Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska

Projektowanie układów sterowania (projekt grupowy)

Sprawozdanie z ćwiczenia laboratoryjnego nr 1

Mateusz Koroś, Ksawery Pasikowski, Mateusz Morusiewicz

Warszawa, 2017

Spis treści

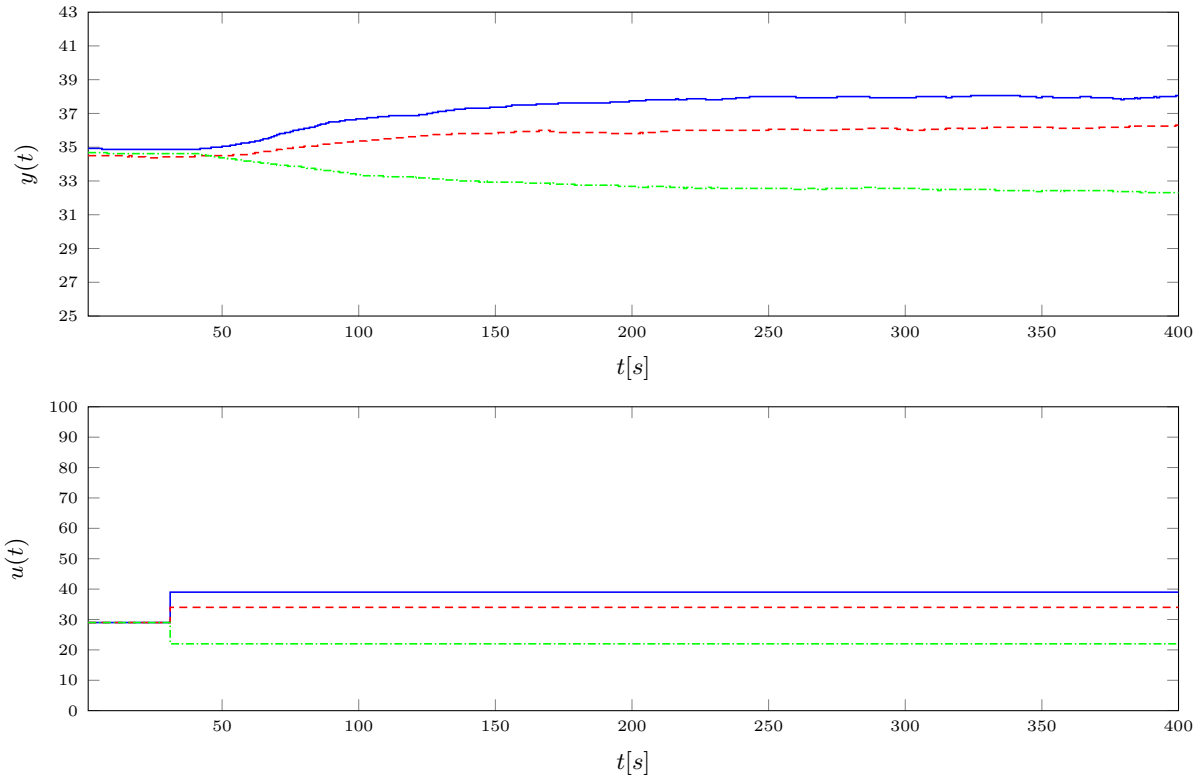
1. Zad. 1

Możliwość sterowania i pomiaru w komunikacji ze stanowiskiem została sprawdzona poprzez funkcję readMeasurements( ) oraz sendControls( ). Sygnały sterujące, które były obsługiwane to moc na grzałce G1 oraz moc wiatraka $W1$, natomiast mierzona była temperatura $T1$ w otoczeniu grzałki $G1$. W punkcie pracy, tzn. dla $(W1, G1) = (50, 29)$ pomiar temperatury wyniósł $35°C$.

