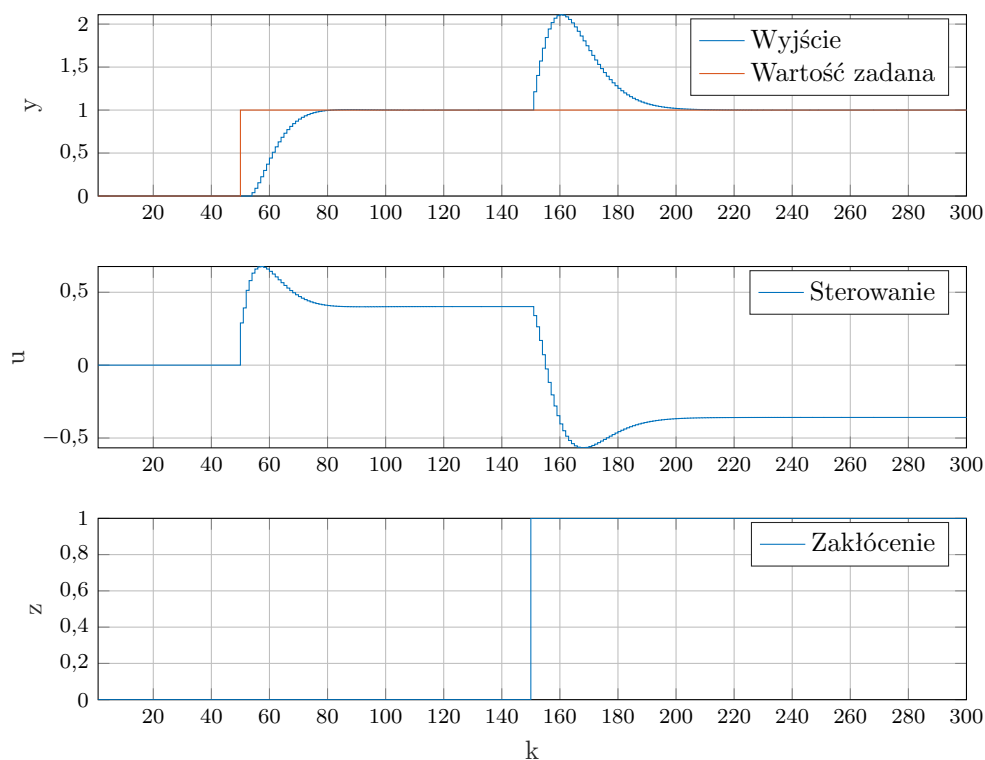
1.5. Wpływ skokowej zmiany sygnału zakłócenia

1.5.1. 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 wartości 0 na 1. Wyznaczona wartość D_z wynosi $= 75$

1.5.2. Regulacja bez uwzględnienia zakłócenia

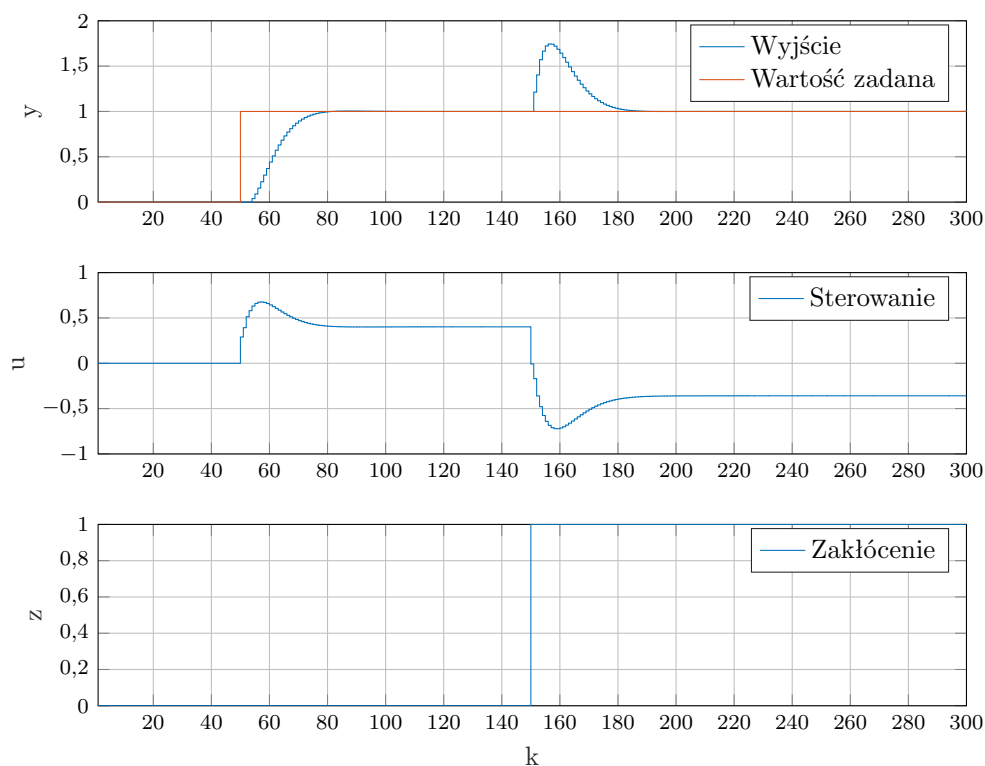
Po osiągnięciu przez proces wartości zadanej wyjścia następuje zmiana sygnału zakłócenia z wartości 0 na 1.



