

Wydział Elektroniki i Technik Informatycznych  
Politechnika Warszawska

Projektowanie układów sterowania  
(projekt grupowy)

Sprawozdanie z projektu i ćwiczenia laboratoryjnego  
nr 3, zadanie nr 4

Piotr Chachuła, Cezary Dudkiewicz, Piotr Roszkowski

Warszawa, 2019

# Spis treści

## I. Projekt

## II. Laboratoria

<b>1. Pomiar w punkcie pracy</b>	4
1.1. Komunikacja z obiektem	4
1.2. Punkt pracy	4
<b>2. Charakterystyka obiektu</b>	5
2.1. Inne punkty pracy	5
2.2. Charakterystyka statyczna obiektu	9
2.3. Wzmocnienie statyczne	9
<b>3. Zastosowanie tradycyjnych regulatorów PID i DMC</b>	10
3.1. Regulator PID	10
3.1.1. Postępowanie	10
3.1.2. Wyniki symulacji	10
3.2. Regulator DMC	10
3.3. Wnioski	11
3.4. Rozmywanie regulatorów	11
3.4.1. Funkcje aktywacji	11

Część I

**Projekt**

Część II

## Laboratoria

# 1. Pomiar w punkcie pracy

## 1.1. Komunikacja z obiektem

Komunikacja z obiektem odbywa się za pomocą funkcji napisanych w środowisku MatLab. Najprostszy program użyty w tym projekcie, który posłużył także do późniejszego pobierania odpowiedzi skokowych znajduje się poniżej (jest to fragment skryptu L3\_1.m).

```
addpath('F:\SerialCommunication'); % add a path to the functions
initSerialControl COM3 % initialise com port

N = 420;
step_response = zeros(N+1,1);
i = 0;

while(i<=N)
    %% obtaining measurements
    measurements = readMeasurements(1:7); % read measurements from 1 to 1

    %% processing of the measurements and new control values calculation
    disp([measurements(1),i]);
    step_response(i+1)=measurements(1);

    %% sending new values of control signals
    sendNonlinearControls(29) % new corresponding control values disp(measurements); % process measurements

    i=i+1;
    step_response(i)=measurements(1);

    waitForNewIteration(); % wait for new batch of measurements to be ready
end
```

W tym zadaniu używamy funkcji `sendNonlinearControls`, który wysyła sterowanie w sposób symulujący nieliniowość obiektu. Napisany skrypt działał poprawnie, pozwala na sterowanie sygnałami G1, W1 oraz pomiar T1.