2. Zad. 2

Odpowiedzi skokowe dla trzech różnych zmian sygnału sterującego $G1$ zostały przedstawione na wykresie 2.1 Właściwości statyczne obiektu można określić jako liniowe, gdyż zmiana sygnału sterującego powoduje liniową zmianę sygnału wyjściowego. Wzmocnienie statyczne procesu zostało obliczone ze wzoru:

$$K_{st} = \frac{\Delta y}{\Delta u} = 0{,}3$$



Rys. 2.1. Wzbudzenia

3. Zad. 3

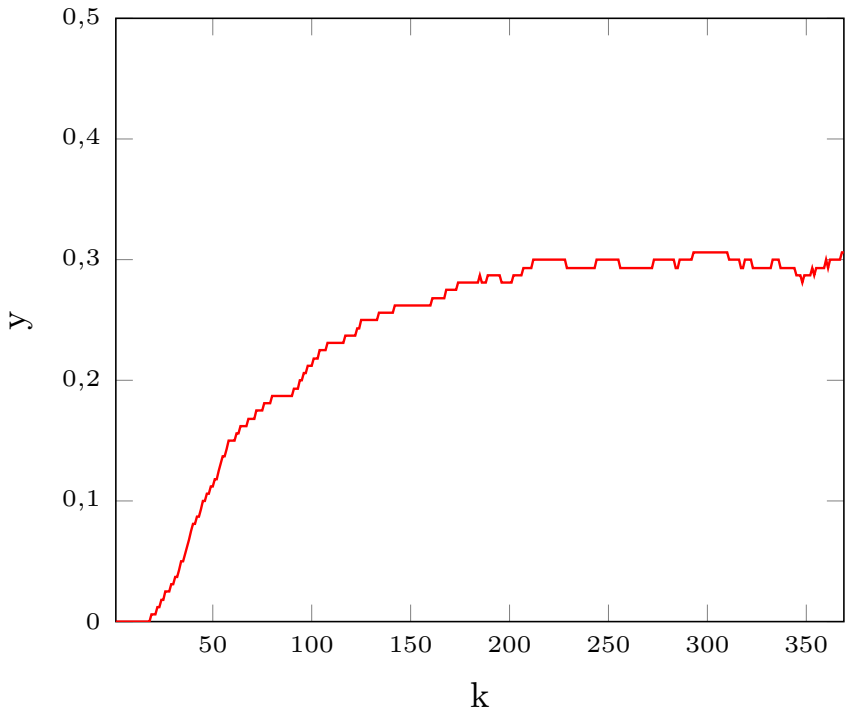
Najlepiej nadającą się odpowiedzią skokową była ta dla skoku sterowania $\Delta u = 10$. Odpowiedź ta została przekształcona do postaci wykorzystywanej w algorytmie DMC w następujący sposób:

```
S = (S - 35) / 10;
S = S(32:end);
```

Odpowiedź ta została przedstawiona na wykresie 3.1

Następnie została wykonana aproksymacja odpowiedzi skokowej, do której został użyty człon inercyjny drugiego rzędu z opóźnieniem, opisany transmitancją:

$$G(z) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-T_d}$$



Rys. 3.1. Odpowiedź skokowa

Gdzie:

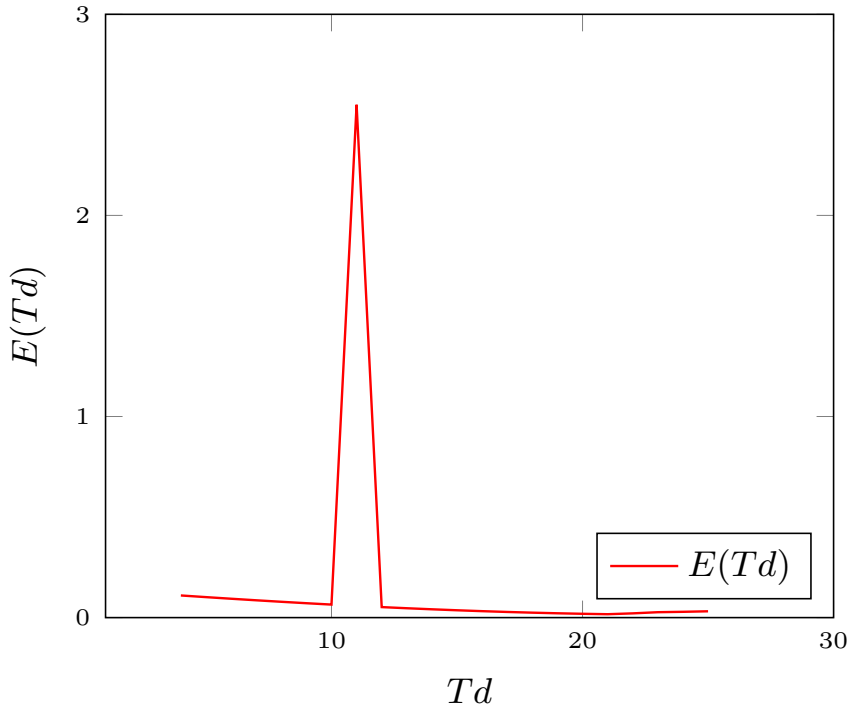
$$\begin{aligned} a_1 &= -\alpha_1 - \alpha_2 \\ a_2 &= \alpha_1 \alpha_2 \\ \alpha_1 &= e^{-\frac{1}{T_1}} \\ \alpha_2 &= e^{-\frac{1}{T_2}} \\ b_1 &= \frac{K}{T_1 - T_2}[T_1(1-\alpha_1) - T_2(1-\alpha_2)] \\ b_2 &= \frac{K}{T_1 - T_2}[\alpha_1 T_2(1-\alpha_2) - \alpha_2 T_1(1-\alpha_1)] \end{aligned} \tag{3.1}$$

Po przekształceniu powyższego równania otrzymujemy równanie różnicowe postaci:

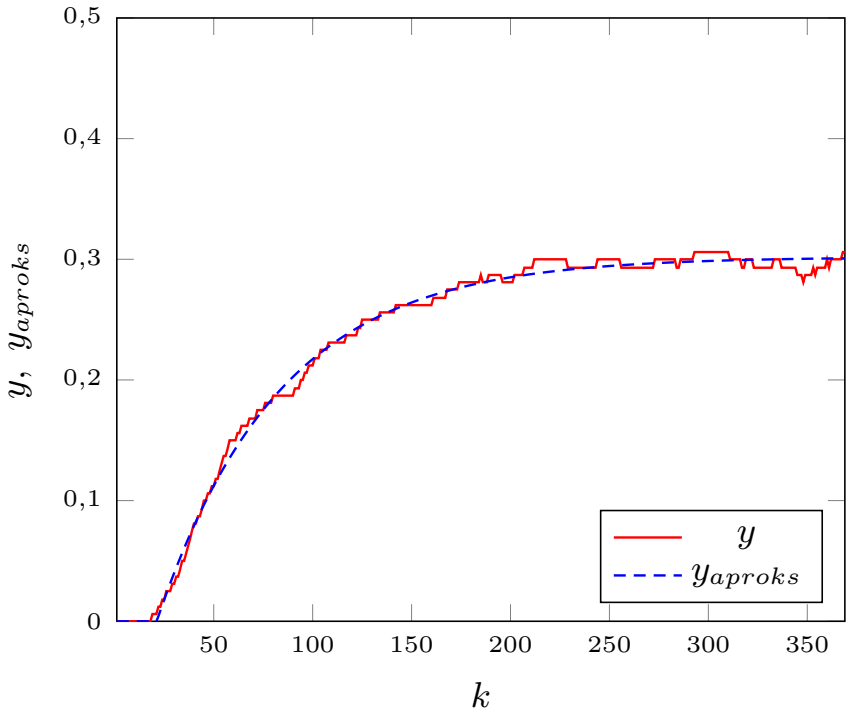
$$y(k) = b_1 u(k - T_D - 1) + b_2 u(k - T_d - 2) - a_1 y(k-1) - a_2 y(k-2)$$