Rys. 1.7. Regulacja bez uwzględnienia zakłócenia

1.5.3. Regulacja z uwzględnieniem zakłócenia

Po osiągnięciu przez proces wartości zadanej wyjścia następuje zmiana sygnału zakłócenia z wartości 0 na 1.



Rys. 1.8. Regulacja z uwzględnieniem zakłócenia

1.5.4. Porównanie wskaźnika jakości

Dla symulacji regulowanego obiektu bez pomiaru zakłóceń wynosi on $E = 27.3670$

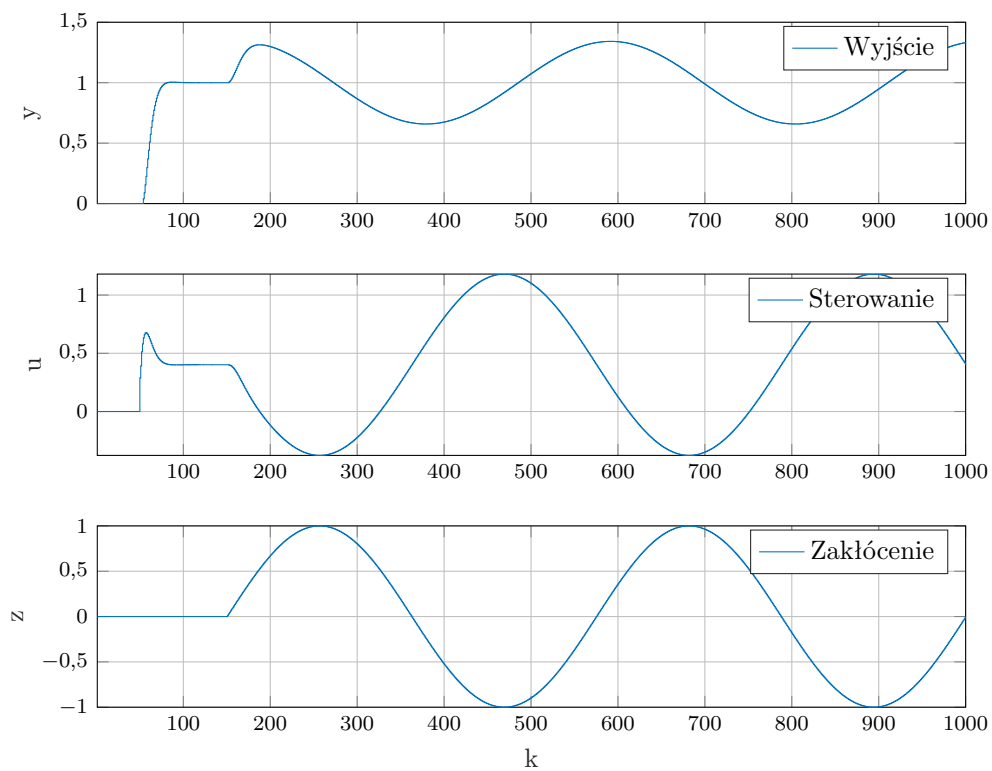
Dla symulacji regulowanego obiektu z pomiarem zakłóceń wynosi on $E = 14.8524$

Wnioski:

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.

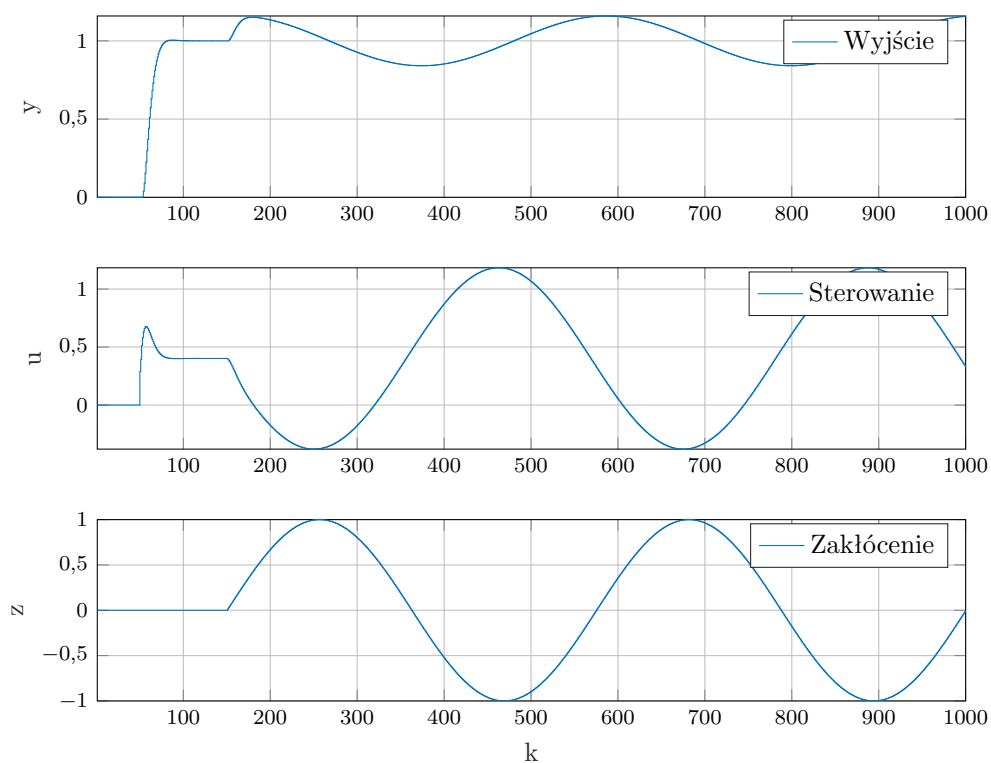
1.6. Wpływ ciągłej sinusoidalnej zmiany sygnału zakłócenia