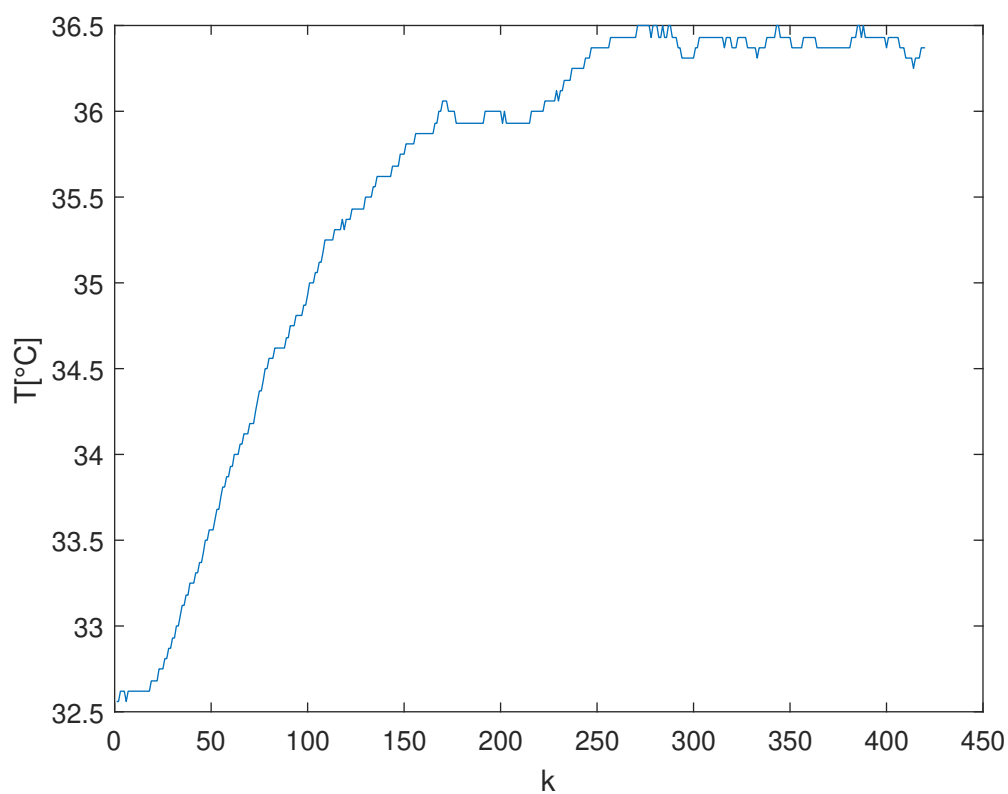
## 1.2. Punkt pracy

Doprowadzono obiekt do punktu pracy, tj. ustawiono wartości sygnałów W1 na 50, G1 na 29 i poczekano na ustabilizowanie obiektu (ponieważ obiekt jest rzeczywisty wahania temperatury są nieuniknione, zwłaszcza biorąc pod uwagę fakt lokalizacji stanowiska nr 4 w miejscu obok którego przechodzi dużo osób - wszelkie pomiary terażniejsze oraz późniejsze mogą być zaburzone właśnie przez to). Wartość temperatury w punkcie pracy wynosi  $T1=35,43^{\circ}\text{C}$ .

