

Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska

Projektowanie układów sterowania
(projekt grupowy)

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

Stanisław Borkowski, Piotr Czajkowski, Maciej Grudziąż

Warszawa, 2018

Spis treści

I. Projekt

1. Wstęp	3
2. Sprawdzenie poprawności punktu pracy	4
3. Przebiegi odpowiedzi skokowych	5
3.1. Tory wejście-wyjście oraz wejście-zakłócenie	5
3.2. Charakterystyka statyczna procesu	7
3.3. Określenie wzmocnienia statycznego	7
4. Przebiegi odpowiedzi skokowych użytecznych w algorytmie DMC	9
5. Implementacja algorytmu DMC	10
6. Dobór parametrów regulacji	12
7. Przebiegi symulacji z zakłóceniem	15
7.1. Dla skokowej zmiany zakłócenia	15
7.1.1. Bez uwzględniania zakłóceń	15
7.1.2. Z uwzględnieniem zakłóceń	15
7.2. Dla zakłócenia zmiennego sinusoidalnie	17
7.2.1. Bez uwzględniania zakłóceń	17
7.2.2. Z uwzględnieniem zakłóceń	17
7.3. Wnioski	17

II. Laboratorium

8. Wstęp	21
9. Ustalenie wartości wyjścia w punkcie pracy	22
10. Zebranie odpowiedzi skokowych dla toru wejścia	23
11. Normalizacja i aproksymacja wybranych odpowiedzi skokowych	24
12. Implementacja regulatora DMC	25
13. Przebiegi procesu	26

Część I

Projekt

1. Wstęp

Celem projektu była implementacja, weryfikacja poprawności działania i dobór parametrów algorytmów regulacji jednowymiarowego procesu. Należało również wziąć pod uwagę zakłócenie występujące w układzie. W tym przypadku jest to zakłócenie mierzalne, a jego wartością można sterować za pomocą programu. Należy pamiętać, że w prawdziwych układach regulacji często nie da się go w tak łatwy sposób zmierzyć, a tym bardziej nim sterować. W tym projekcie najważniejsze jest jednak dokładne zrozumienie mechanizmów działających w procesach jednowymiarowych z zakłóceniem, takie uproszczenie jest więc dozwolone.

Praca przebiegała na przygotowanej uprzednio symulacji rzeczywistego obiektu. W ramach projektu przygotowano (w środowisku MATLAB) programy analizujące jego odpowiedzi skokowe oraz implementujące algorytm regulacji DMC.

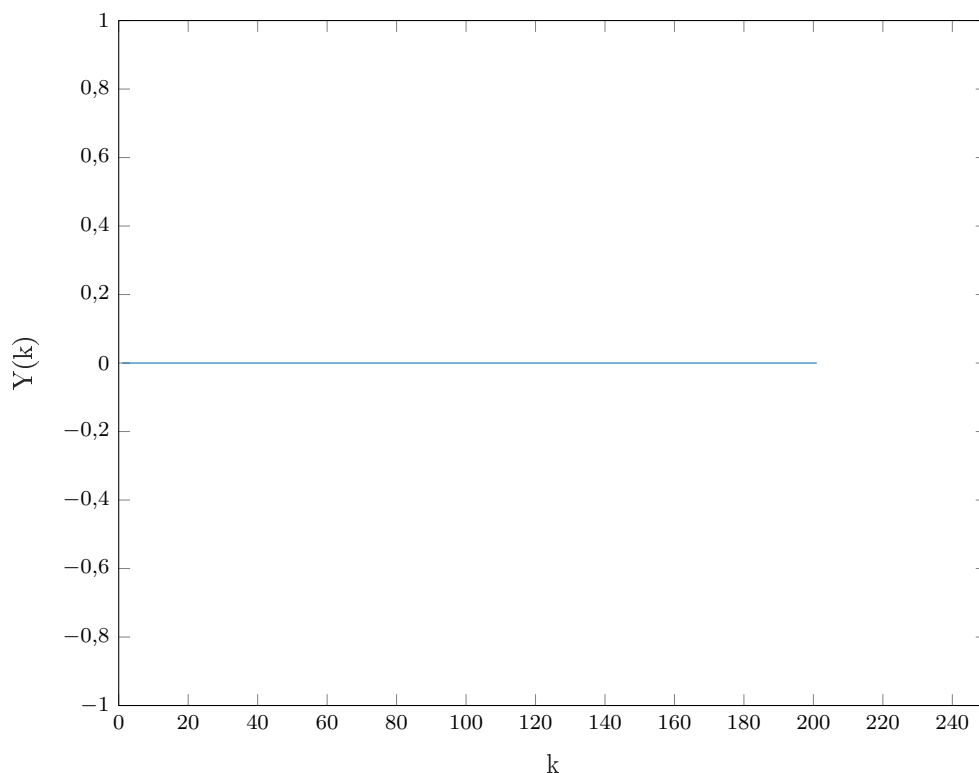
2. Sprawdzenie poprawności punktu pracy

Punktem pracy obiektu nazywa się taki punkt na jego charakterystyce, w którym zachodzi jego działanie. W stanie ustalonym nie następują zmiany w wartościach wyjścia i wejścia.

Podstawową funkcją używaną do badania odpowiedzi skokowych jest $Y = \text{symulacja_obiektu_UppYppZpp}(\text{Upp}$ przyjmująca jako parametry wejściowe wartości skoku sygnału wejściowego oraz sterowania. Na ich podstawie dokonuje ona symulacji sygnału wyjściowego przy skoku od zera do zadanej wartości. Jej szczegółowy opis zamieszczony jest w dalszej sekcji niniejszego sprawozdania. Można jej użyć także w tym przypadku, ustalając wartość skoku sterowania i zakłócenia na zero.

Aby sprawdzić poprawność podanego punktu pracy tj. $u = y = z = 0$ należy przekazać ww. wartości jako parametry wspomnianej funkcji. W rezultacie otrzymano przebieg przedstawiony na Rys. 2.1 Wynik symulacji zgadza się z podanym w skrypcie wynikiem oczekiwanym, można więc uznać powyższą procedurę za udaną, a punkt pracy za poprawny.

Rys. 2.1. Wartość sygnału wyjściowego w punkcie pracy $u = y = z = 0$



3. Przebiegi odpowiedzi skokowych

Wspomniana wcześniej funkcja `Y=symulacja_obiektu_UppYppZpp(Upp,Zpp,time,t)` ma za zadanie symulować zachowanie obiektu w przypadku skoku sterowania (argument `Upp`) lub zakłócenia (argument `Zpp`) od zera do podanej wartości. Zazwyczaj w obliczeniach używa się skoku sterowania następującego w dyskretnych chwilach. Program MATLAB nie umożliwia jednak indeksowania od zera ani liczb ujemnych, w związku z czym funkcja pobudza obiekt skokiem dopiero w chwili $k = 16$, a po wykonaniu całej symulacji obcina wektor wyjścia Y tak, aby wyniki symulacji zgadzały się z powyższym założeniem.

Czas symulacji, tj. ilość dyskretnych chwil, w których liczona jest odpowiedź skokowa ustalone zostało na `time=200`. Taka ilość próbek pozwala zaobserwować ustalenie się procesu (pod koniec symulacji wartości wyjścia są niemal identyczne).