1.6.1. Regulacja bez uwzględnienia zakłócenia



Rys. 1.9. Regulacja bez uwzględnienia zakłócenia

1.6.2. Regulacja z uwzględnieniem zakłócenia



Rys. 1.10. Regulacja uwzględniająca zakłócenie

Wnioski:

Zakłócenie sinusoidalne wpłynęło bardzo negatywnie na układ, w którym nie jest uwzględniona kompensacja zakłócenia co spowodowało, że wskaźnik jakości wynosi: 34,2 .

Uwzględniając mierzone zakłócenie w algorytmie udało się skompensować uchyb co poprawiło wskaźnik jakości do 20,8:

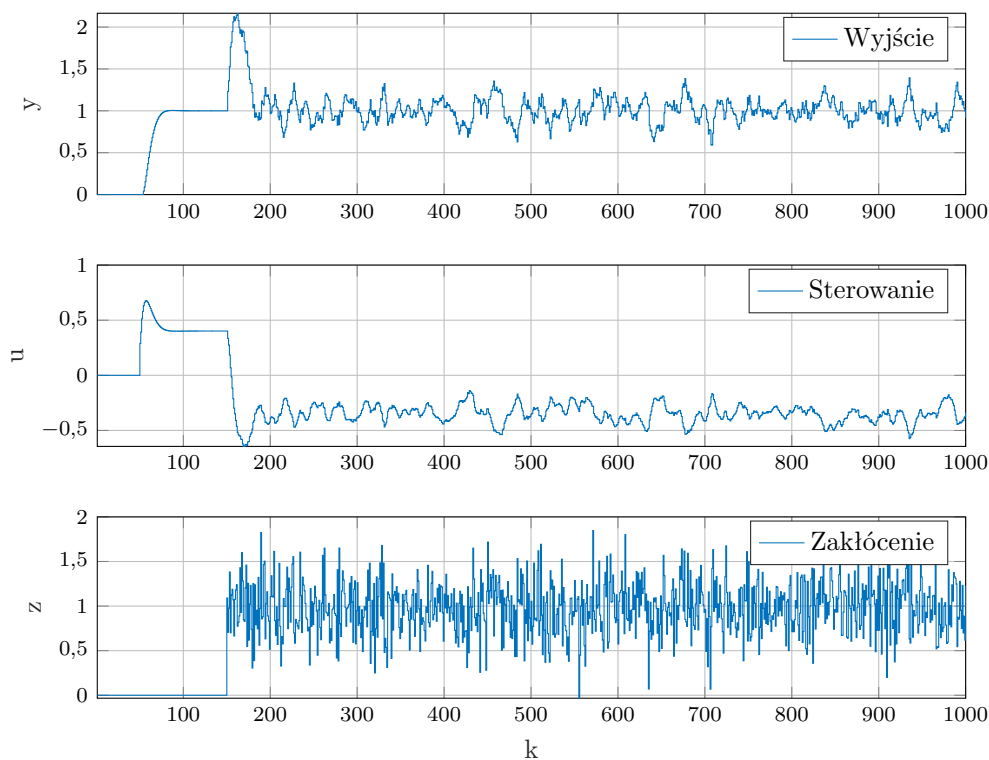
Zakłócenie zmienne sinusoidalne jest trudniejsze do kompensacji niż zwykły skok zakłócenia, ponieważ wymaga ono innego modelu zakłóceń, którego nie zastosowano.