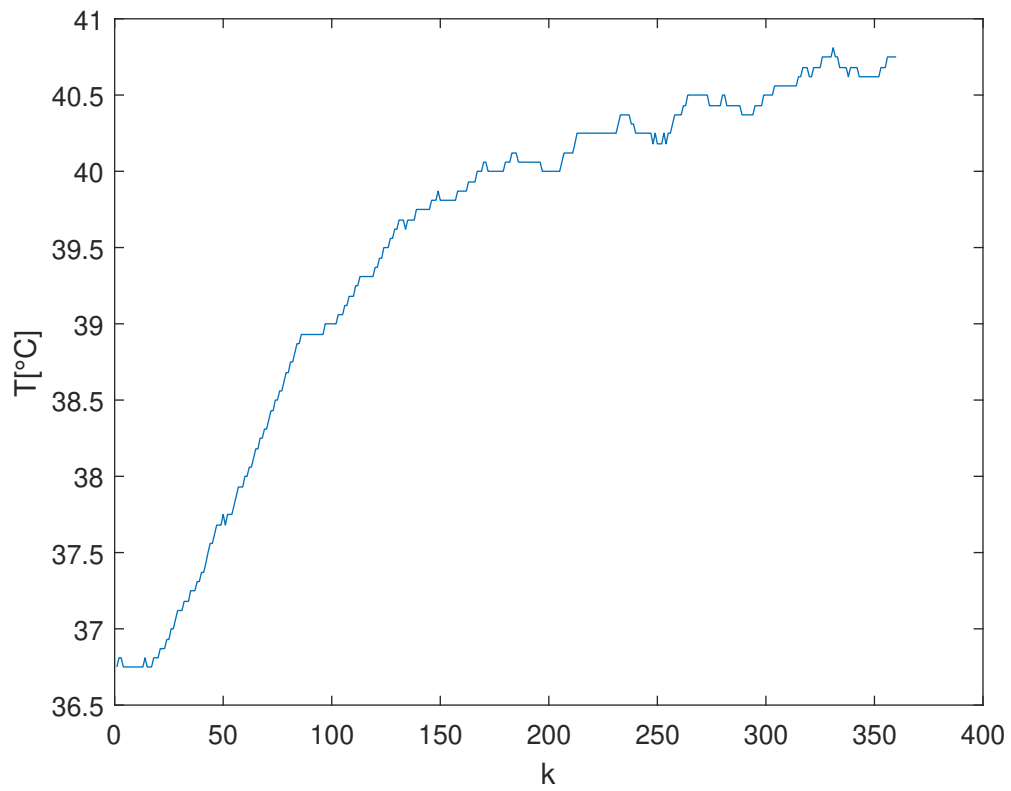
## 2. Charakterystyka obiektu

### 2.1. Inne punkty pracy

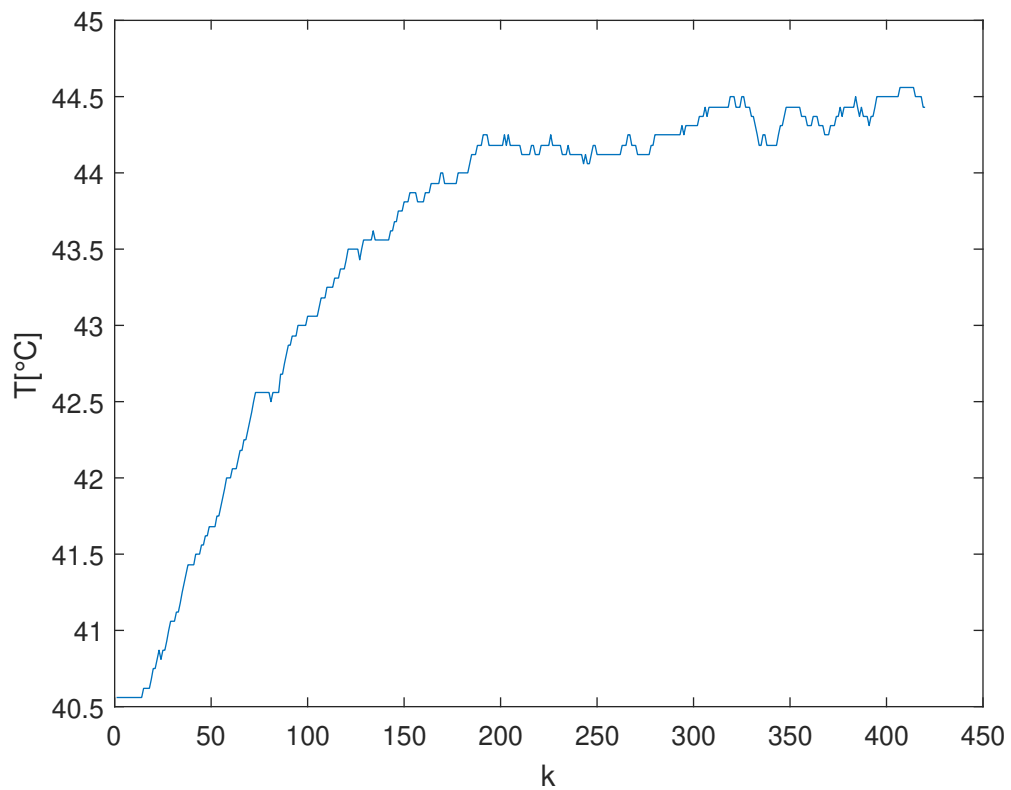
W celu pobrania wartości wyjścia dla innych punktów sterowania postępowano następująco: najpierw pobudzono układ sterowaniem równym  $G1=20$  i poczekano na jego stabilizację. Następnie dokonywano skoków tej wartości sterowania o 10, aż do wartości  $G1=80$ . Przebieg eksperymentu ilustrują poniższe wykresy (skoki sterowania następowały w chwili  $t=0$ ):