W celu doboru parametrów modelu została użyta funkcja wewnętrzna środowiska MATLAB: fmincon(). Parametry modelu, zwrócone przez ową funkcję to $T1 = 0{,}133\,014\,140\,832\,415 \quad T2 = 69{,}349\,053\,332\,973\,114 \quad K = 0{,}291\,749\,985\,778\,358$. Parametry te zostały dobrane w taki sposób, aby błąd średniokwadratowy między odpowiedzią aproksymowaną, a tą rzeczywistą był jak najmniejszy. Wartość błędu w zależności od parametru T_d widać na rysunku 3.2. Dla powyższych parametrów wyniósł on 0,016 871 238 030 787 4 Porównanie odpowiedzi skokowej oryginalnej oraz wersji aproksymowanej widać na rysunku 3.3

Jak widać na wykresie, funkcja aproksymująca jest bardzo dobrym przybliżeniem oryginalnego przebiegu. Sumaryczny błąd jest niewielki.



Rys. 3.2. Błąd średniokwadratowy w zależności od T_d



Rys. 3.3. Odpowiedź skokowa oryginalna i aproksymowana

4. Zad. 4

4.1. PID

```
addpath('F:\SerialCommunication');
initSerialControl COM3

Upp = 29;
Ypp = 35;
Kk = 800;
U_min = 0;
U_max = 100;

% nastawy regulatora PID
% Kp = 3 ;
% Ti = 10 ;
% Td = 3.2 ;
Kp = 3;%5.94 ;
Ti = 40;%5.64 ;
Td = 5;%3.16 ;
Tp = 1;

r2 = (Kp * Td) / Tp ;
r1 = Kp * ( (Tp/(2*Ti)) - 2*(Td/Tp) - 1 ) ;
r0 = Kp * ( 1 + Tp/(2*Ti) + Td/Tp ) ;

% warunki poczatkowe
u(1:31) = Upp ;
U(1:31) = Upp ;
y(1:31) = Ypp ;
y2(1:31) = Ypp ;
e(1:31) = 0 ;
delta_u = 0;
index = 1;
yzads = [38, 34];
yzad = yzads(index);   %skok wartosci zadanej
yzad2 = yzad - Ypp;
yzadVec(1:800) = yzad;

figure;
% glowna petla symulacji
for k = 32 : 800
if mod(k,400) == 0
index = index + 1;
if index > length(yzads)
```

```
index = length(yzads);
end
yzad = yzads(index);
yzad2 = yzad - Ypp;
end
yzadVec(k) = yzad;

y(k) = readMeasurements(1);

y2(k) = y(k) - Ypp;
e(k) = yzad2 - y2(k) ;

u(k) = r2 * e(k-2) + r1 * e(k-1) + r0 * e(k) + u(k-1) ;

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

%if delta_u > dU_max
%    delta_u = dU_max;
%elseif delta_u < -dU_max
%    delta_u = -dU_max;
%end

u(k) = u(k-1) + delta_u;

if u(k) > U_max - Upp
u(k) = U_max - Upp;
elseif u(k) < U_min - Upp
u(k) = U_min - Upp;
end

U(k) = u(k) + Upp;
sendControls([ 1,5], [ 50,U(k)]);

stairs(y);
pause(0.01);

waitForNewIteration();
end

E = (yzadVec - y)*(yzadVec - y)'
```