1.7. Odporność algorytmu przy błędach pomiarowych sygnału zakłócenia

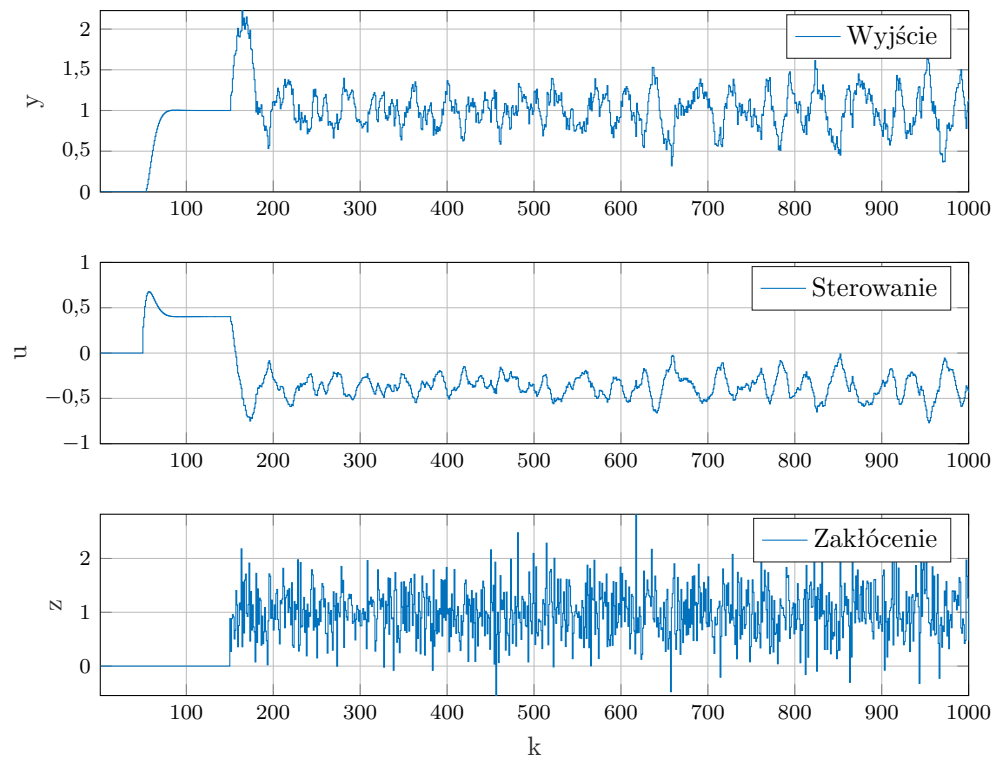
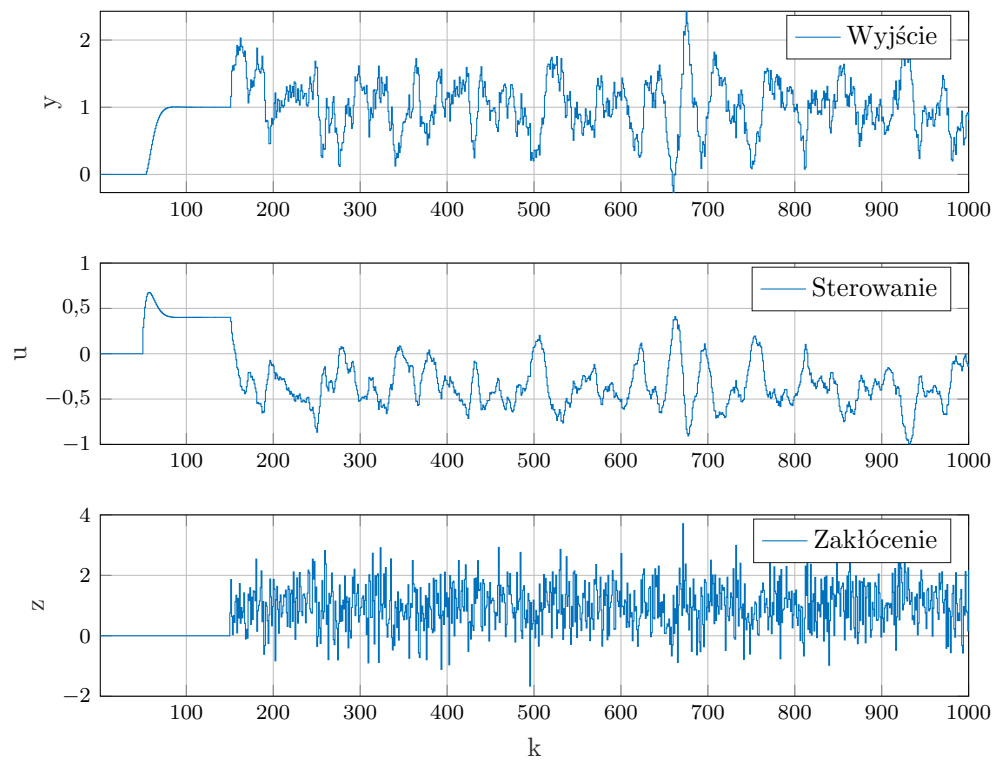
Szum pomiarowy wygenerowano za pomocą dodania do wartości sygnału zakłócenia dodajemy funkcję MATLAB'a `normrnd()`, gdzie jako parametry podajemy 0 oraz sigma.

Dzięki tej funkcji sygnał zakłócenia zmienia się zgodnie z rozkładem Gauss'a. Poprzez zwiększanie parametru sigma, zwiększamy zmiany sygnału zakłócenia, a co za tym idzie - większe zakłócenia. Rozważono trzy różne wartości zakłócenia:

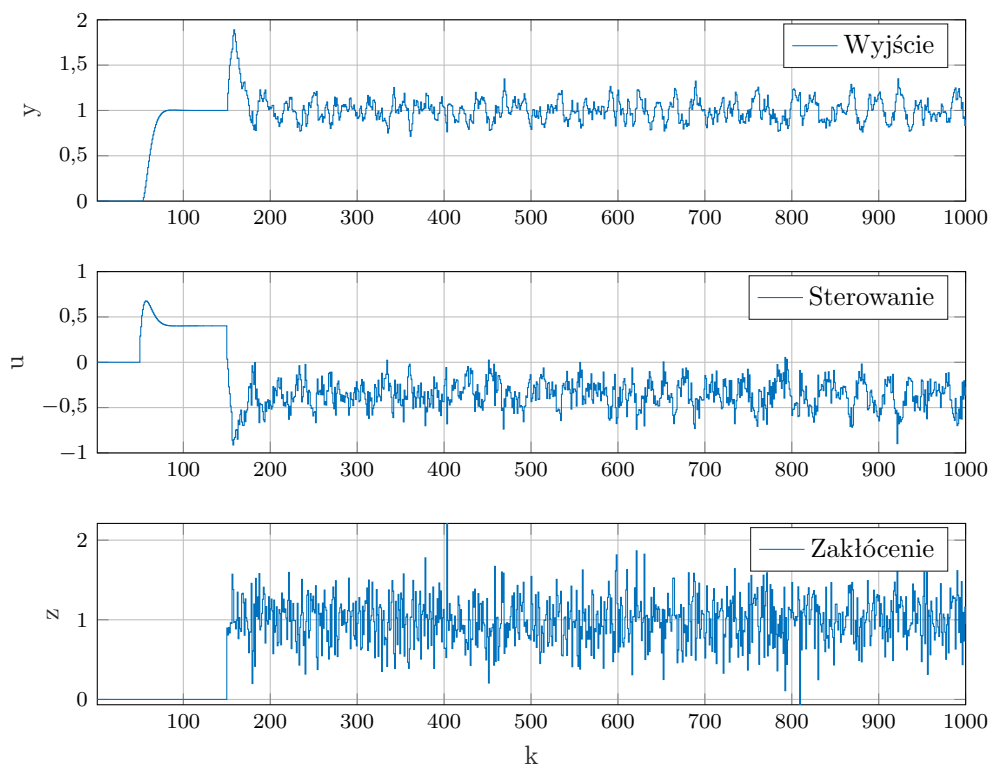
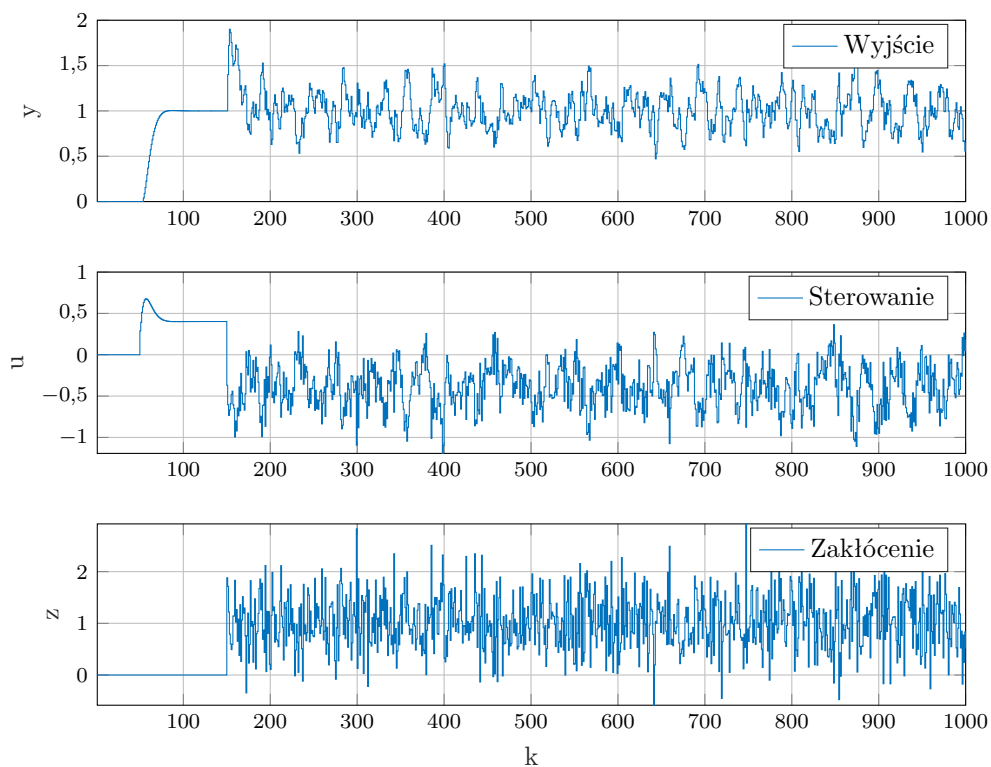
1.7.1. Regulacja bez uwzględnienia zakłócenia