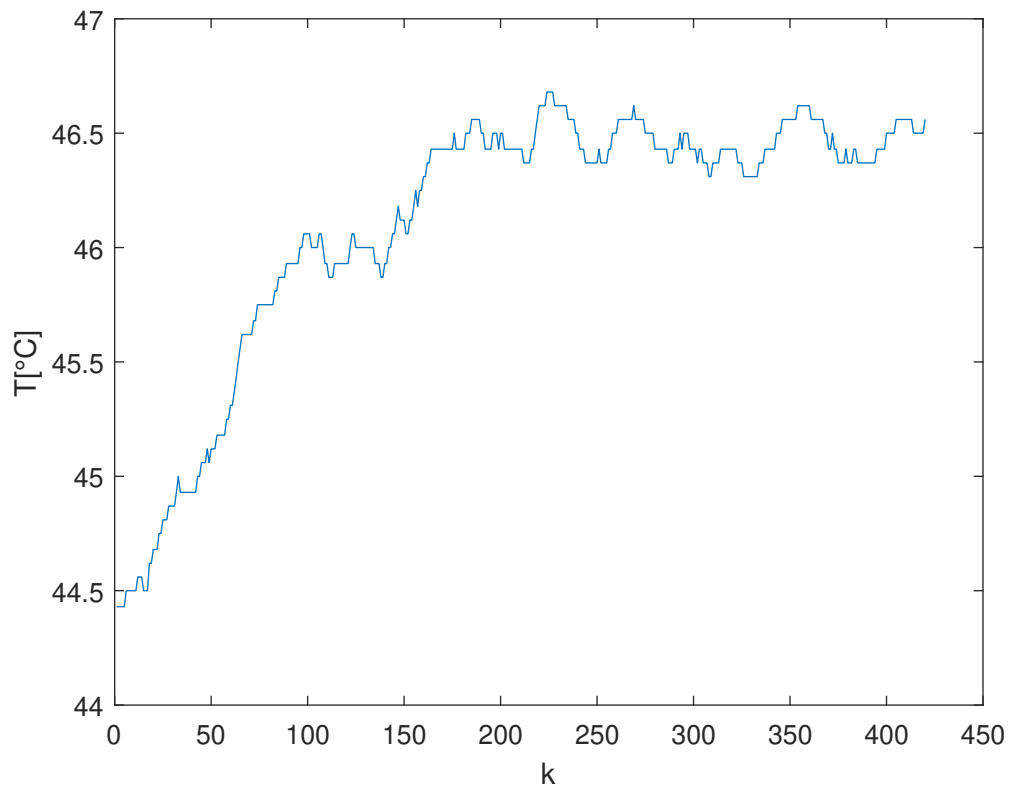
Rys. 2.1. Skok wartości sterowania z 20 do 30



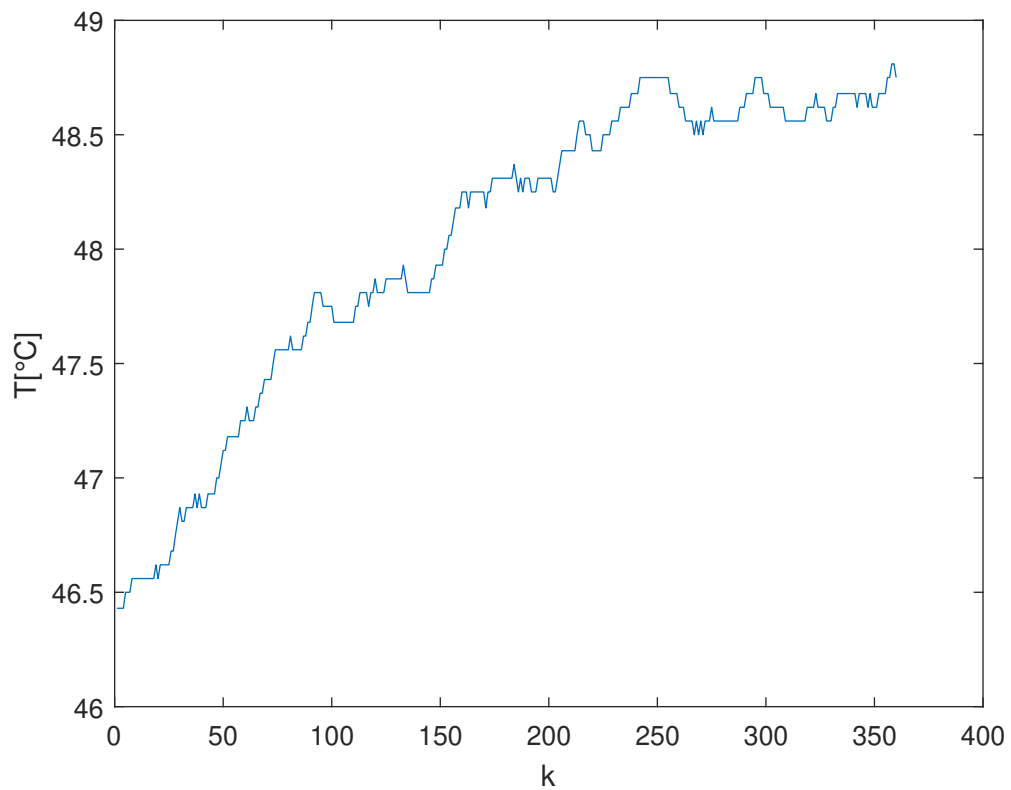
Rys. 2.2. Skok wartości sterowania z 30 do 40