4.2. DMC

```
addpath('F:\SerialCommunication');
initSerialControl COM3

Upp = 29;
Ypp = 34.3;
U_min = 0;
U_max = 100;
```

```
Kk  = 800;

D = 350 ;   %horyzont dynamiki
N = D;      %horyzont predykcji
Nu = D;     %horyzont sterowania
lambda = 1;
yzads = [38, 34];
index = 1;
yzad = yzads(index);   %skok wartosci zadanej
yzadVec(1:Kk) = yzad;

%sygnal sterujacy
u = Upp + zeros(1,N) ;
U = Upp + zeros(1,N);
%uchyb
e = zeros(1,N) ;
%wyjscie ukladu
y = zeros(1,Kk) + Ypp ;

du = (zeros(1,D-1))' ;
s=S; % S-policzona wcześniej odpowiedź skokowa
M = zeros(N, Nu) ;
for i = 1:N
for j = 1:Nu
if (i-j+1 > 0)
M(i,j) = s(i-j+1) ;
else
M(i,j) = 0 ;
end
end
end

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

K = (M'*M + lambda*eye(Nu))^-1 * M' ;

%liczenie ke
ke = 0;
for i = 1:N
ke = ke + K(1, i);
end
```

```
kju = K(1,:)*Mp;
y2 = zeros(Kk, 1);

for k = 32:Kk

if mod(k,400) == 0
index = index + 1;
if index > length(yzads)
index = length(yzads);
end
yzad = yzads(index);
end
yzadVec(k) = yzad;

y(k) = readMeasurements(1);

sum = 0;    %suma potrzebna do obliczenia skladowej swobodnej
for j = 1:D-1
if(k-j > 0)
sum = sum + kju(j)*du(k-j);
%w innym przypadku du = 0 wiec sum sie nie zmienia
end
end
y2(k) = y(k) - Ypp;
yzad2 = yzad - Ypp;
du(k) = ke * (yzad2-y2(k)) - sum ;

% --- sprawdzenie, czy przyrost znajduje sie w ograniczeniach  ---
%if du(k) > dU_max
%    du(k) = dU_max;
%elseif du(k) < -dU_max
%    du(k) = -dU_max;
%end

u(k) = u(k-1) + du(k);

if u(k) > U_max - Upp
u(k) = U_max - Upp;
elseif u(k) < U_min - Upp
u(k) = U_min - Upp;
end

U(k) = u(k) + Upp;
sendControls([ 1,5], [ 50,U(k)]);

stairs(y);
pause(0.01);

waitForNewIteration();
```

```
end
```