Implementację funkcji przedstawia poniższy listing:

```
function Y=symulacja_obiektu_UppYppZpp(Upp,Zpp,time,t)

    %alokacja pamieci
    U=zeros(time+16,1);
    Y=zeros(time+16,1);
    Z=zeros(time+16,1);

    for k=16:time+16
        U(k)=Upp;
        Z(k)=Zpp;

        %skok wejścia lub zakłócenia w zależności od parametru t
        if (t == 1)
            Y(k)=symulacja_obiektu7y(U(k-4),U(k-5),Z(k-1),...
                Z(k-2),Y(k-1),Y(k-2));
        end
        if (t == 2)
            Y(k)=symulacja_obiektu7y(0,0,Z(k-1),Z(k-2),...
                Y(k-1),Y(k-2));
        end

        %obcięcie pierwszych elementów wektora
        Y=Y(16:time+16,1);
    end
```

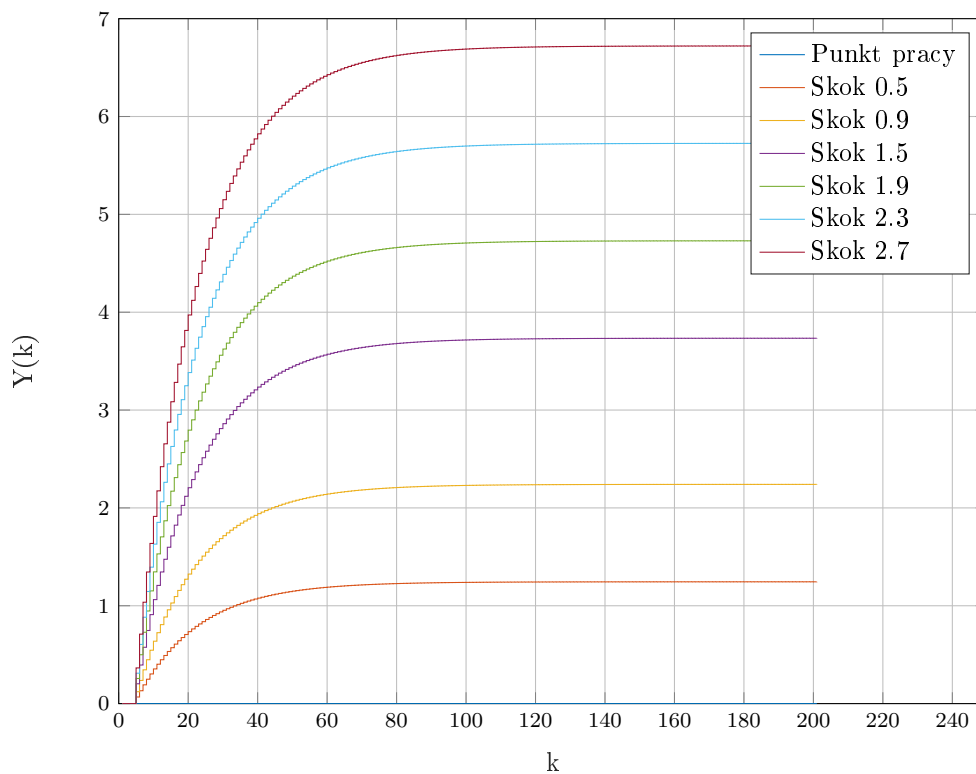
3.1. Tory wejście-wyjście oraz wejście-zakłócenie

Kolejnym krokiem związanym z badaniem symulowanego obiektu jest zbadanie jego odpowiedzi skokowych na różne wartości skoków sterowania i zakłócenia. Zaczynając od punktu pracy i używając funkcji `Y=symulacja_obiektu_UppYppZpp(Upp,Zpp,time,t)` dokonano tej procedury, a wyniki symulacji przedstawiono na Rys. 3.1 i 3.2.

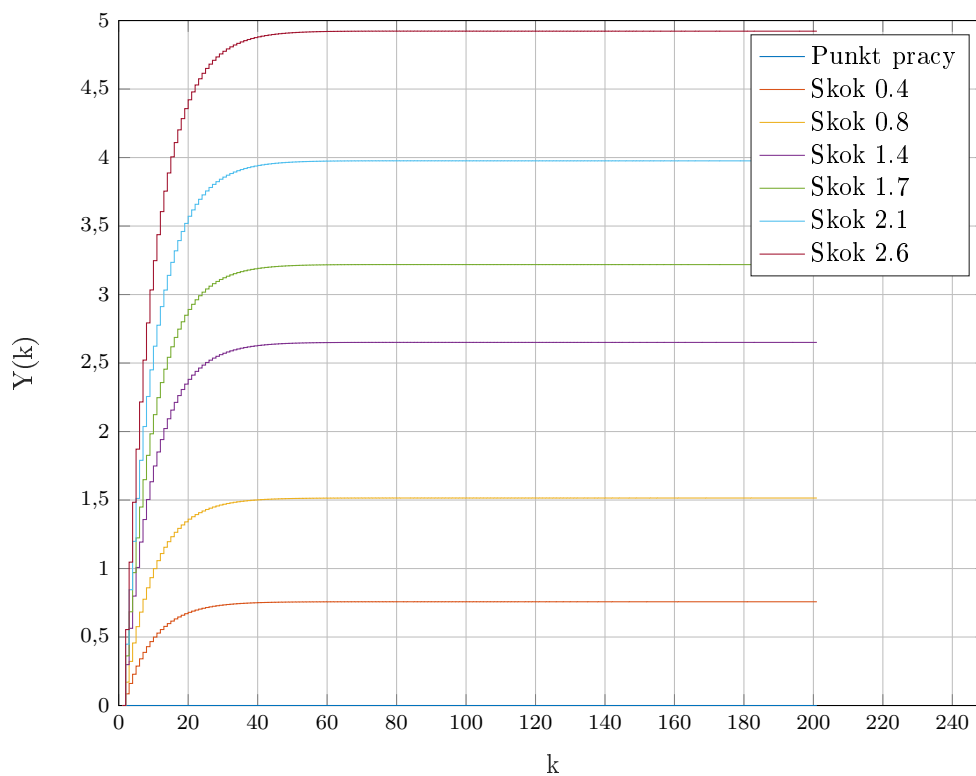
Dokładne działanie obiektu jest nieznane, można jednak stwierdzić, że symulacja została przeprowadzona prawidłowo ze względu na widoczne w obu przypadkach opóźnienie.

Na podstawie tych wykresów można też uznać, że właściwości dynamiczne obiektu są liniowe, tzn. wartości odpowiedzi skokowych w stanie ustalonym są wprost proporcjonalne do skoków sterowania/zakłócenia, które je wywołały.

Rys. 3.1. Charakterystyka dla skoku wartości sterowania



Rys. 3.2. Charakterystyka dla skoku wartości zakłócenia



3.2. Charakterystyka statyczna procesu

Charakterystyka statyczna obiektu to zależność między sygnałem wyjściowym a sygnałem wejściowym (w tym przypadku sygnałem sterującym lub zakłóceniem) w stanie ustalonym. Do jej wyznaczenia użyte zostały odpowiedzi skokowe wyliczone w punkcie poprzednim. Ponieważ w każdym z przypadków w dyskretnej chwili $k = 200$ obiekt jest już w stanie ustalonym, najwygodniejszym sposobem było użycie ostatniej wartości w wektorze odpowiedzi dla każdego skoku. Na tej podstawie sporządzono wykres zależności $y(u, z)$ przedstawiony na Rys. 3.3. Dla zwiększenia przejrzystości i ułatwienia analizy przedstawiono również widok z dwóch perspektyw na ten wykres - $Y(U)$ i $Y(Z)$ - rysunki odpowiednio 3.4 i 3.5.