Rys. 2.3. Skok wartości sterowania z 40 do 50

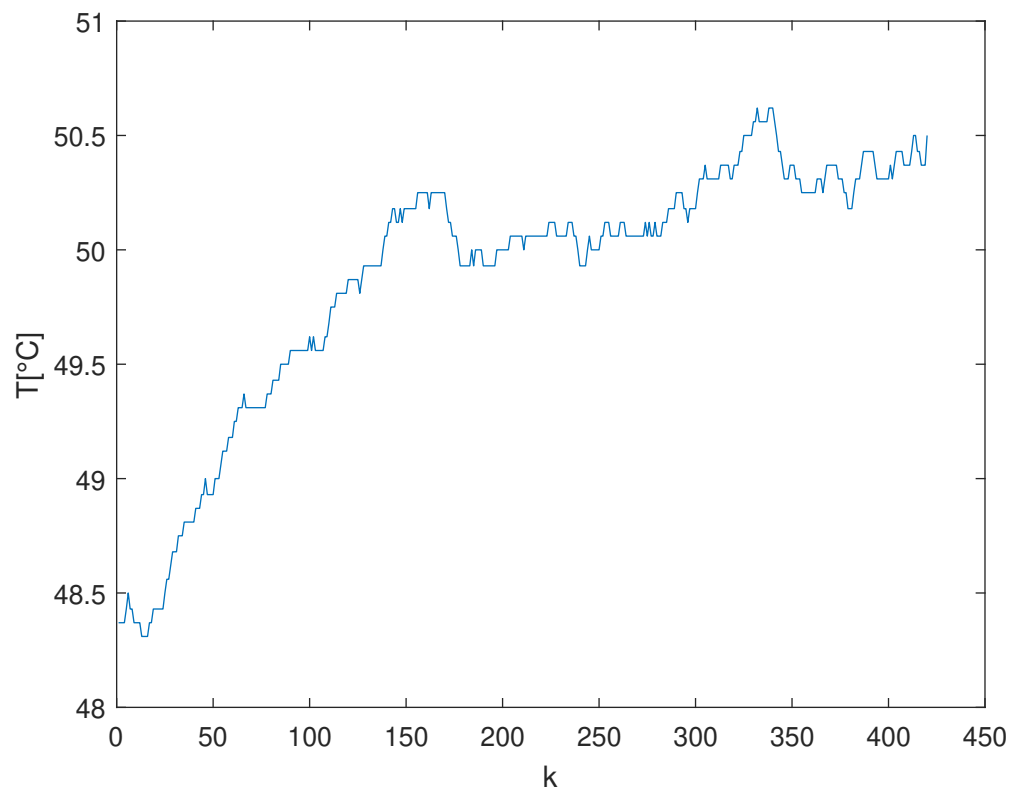


Rys. 2.4. Skok wartości sterowania z 50 do 60



Rys. 2.5. Skok wartości sterowania z 60 do 70





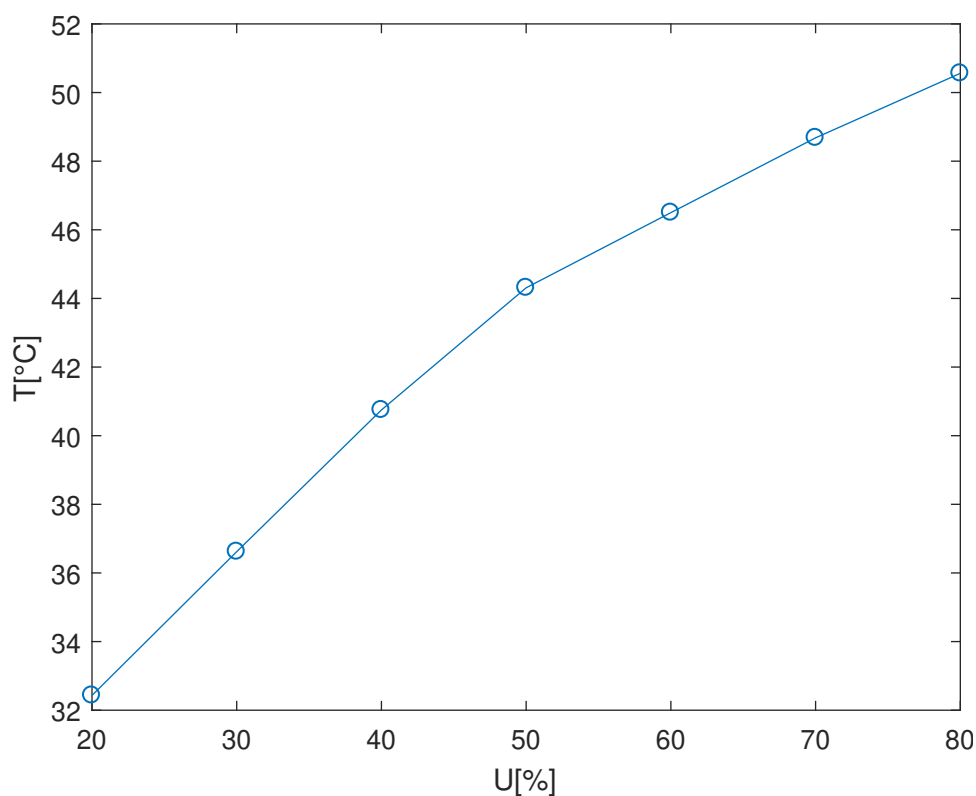
Rys. 2.6. Skok wartości sterowania z 70 do 80

Wyniki przedstawiono w tabeli:

G1[%]	T[°C]
20	32,43
30	36,62
40	40,75
50	44,31
60	46,5
70	48,68
80	50,56

## 2.2. Charakterystyka statyczna obiektu

Charakterystykę statyczną obiektu w przedziale sterowań G1 od 20 do 80% przedstawiono na wykresie 2.7:



Rys. 2.7. Charakterystyka statyczna obiektu

## 2.3. Wzmocnienie statyczne

Z wykresu charakterystyki liniowej można stwierdzić, że obiekt nie jest całkowicie liniowy, występuje załamanie charakterystyki w punkcie  $U=50\%$ . Jednak obiekt jest kawałkami liniowy, przejawia właściwości liniowe w przedziale 20-50% oraz inne właściwości liniowe w przedziale 50-80% - na tych odcinkach obiekt zachowuje się praktycznie w sposób liniowy. Dlatego też możemy policzyć wzmocnienie statyczne obiektu w tych przedziałach sterowania:

$$K_{20-50\%} = \frac{Y(50) - Y(20)}{50 - 20} = \frac{44,31 - 32,43}{50 - 20} = 0,396 \quad (2.1)$$

$$K_{50-80\%} = \frac{Y(80) - Y(50)}{80 - 50} = \frac{50,56 - 44,31}{80 - 50} \approx 0,208 \quad (2.2)$$