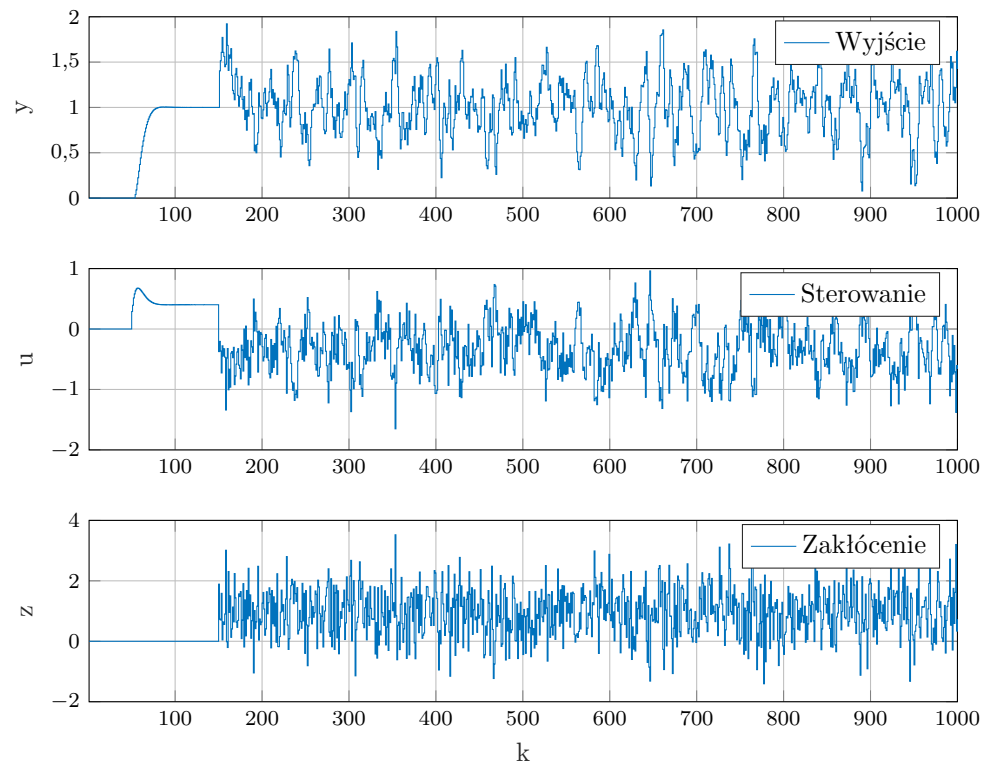


Rys. 1.11. Regulacja bez uwzględnienia zakłócenia, sigma = 0,3

Rys. 1.12. Regulacja bez uwzględnienia zakłócenia, $\sigma = 0,5$ Rys. 1.13. Regulacja bez uwzględnienia zakłócenia, $\sigma = 0,8$

1.7.2. Regulacja z uwzględnieniem zakłócenia

Rys. 1.14. Regulacja uwzględniająca zakłócenie, $\sigma = 0,3$ Rys. 1.15. Regulacja uwzględniająca zakłócenie, $\sigma = 0,5$

Rys. 1.16. Regulacja uwzględniająca zakłócenie, $\sigma = 0,8$

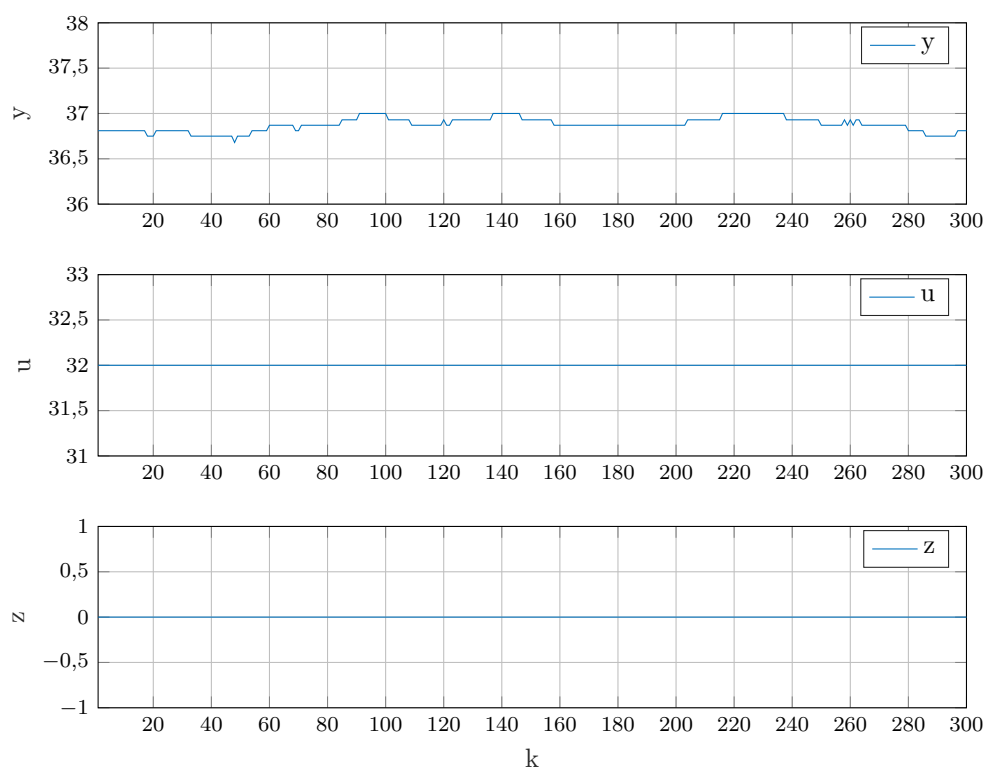
Wnioski:

Im większe zakłócenia tym jakość regulacji jest mniejsza.