Na pierwszy rzut oka widać, że charakterystyki są liniowe, tzn. wartości $Y(U)$ i $Y(Z)$ ustalają się w przybliżeniu na linii prostej, za takie też będą traktowane w dalszych rozważaniach.

3.3. Określenie wzmocnienia statycznego

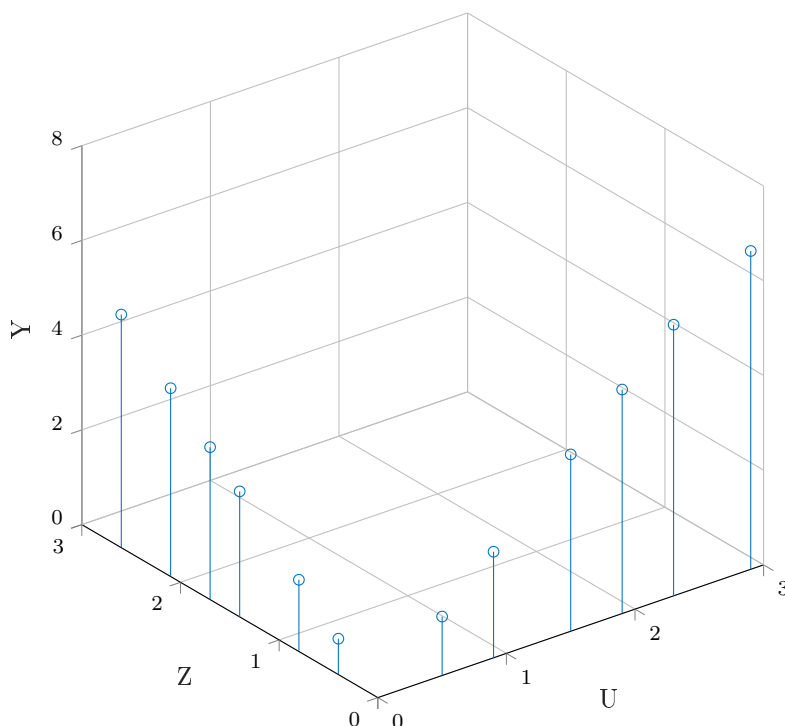
Na podstawie tej charakterystyki można obliczyć wzmocnienia statyczne obu torów procesu. Definiuje się je jako

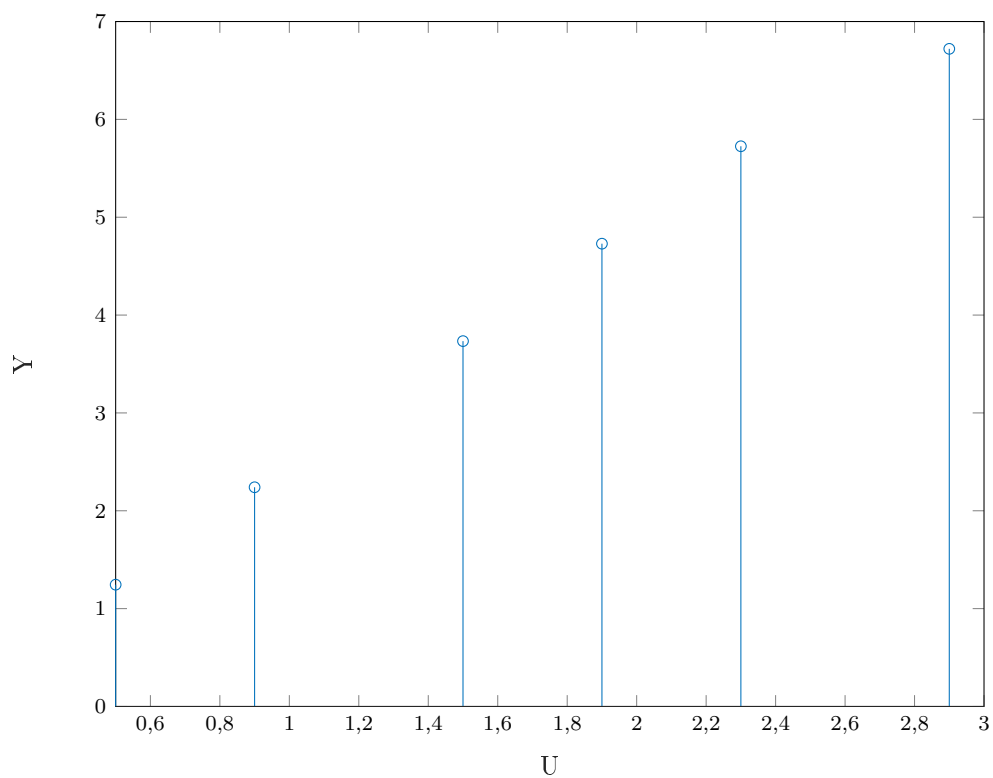
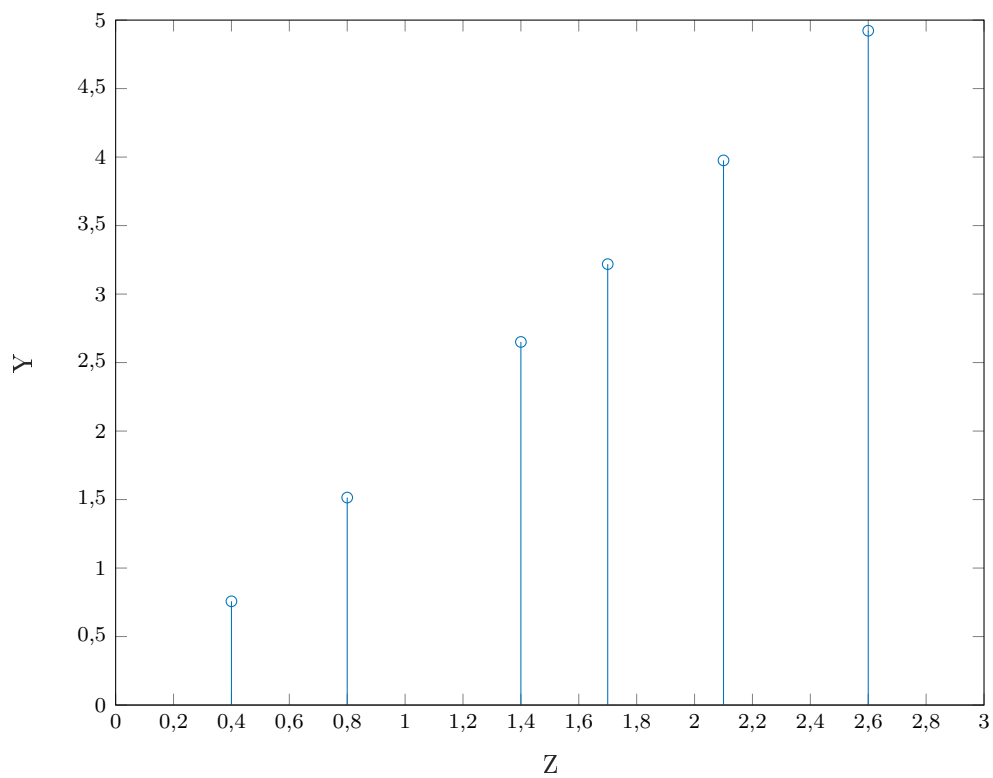
$$K_U = \frac{\Delta Y}{\Delta U}, K_Z = \frac{\Delta Y}{\Delta Z} \quad (3.1)$$