## 3. Zastosowanie tradycyjnych regulatorów PID i DMC

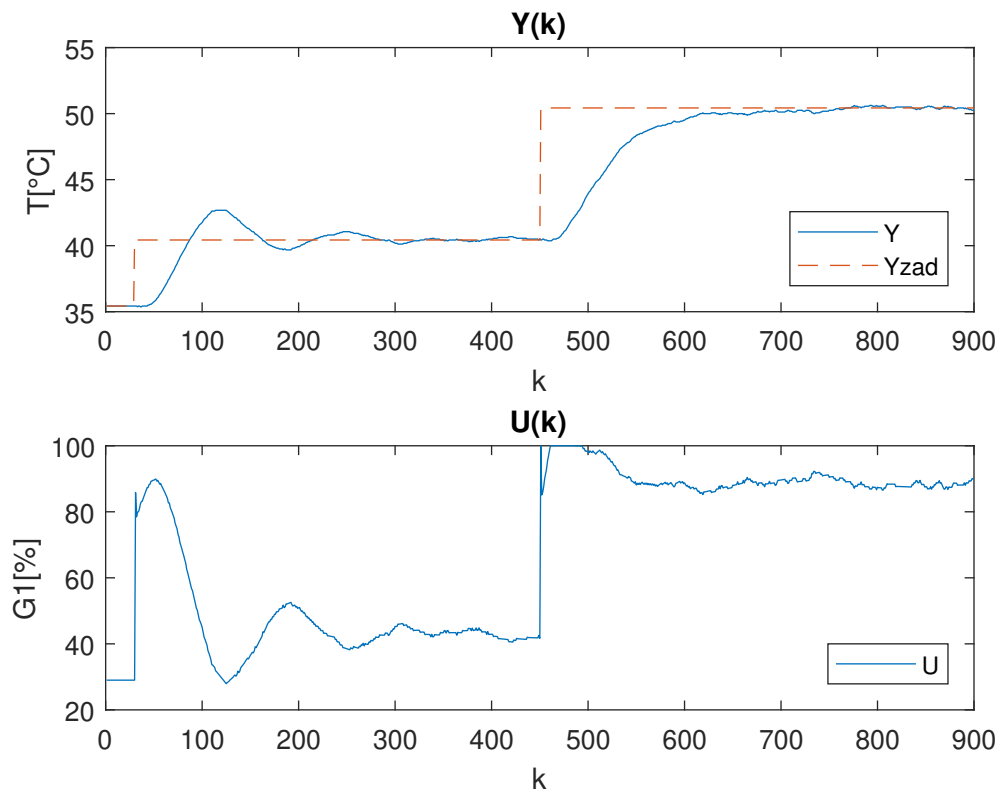
### 3.1. Regulator PID

#### 3.1.1. Postępowanie

Zarówno w przypadku regulatora PID oraz DMC zostaną użyte skrypty z projektu nr 1 ( `doLabPID.m` `doLabDMC.m`). Jedyna zmiana nastąpi w wysyłaniu sterowania do obiektu, gdzie funkcję `sendControls` zastępujemy funkcją `sendNonlinearControls`, która ma na celu symulację braku liniowości obiektu na całym obszarze wartości sterowań.

#### 3.1.2. Wyniki symulacji

Wyniki symulacji przedstawiono na wykresie 3.1:



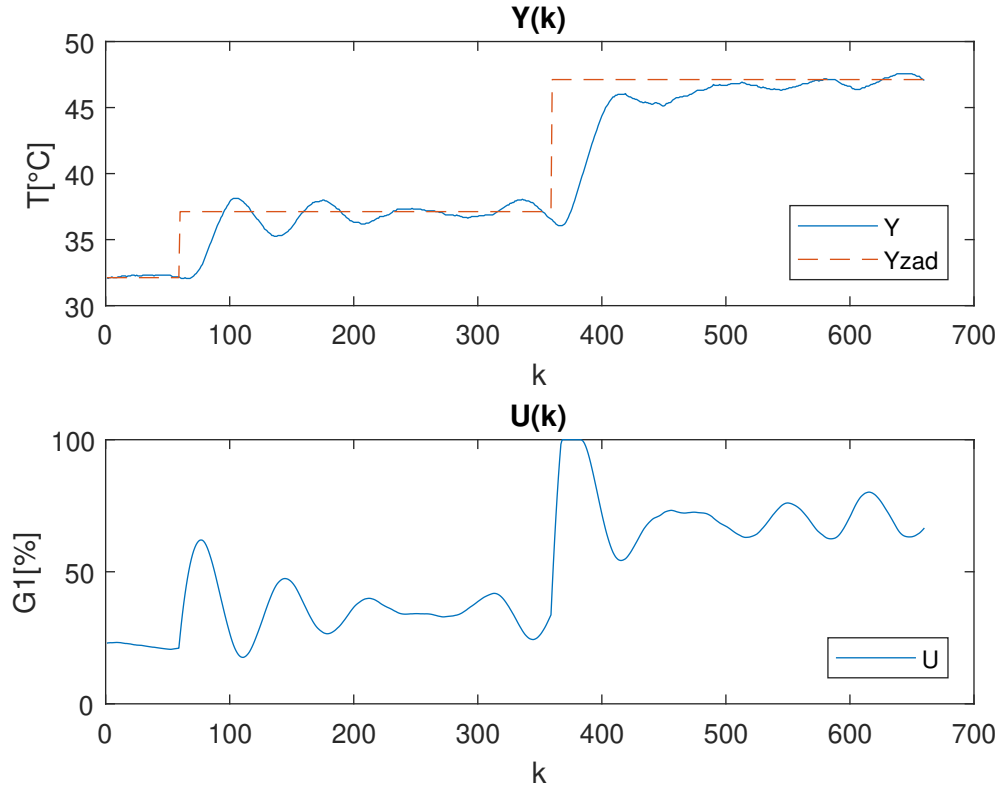
Rys. 3.1. Symulacja dla pojedynczego regulatora PID o parametrach  $K = 9,65$ ,  $T_i = 60$ ,  $T_d = 0,17$

Błąd (suma kwadratów odchyłek) wyniósł  $E = 6077$ .