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

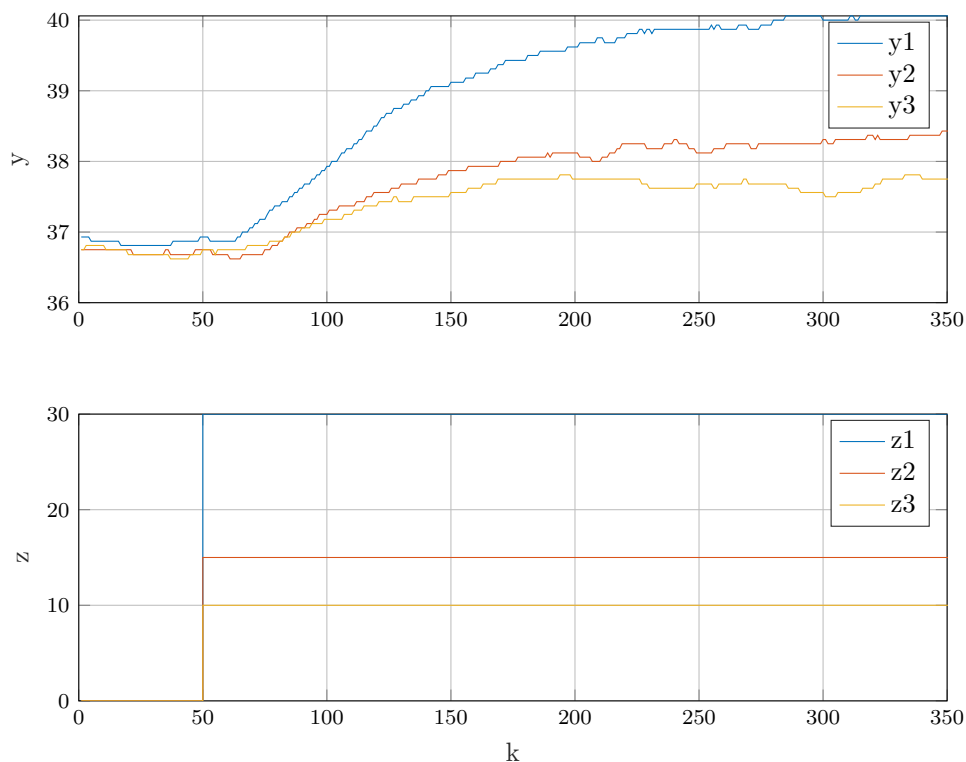
Ustalona wartość wyjścia obiektu wynosi $T1 = 35,4^{\circ}C$.

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

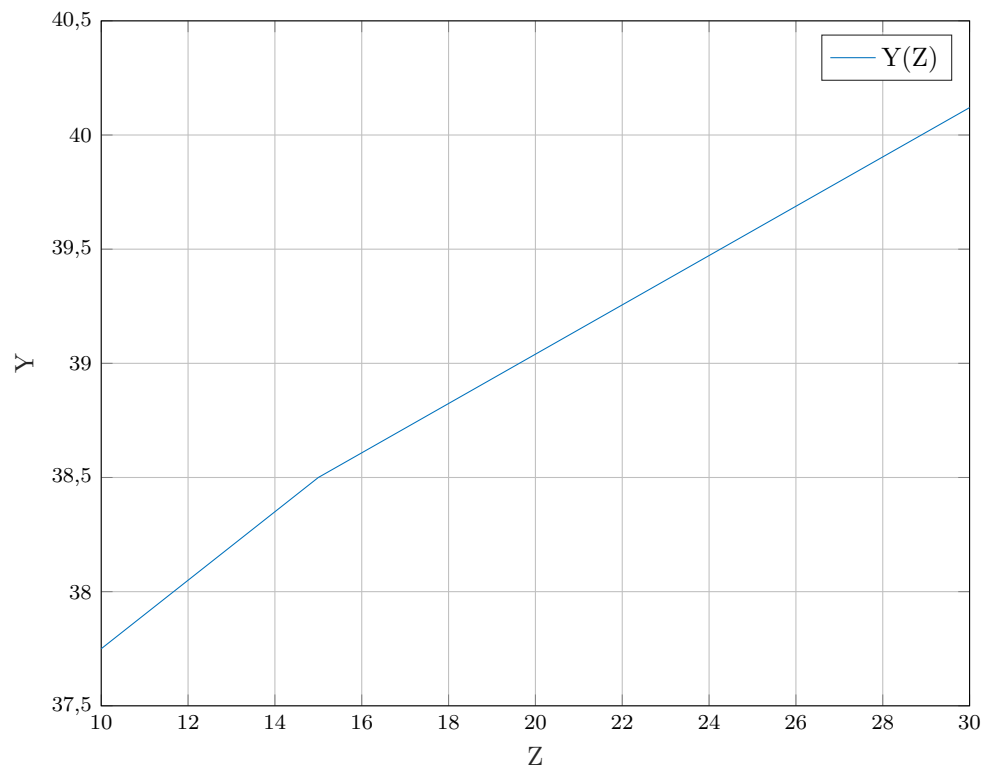
Do uzyskania odpowiedzi skokowych dla tego toru ustawiono sygnał wejściowy na stałą wartość $U=32$, przy ustabilizowanej temperaturze $T_1 = 35,4^{\circ}\text{C}$ przeprowadzone zostały skoki sterowania z $Z = 0$ na 10, 15 oraz 30.



Rys. 2.2. Odpowiedzi skokowe obiektu od zakłócenia

2.2.2. Właściwości statyczne obiektu

Dzięki uzyskanym odpowiedziom skokowym otrzymano charakterystykę statyczną zakłócenia.