Pary parametrów ΔY i ΔU oraz ΔY i ΔZ są stałe dla dowolnie wybranych punktów na wykresach $Y(U)$ i $Y(Z)$, ponieważ charakterystyka jest liniowa. Należy więc wybrać dowolne punkty (U_i, Y_i) i (U_j, Y_j) oraz (Z_i, Y_i) i (Z_j, Y_j) i na ich podstawie obliczyć wzmocnienia statyczne K_U i K_Z . Wyniki:

$$K_U = 2,49, K_Z = 1,89 \quad (3.2)$$

Rys. 3.3. Charakterystyka statyczna

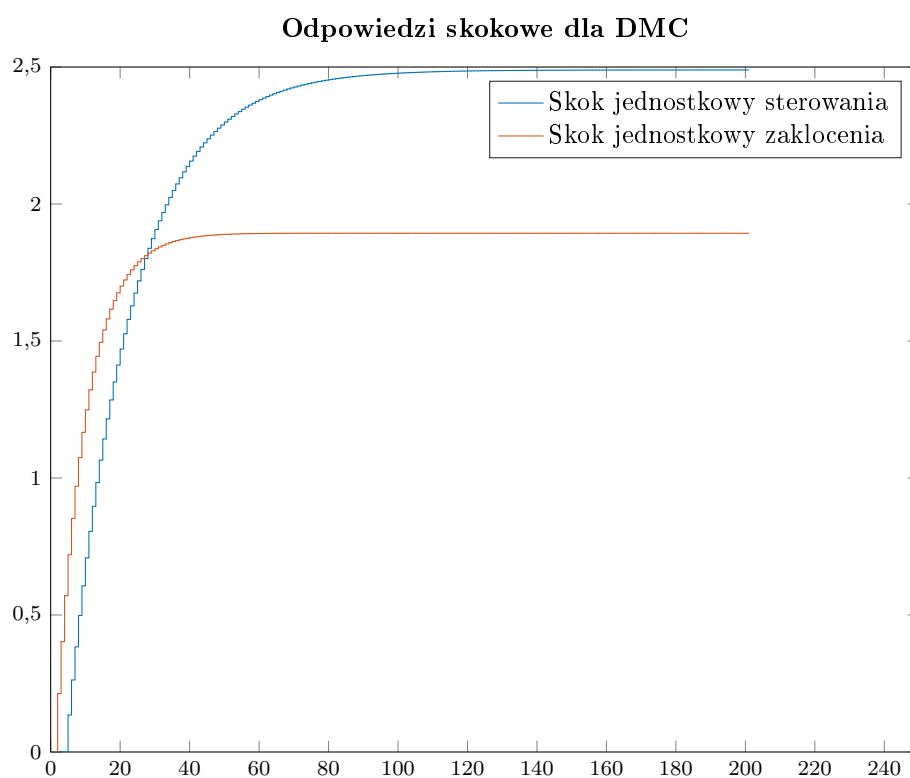


Rys. 3.4. Charakterystyka statyczna - $Y(U)$ Rys. 3.5. Charakterystyka statyczna - $Y(Z)$ 

4. Przebiegi odpowiedzi skokowych użytecznych w algorytmie DMC

Aby otrzymać odpowiedź skokową wykorzystywaną w algorytmie DMC, należy postępować analogicznie - należy pobudzić obiekt skokiem jednostkowym, gdzie od chwili zerowej sygnał sterujący ma wartość 1, a w przeszłości jest zerowy. Wynik tej odpowiedzi skokowej jest przedstawiony na Rys. 4.1.

Rys. 4.1.



5. Implementacja algorytmu DMC

W języku Matlab zaimplementowano program do symulacji działania algorytmu DMC, pozwalającym na prace regulatora w trybie z uwzględnieniem zakłócenia mierzalnego, jak i bez jego uwzględniania.

Algorytm DMC zaimplementowano w wersji oszczędnej. Jako prawo sterowania stosowano wzór 5.1 w przypadku, gdy nie uwzględniano zakłócenia mierzalnego.

$$\Delta u(k) = K_e e(k) - \bar{K}_1 M^p \Delta u^p(k) \quad (5.1)$$

W przypadku uwzględniania zakłócenia mierzalnego jako prawo sterowania stosowano wzór 5.2

$$\Delta u(k) = K_e e(k) - \bar{K}_1 M^p \Delta u^p(k) - \bar{K}_1 M_z^p \Delta z^p(k) \quad (5.2)$$

Parametry K_e , \bar{K}_1 , M^p oraz M_z^p są stałe dla każdego przebiegu i wyznaczone na podstawie uzyskanych odpowiedzi skokowych \mathbf{s} , \mathbf{s}_z i zadanych parametrów D , N , N_u , λ . Uzyskiwanie parametrów prezentuje poniższy listing:

```
M = zeros(N,Nu);
M(:,1) = s(1:N);

for i = 2:Nu
M(i:N,i)=M(1:N-i+1,1);
end;

MP = zeros(N,D-1);
for i = 1:D-1
MP(:,i) = s((i+1):(i+N))-s(i);
end;

MPz = zeros(N,Dz);
for i = 1:Dz
MPz(:,i) = sz((i+1):(i+N))-sz(i);
end;

K = (M' * M + lambda * eye(Nu))\M';
K1 = K(1,:);
Ku = K1 * MP;
Ke = sum(K1);

DUP = zeros(D-1,1);
DZP = zeros(Dz,1);

Y(1:t)=Ypp;
U(1:t)=Upp;
```

```
y(1:t) = Y(1:t) - Ypp;
u(1:t) = U(1:t) - Upp;
```

Dla wyznaczonych parametrów sterowania przeprowadzano symulację procesu dla różnych przebiegów zakłócenia $Z(k)$ oraz stałej trajektorii wartości zadanej $Y^{zad}(k)$ w postaci skoku z wartości 0 do wartości 1 w chwili odpowiadającej 21 próbie sygnału.

```
for k=t:kk

Y(k) = symulacja_obiektu7y(U(k-4),U(k-5),Z(k-1),Z(k-2),Y(k-1),Y(k-2));
y(k) = Y(k) - Ypp;

DUP = [u(k-1) - u(k-2) ; DUP(1:D-2)];
DZP = [z(k-1) - z(k-2) ; DZP(1:Dz-1)];

e = yzad(k) - y(k);

du = Ke*e - Ku*DUP;

if distortion ~= 0
du = Ke*e - K1*MP*DUP - K1*MPz*DZP;
end;

u(k) = du(1)+u(k-1);
U(k) = u(k)+Upp;
%U(k)=1;

end;

e = (yzad-y)*(yzad-y)';
E = (Yzad-Y)*(Yzad-Y)';
```

6. Dobór parametrów regulacji

W celu dobrania parametrów regulacji algorytmu DMC wykonano serię symulacji, w których przebieg wartości zadanej $Y^{zad}(k)$ miał postać skoku z wartości 0 do wartości 1 w chwili odpowiadającej 21 próbkę w zakresie 400 próbek procesu. Sygnał zakłócenia $Z(k)$ miał wartość 0 dla każdej chwili k .

Za wartości początkowe parametrów regulatora przyjęto $D = N = 201$, $N_u = 200$, $\lambda = 1,0$. Dla kolejnych symulacji modyfikowano wartość parametru N_u obserwując zmianę wskaźnika jakości e oraz przebieg sygnału. Wykonano sześć symulacji dla parametru $N_u = 2, 3, 4, 5, 6, 200$. Wyniki przebiegów zestawiono ze sobą (Rys. 6.1), wartość wskaźnika jakości e zebrano w tabeli (Tab. 6.1).