### 3.2. Regulator DMC

Wyniki symulacji przedstawiono na wykresie 3.2:

Błąd (suma kwadratów odchyłek) wyniósł  $E = 3795$  (wyniki można ze sobą porównywać mimo krótszego czasu trwania symulacji, liczba skoków pozostała ta sama - tam są generowane głównie uchyby).



Rys. 3.2. Symulacja dla pojedynczego regulatora DMC o parametrach  $D = 360$ ,  $N = 120$ ,  $N_u = 20$ ,  $\lambda = 1$

### 3.3. Wnioski

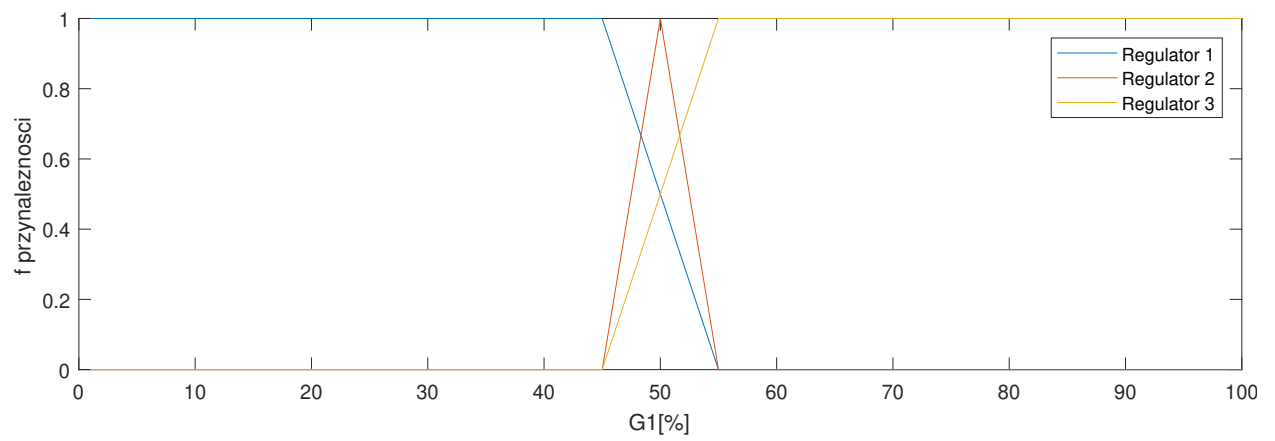
Pomimo że regulatory działają, temperatura zadana jest mniej więcej osiągnięta, jednak ich regulacja jest dosyć wolna, a wyjście oscyluje. W celu poprawy regulacji dokonano rozmycia tych regulatorów.

### 3.4. Rozmywanie regulatorów

#### 3.4.1. Funkcje aktywacji

Dla obu regulatorów dobrano identyczne funkcje aktywacji dla trzech regulatorów lokalnych. Rozmywane one są względem wartości sterowana w poprzednim momencie. Zdecydowano się na to z dwóch powodów: po pierwsze układ może zmieniać swoje punkty pracy w zależności od warunków atmosferycznych otoczenia, rozmywanie po wyjściu naraża nas na wpływ takich odchył i spadek jakości regulacji; po drugie nieliniowość na obiekcie jest wprowadzana sterowaniem, tzn. obiekt (według tego co ustalono na projektach 1 oraz 2) jest w miarę liniowy, a jego właściwości nie mogły się zmienić, nieliniowość narzucana jest przez funkcję `sendNonlinearControls`, która jakoby przerabia sterowanie tak, aby układ zachowywał się jak nieliniowy.

Pierwszy regulator lokalny będzie aktywny głównie w przedziale 0-50% (na tym przedziale obiekt zachowuje się jak liniowy), drugi w przedziale 45-55%, a trzeci głównie w przedziale 50-100%, patrz rysunek 3.3:



Rys. 3.3. Funkcje aktywacji regulatorów lokalnych