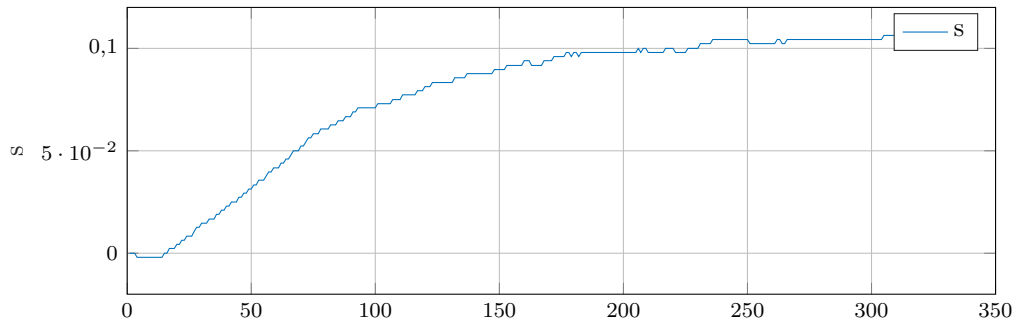
Rys. 2.3. Charakterystyka statyczna wyjścia od zakłócenia

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:

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

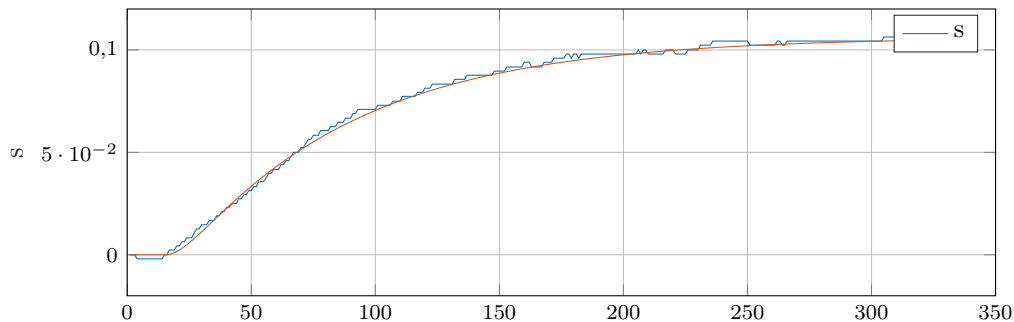
Powyższa transmitancja po przekształceniu do dziedziny czasu dyskretnego i przejściu na postać równania różnicowego:

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

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' );
```

```
% górne 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 początkowe
```

```
kk = 2000; %koniec symulacji
```

```
start = 300; %start symulacji
```

```
start_z = 1000; %start symulacji
```

```
u=ones(1, kk)*39; %sygnał sterujący
```

```
E = 0;
```

```
zakl = 1; %czy są zakłócenia 1-T/0-N
```

```
licz_z = 0; %czy mamy uwzględnić 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 wyjść 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

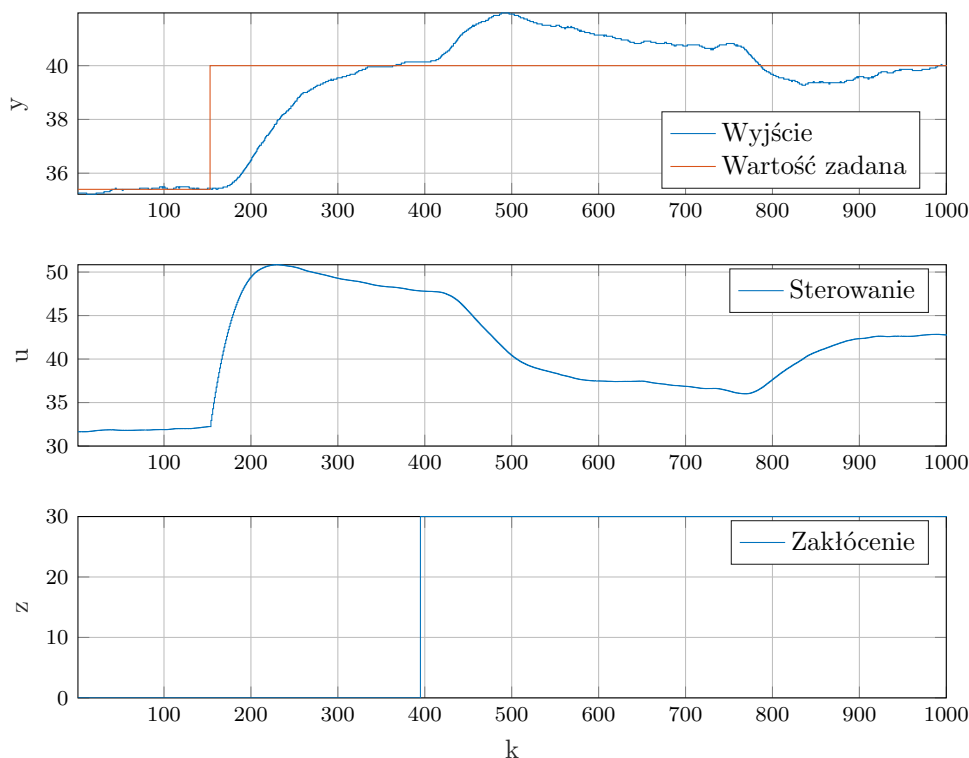
2.5.1. Dobór parametru D_z

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. Re regulacja 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.