Wyraźnie widać, że najlepszy rezultat pod kątem minimalizacji wskaźnika jakości daje $N_u = 4$. Przebieg procesu dla tej wartości mimo iż nie jest całkowicie gładki, nie wpada w oscylacje i może być uznany za zadawalający.

Po wyznaczeniu parametru $N_u = 4$ wykonano kolejne symulacji tym razem manipulując wartością parametru λ . Wyniki przebiegów zestawiono ze sobą (Rys. 6.2), a wartość wskaźnika jakości e zebrano w tabeli (Tab. 6.2).

Pomimo iż wartość wskaźnika jakości jest najmniejsza dla $\lambda = 0,25$ proces już przy wartości $\lambda = 0,50$ wpada w oscylacje. W związku z tym zdecydowano na zachowanie parametru $\lambda = 1$.

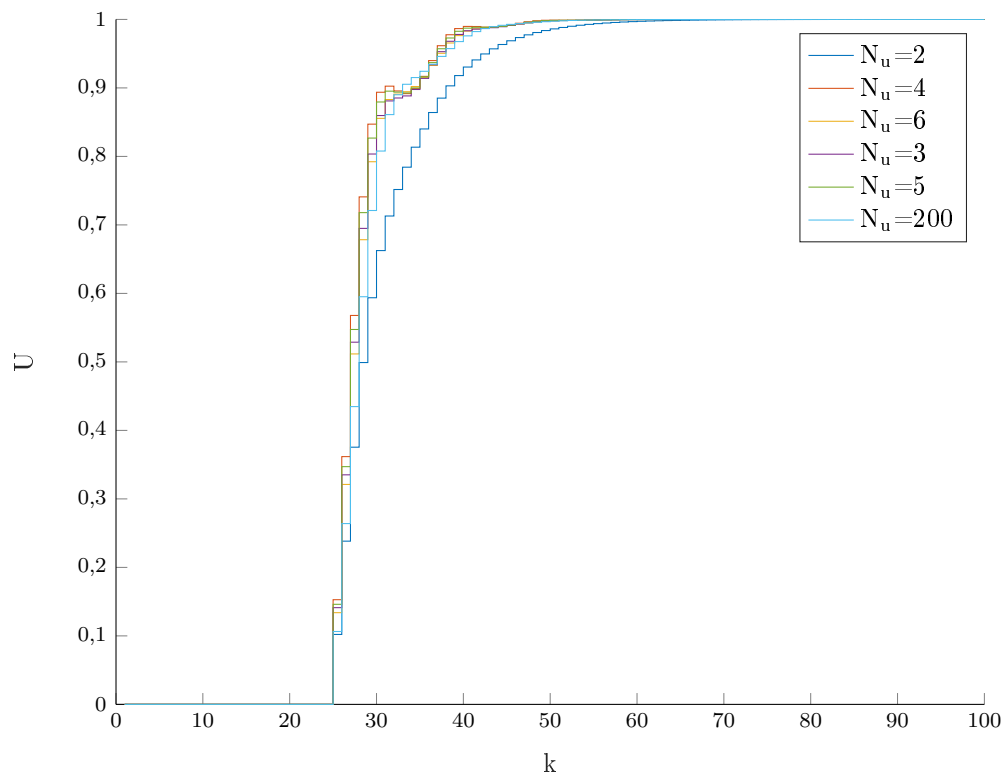
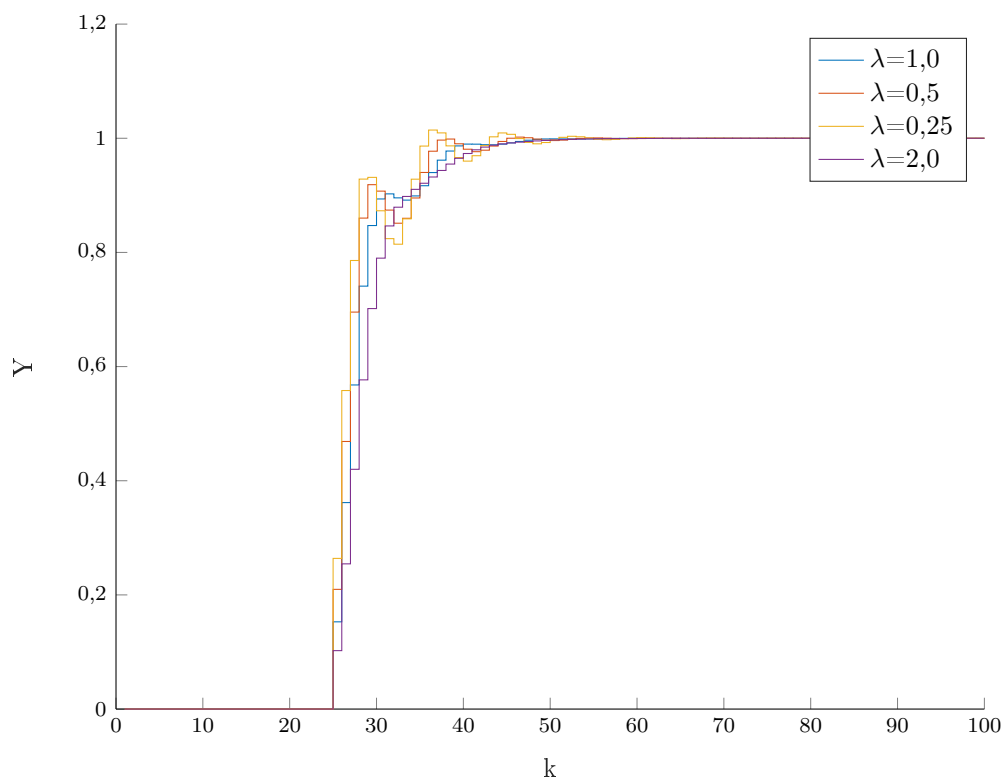
Dla dobranych parametrów $D = N = 201$, $N_u = 4$, $\lambda = 1$ wykonano wizualizację przebiegu (Rys. 6.4)