5. Zad. 5

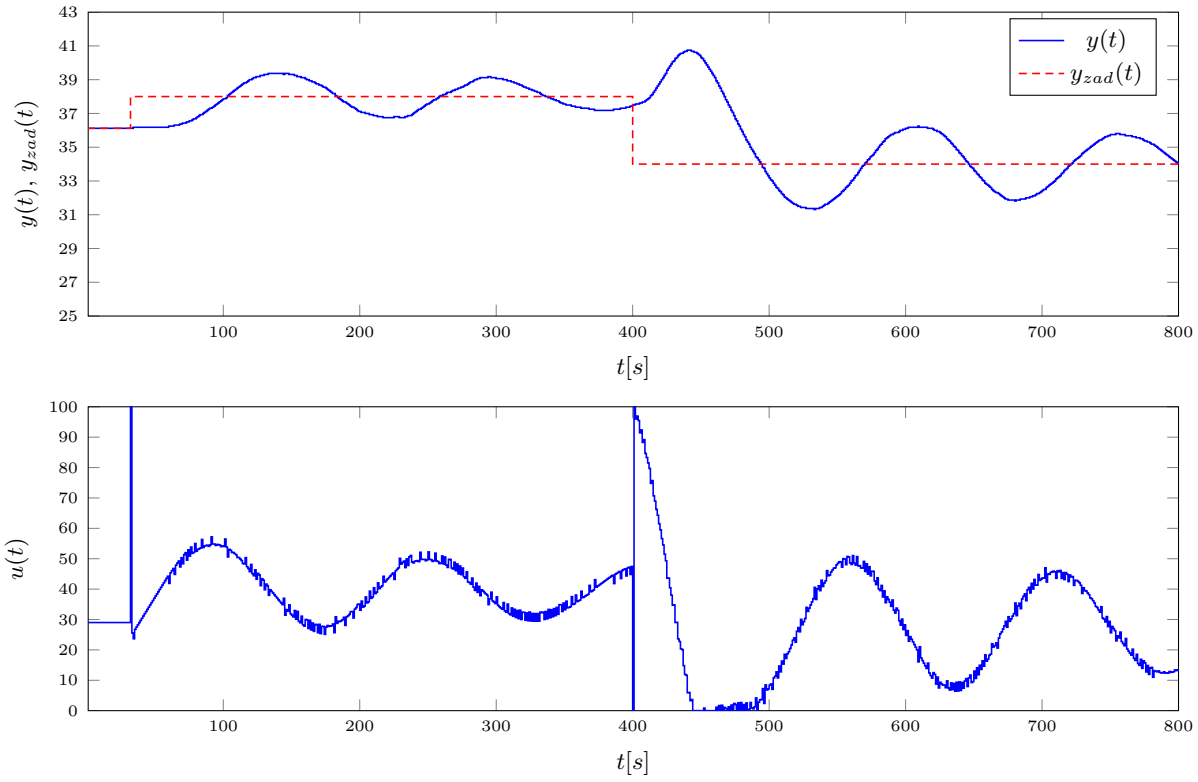
5.1. PID

5.1.1. Próba 1.

Nastawy regulatora PID zostały dobrane metodą eksperymentalną. Za pierwszym razem dobrane parametry($K = 4, T_i = 10, T_d = 10$) nie spełniały oczekiwań, proces regulacji przebiegał bardzo wolno, sygnał wyjściowy wpadł w oscylacje, który były co prawda gasnące, lecz w zdecydowanie zbyt wolnym tempie. Wskaźnik jakości regulacji (błąd) również był bardzo wysoki. Czas regulacji został wyznaczony na 400 sekund, biorąc pod uwagę to, jak wolno zmienia się temperatura na grzałce $G1$ badanego obiektu. Niestety w tym czasie zadana wartość wyjścia nie została osiągnięta. Na wykresie 5.1 został przedstawiony przebieg sygnału wyjściowego na przestrzeni 800 sekund, po pierwszych 400 sekundach wartość zadana uległa zmianie.

5.1.2. Próba 2.

Dla kolejnych parametrów wyznaczonych metodą eksperymentalną, tj. $K = 3, T_i = 40, T_d = 5$ przebieg wyjścia był już zadowalający. Błąd, czas regulacji oraz przeregulowanie były stosunkowo niewielkie. Niepokojące było jednak zjawisko , występujące w chwili zmiany wartości zadanej, tzn. bardzo duży, chwilowy skok sterowania w wyniku czego sygnał wyjściowy również



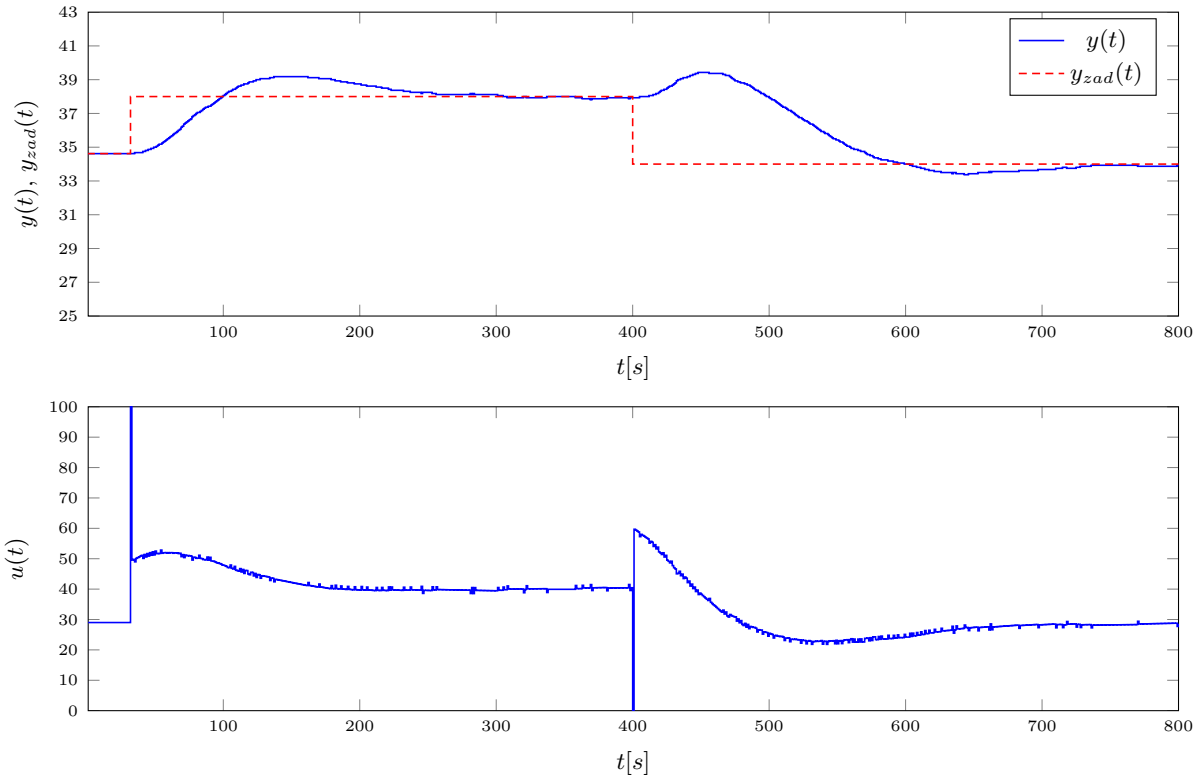
Rys. 5.1. Działanie pierwszego regulatora PID