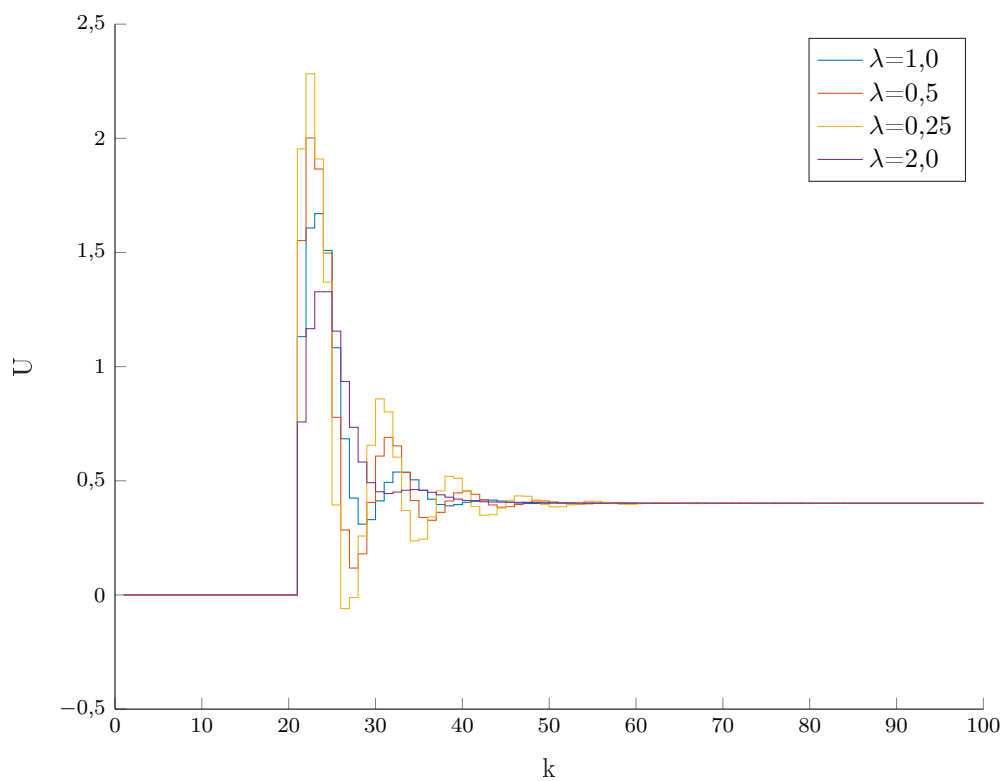
Tab. 6.1. Wartość wskaźnika e jakości dla różnych wartości parametru N_u

Horyzont sterowania N_u	Błąd średniokwadratowy e
200	6,0032
2	6,6227
3	5,6199
4	5,4694
5	5,5429
6	5,6802

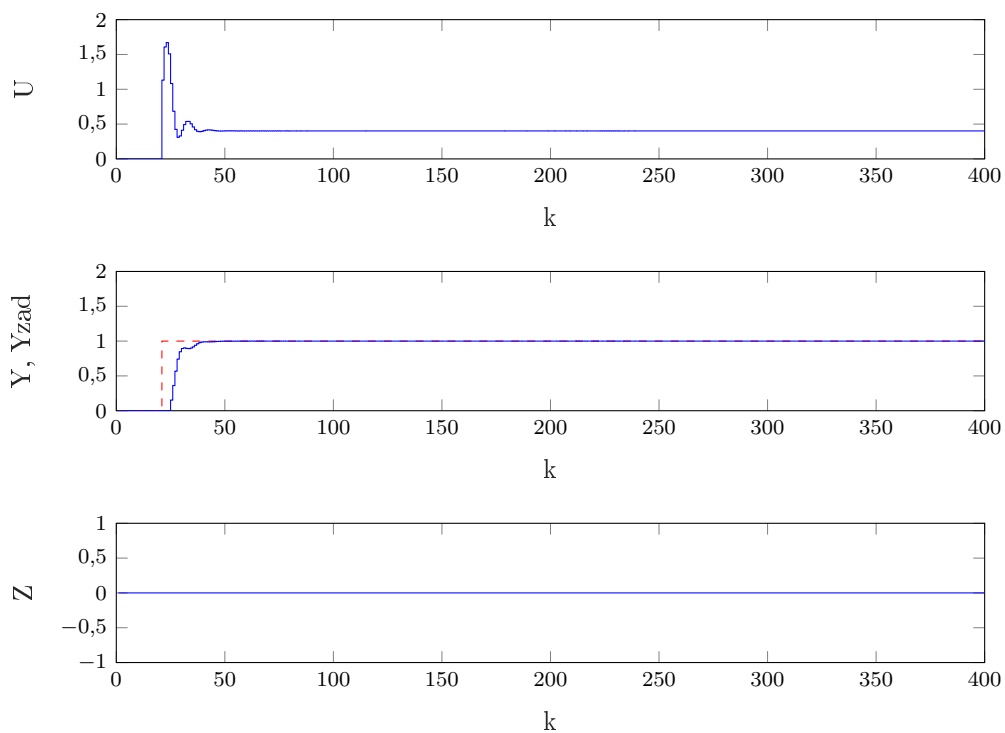
Tab. 6.2. Wartość wskaźnika e jakości dla różnych wartości parametru λ

Współczynnik λ	Błąd średniokwadratowy e
1,00	5,4694
0,50	5,1086
0,25	4,9042
2,00	6,0860

Rys. 6.1. Przebieg symulacji algorytmu DMC dla różnych wartości parametru N_u Rys. 6.2. Przebieg symulacji algorytmu DMC dla różnych wartości parametru λ 

Rys. 6.3. Przebieg sterowania algorytmu DMC dla różnych wartości parametru λ 

Rys. 6.4. Przebieg symulacji algorytmu DMC dla dobranych paramterów



7. Przebiegi symulacji z zakłóceniem

Dla dobranych parametrów algorytmu DMC wykonano symulacje dla zmiennego toru zakłócenia mierzalnego. W wyniku symulacji przeprowadzonych podczas doboru parametrów ustalono, że w chwili $k = 100$ proces ma osiągniętą wartość wyjściową $y(k)$ równą wartości zadanej. Dlatego wszelkie zmiany wartości zakłócenia następowały w tej właśnie chwili. Za wartość parametru D_z przyjęto $D_z = 201$, będące długością uzyskanej odpowiedzi skokowej toru zakłócenie-wyście.

7.1. Dla skokowej zmiany zakłócenia

Zasymulowano przebiegi procesu dla zmiany wartości zakłócenia mierzalnego z wartości 0 na wartość 1 w chwili odpowiadającej 100-nej próbce symulacji. Symulacje uruchomiono w w dwóch trybach pracy algorytmu DMC.

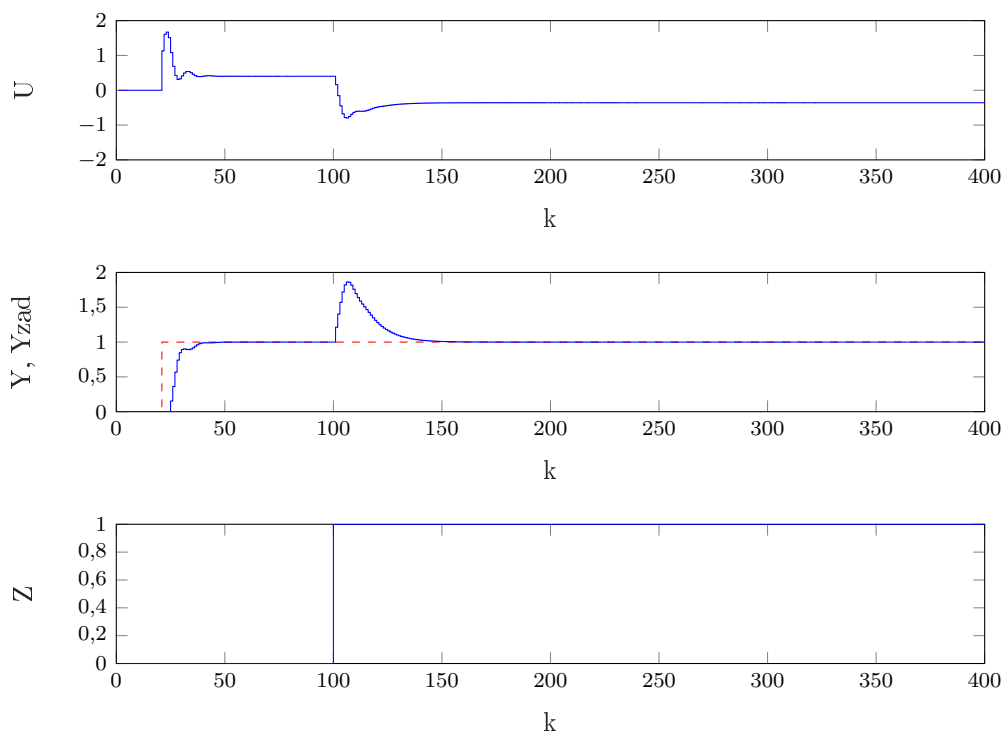
7.1.1. Bez uwzględniania zakłóceń

W trybie pracy algorytmu bez uwzględniania zakłóceń otrzymano wskaźnik jakości $e = 12,8721$. Przebieg przedstawiono na rysunku 7.1

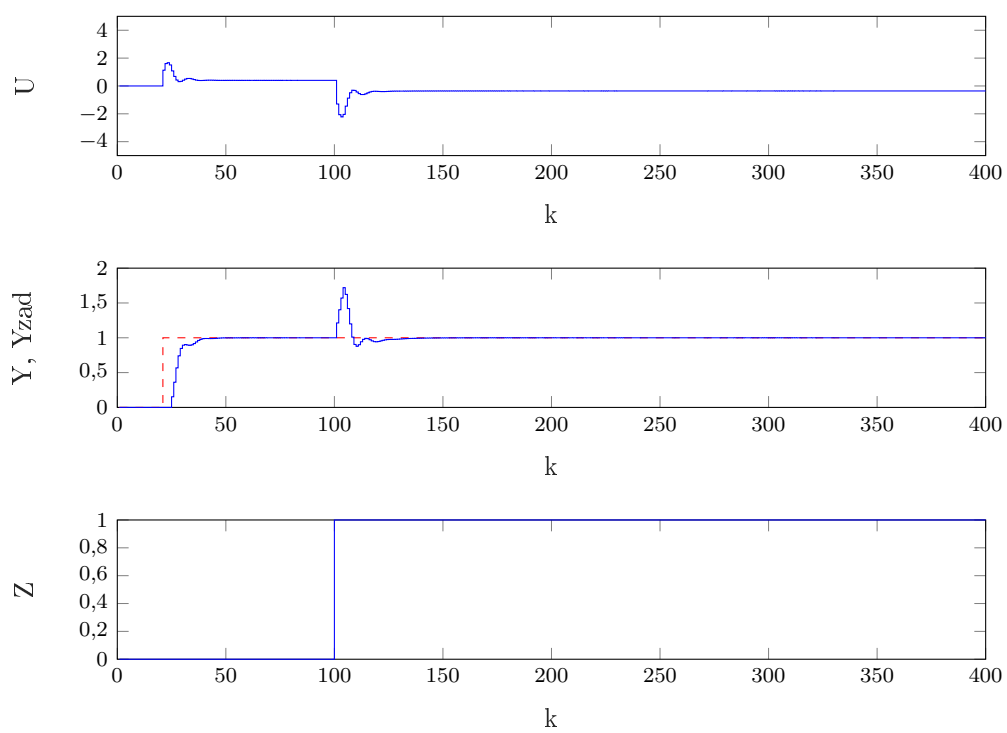
7.1.2. Z uwzględnieniem zakłóceń

W trybie pracy algorytmu z uwzględnieniem zakłóceń otrzymano wskaźnik jakości $e = 7,1838$. Przebieg przedstawiono na rysunku 7.2

Rys. 7.1. Przebieg symulacji algorytmu DMC dla skoku zakłócenia bez uwzględniania zakłóceń



Rys. 7.2. Przebieg symulacji algorytmu DMC dla skoku zakłócenia z uwzględnianiem zakłóceń



7.2. Dla zakłócenia zmiennego sinusoidalnie

Zasymulowano przebiegi procesu dla zmiany wartości zakłócenia mierzalnego w chwili odpowiadającej 100-nej próbkę symulacji na sygnał sinusoidalny o amplitudzie 0,2. Symulacje uruchomiono w w dwóch trybach pracy algorytmu DMC.

7.2.1. Bez uwzględniania zakłóceń

W trybie pracy algorytmu bez uwzględniania zakłóceń otrzymano wskaźnik jakości $e = 12,3087$. Przebieg przedstawiono na rysunku 7.3

7.2.2. Z uwzględnieniem zakłóceń

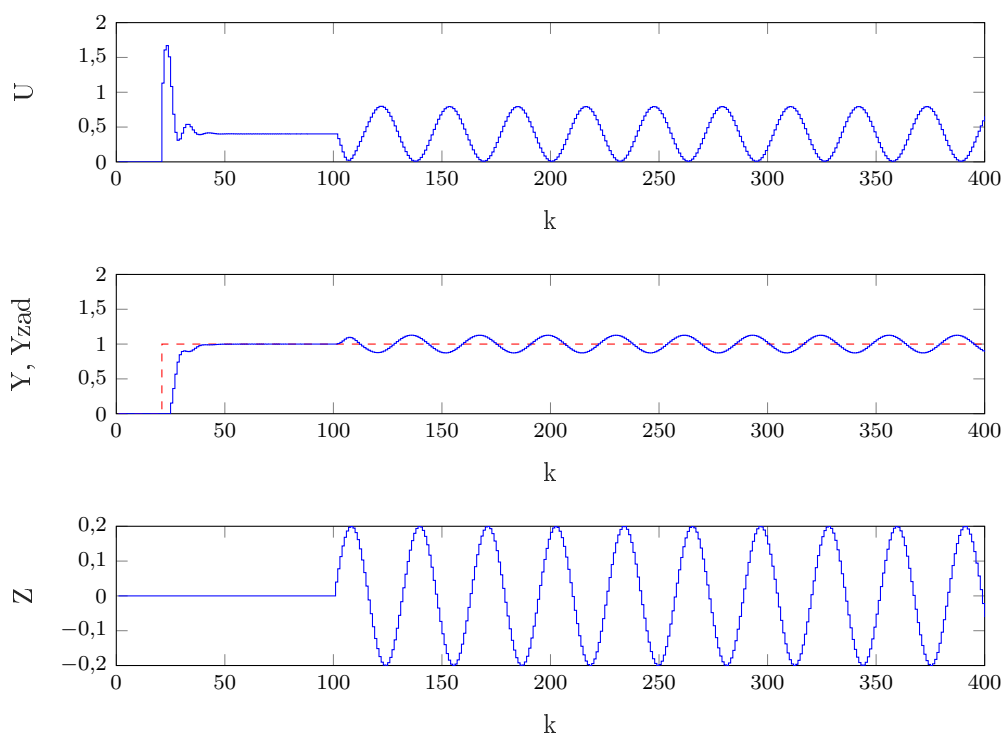
W trybie pracy algorytmu z uwzględnianiem zakłóceń otrzymano wskaźnik jakości $e = 7,8234$. Przebieg przedstawiono na rysunku 7.4

Ponieważ różnica pomiędzy rysunkiem 7.3, a rysunkiem 7.2 nie jest zbyt wyraźna dla zadanej skali wykonano zestawienie wyjścia dla obydwu przebiegów (Rys. 7.5)