osiągał duże wartości. Po chwili proces wracał do prawidłowego przebiegu i dążył do stabilizacji w wartości zadanej. Nie były to zakłócenia, gdyż zjawisko to było powtarzalne. Przyczyna tego problemu pozostała niestety nieznana, jednak pomimo tego regulator działał zadowalająco, jego wyniki były bardzo dobre, jak na drugie podejście w metodzie eksperymentalnej.Działanie regulatora dla dwóch wartości zadanych można zaobserwować na rysunku 5.2

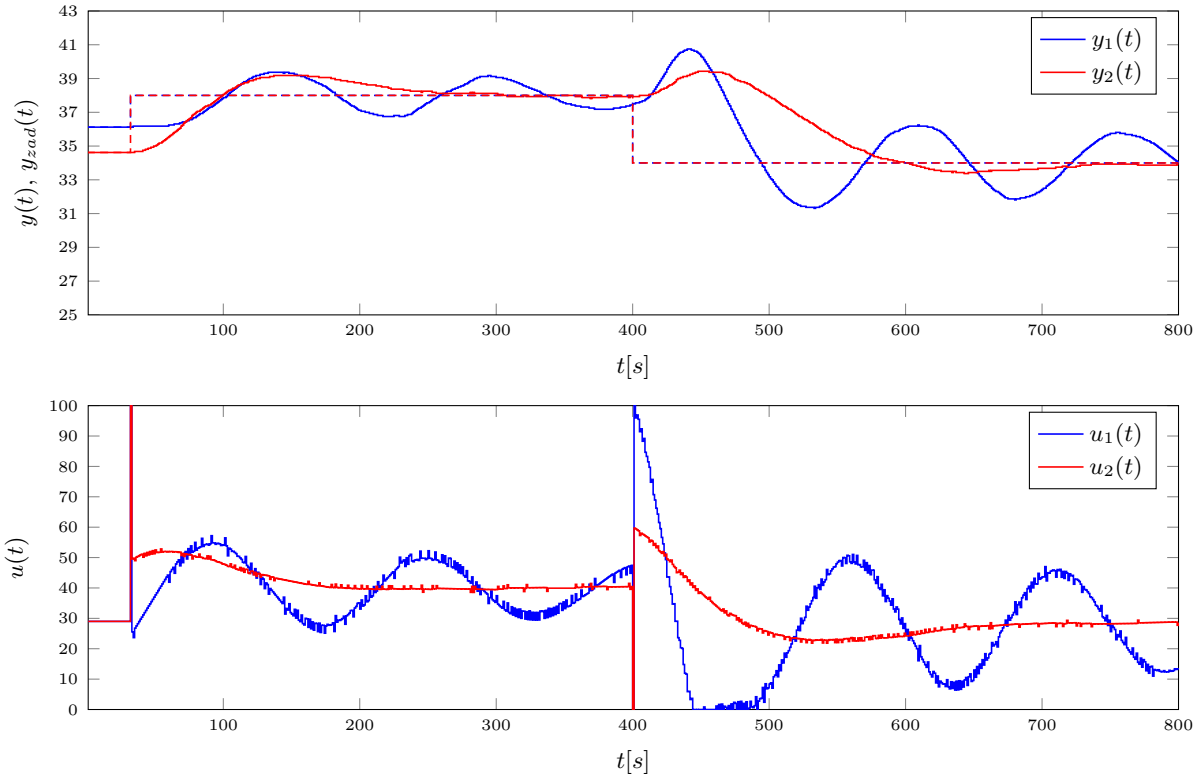
Porównanie obu regulatorów widać na rysunku 5.3