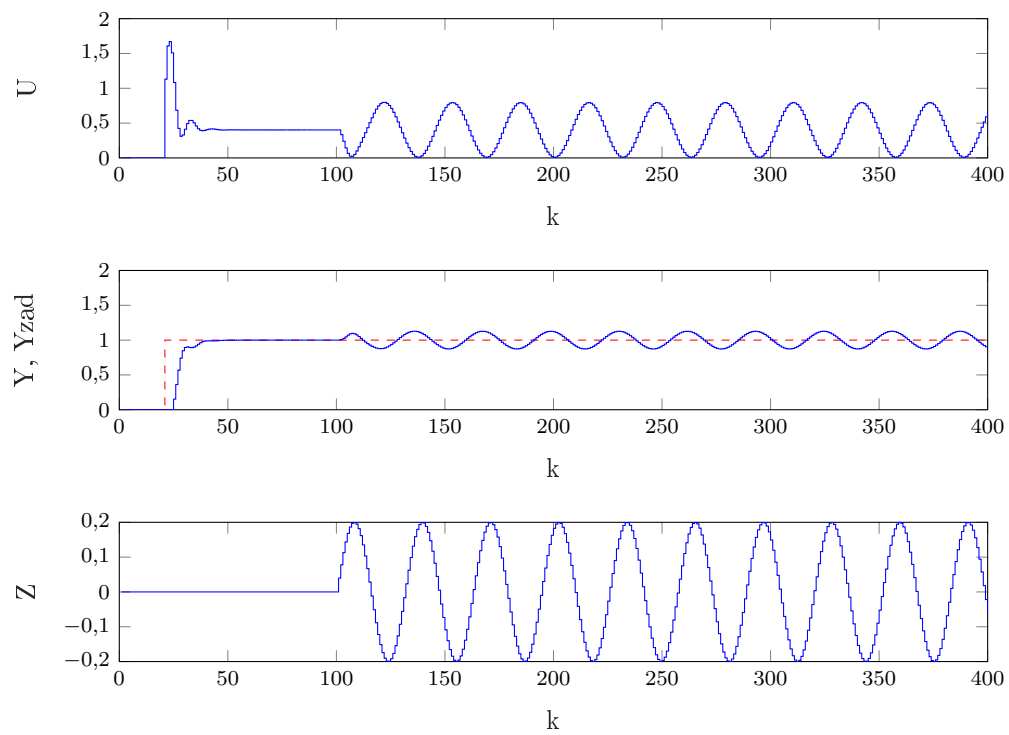
7.3. Wnioski

Zgodnie z oczekiwaniami uwzględnianie zakłócenia mierzalnego w sterowaniu z wykorzystaniem algorytmu DMC poprawia rezultat zmniejszając błąd oraz przyspieszając stabilizację procesu po wystąpieniu zakłócenia. Można zaobserwować że uwzględnienie zakłócenia działa lepiej w przypadku gdy zakłócenie ma charakter stały niż sinusoidalnie zmienny

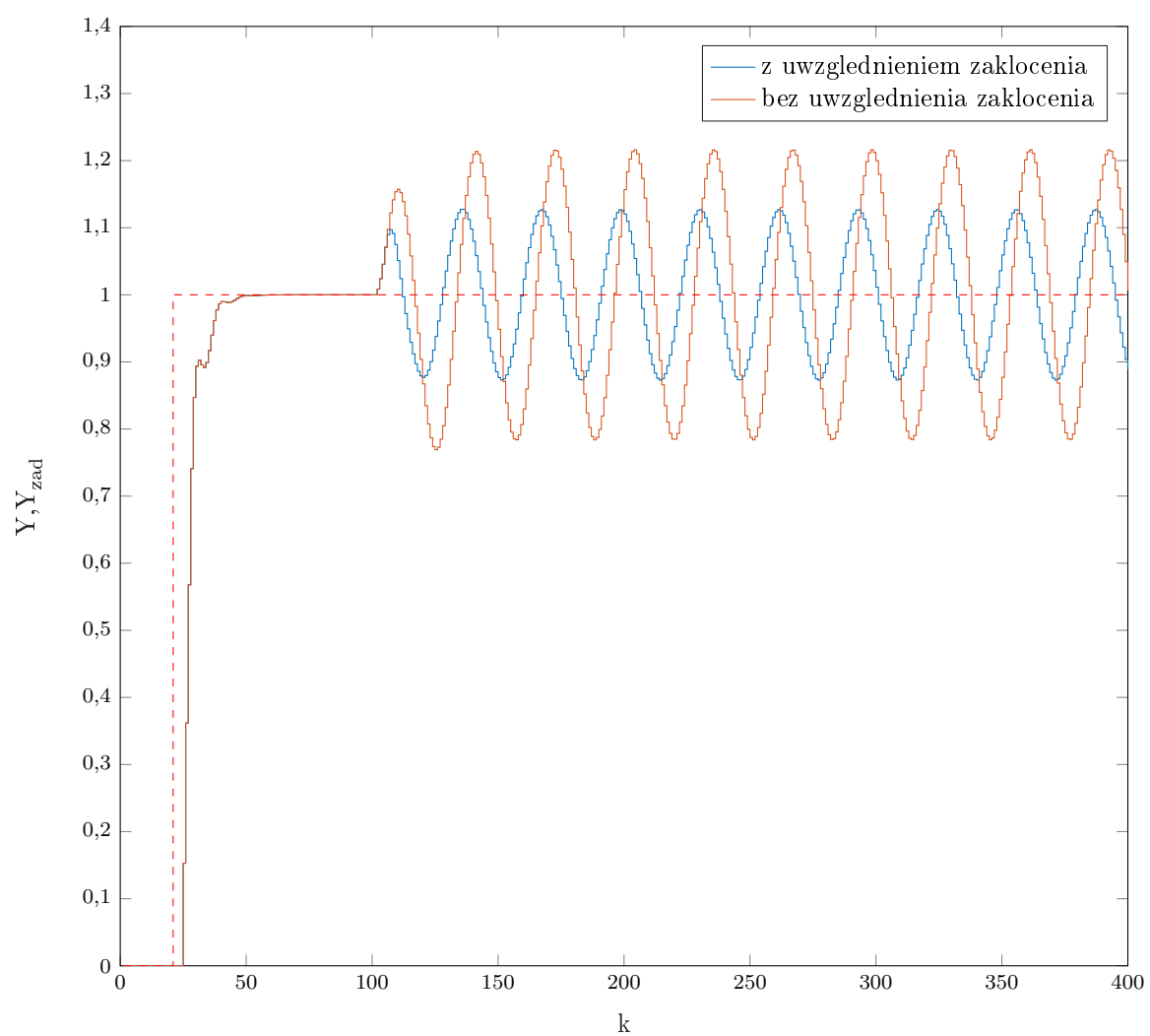
Rys. 7.3. Przebieg symulacji algorytmu DMC dla zakłócenia zmiennego sinusoidalnie bez uwzględniania zakłóceń



Rys. 7.4. Przebieg symulacji algorytmu DMC dla zakłócenia zmiennego sinusoidalnie z uwzględnianiem zakłóceń



Rys. 7.5. Przebieg wyjść symulacji algorytmu DMC dla zakłócenia zmiennego sinusoidalnie



Część II

Laboratorium

8. Wstęp

Po pomyślnej implementacji algorytmu DMC dla obiektu symulowanego zajęliśmy się rzeczywistym obiektem - stanowiskiem grzejąco-chłodzącym. Wykonywaliśmy na nim podobne działania, jakimi traktowaliśmy obiekt symulowany.

9. Ustalenie wartości wyjścia w punkcie pracy

10. Zebranie odpowiedzi skokowych dla toru wejścia

11. Normalizacja i aproksymacja wybranych odpowiedzi skokowych

12. Implementacja regulatora DMC

13. Przebiegi procesu