5.2. DMC

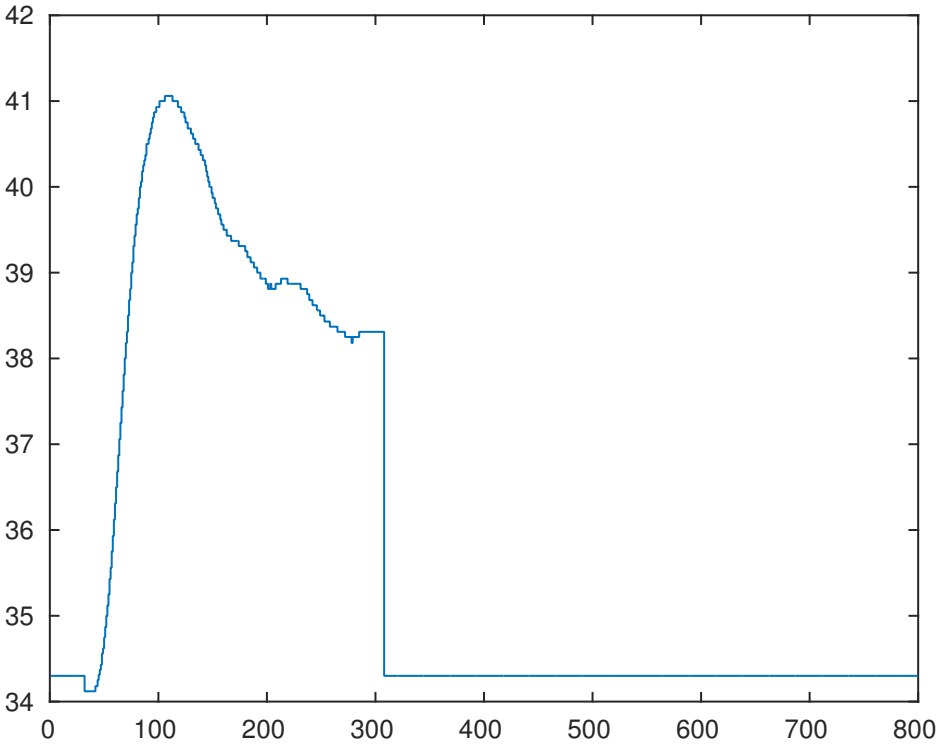
Parametry regulatora DMC również zostały wyznaczone metodą eksperymentalną. W tym przypadku nastawy ($D = N = N_u = 350, \lambda = 1$) zostały wyznaczone tylko raz, gdyż regulator działał już wtedy w miarę poprawnie, a na testowanie innych zabrakło czasu podczas laboratorium. Wskaźnik jakości regulacji w postaci błędu był niewielki, a cały proces regulacji przebiegał bez zastrzeżeń - znośne przeregulowanie, mały czas regulacji. Niestety wykres 5.4, prezentujący działanie tego regulatora jest krótszy niż w przypadku PID(ucięty w 300. sekundzie) co jest spowodowanie brakiem czasu na dokończenie testu. Mimo tego można zaobserwować jego prawidłowe działanie podczas zmiany wartości zadanej z 34 do 38, gdyż stabilizował się przy wartości zadanej.



Rys. 5.2. Działanie drugiego regulatora PID



Rys. 5.3. Porównanie działania obu regulatorów PID



Rys. 5.4. Działanie regulatora DMC