

Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska

Projektowanie układów w sterowania
(projekt grupowy)

Sprawozdanie z projektu i ćwiczenia laboratoryjnego
nr 1, zadanie nr 1

Hubert Kozubek, Przemysław Michalczewski

Warszawa, 2021

Spis treści

1.	Cele projektu i laboratoriów	1
2.	Przebieg laboratorium	1
2.1.	Zad 1	3
2.2.	Zad 2	3
2.3.	Zad 3	3
2.4.	Zad 4	6
2.5.	Zad 5	9

1. Cele projektu i laboratoriów

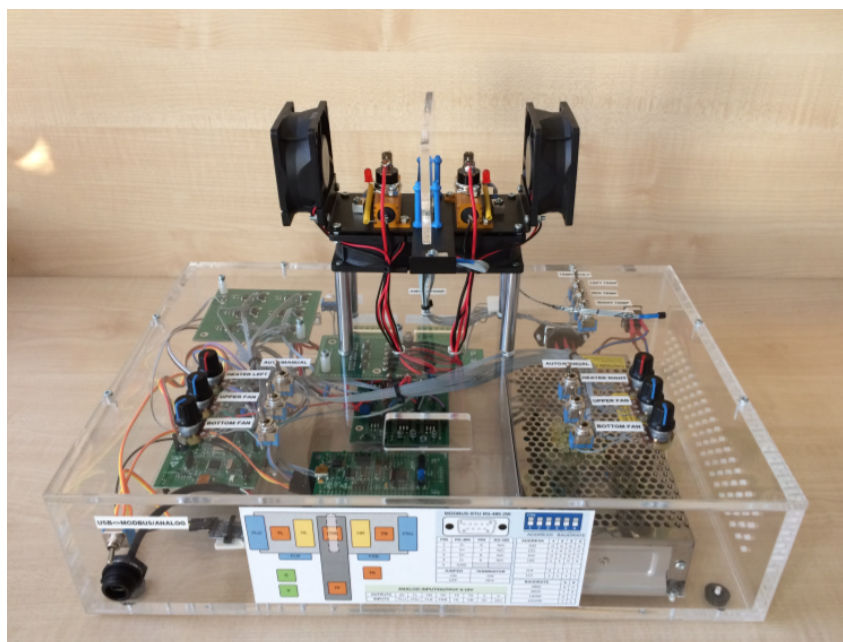
Celem niniejszego laboratorium oraz projektu było zaprojektowanie, implementacja, weryfikacja poprawności działania oraz dobór parametrów algorytmów regulacji jednowymiarowego procesu na grzewczym stanowisku laboratoryjnym przedstawionym na rys 1.

2. Przebieg laboratorium

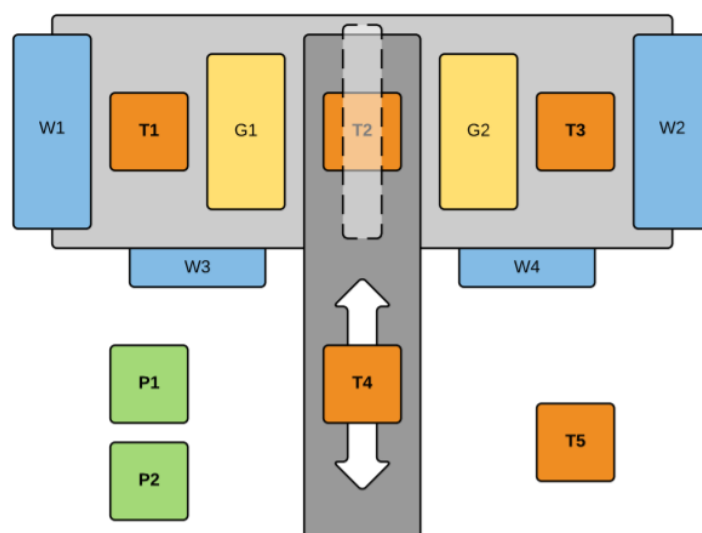
Rozpoczynając pracę na stanowisku laboratoryjnym należało ustawić moc wentylatora W1 na 50%. Wentylator ten był traktowany jako cecha otoczenia. Dodatkowo sprawiał on, że temperatura grzałki opadała szybciej, co było szczególnie przydatne pomiędzy doświadczeniami.

W ramach laboratorium należało wykonać 5 zadań.

1. Odczytać wartość pomiaru termometru T1 dla mocy 26 grzałki G1%.
2. Wyznaczyć odpowiedź skokową procesu dla 3 różnych wartości G1%.



Rys. 1. Stanowisko grzewąco-chłodzące, używane w trakcie laboratoriów



Rys. 2. Schemat stanowiska grzejąco-chłodzącego

3. Wybrać jedną z odpowiedzi skokowych, przekształcić ją i wykorzystać w algorytmie DMC.
4. Zaimplementować algorytm PID i DMC, od regulacji procesu stanowiska, w języku MATLAB.
5. Dobrać nastawy algorytmu PID oraz parametry algorytmu DMC metodą eksperymentalną.

2.1. Zad 1

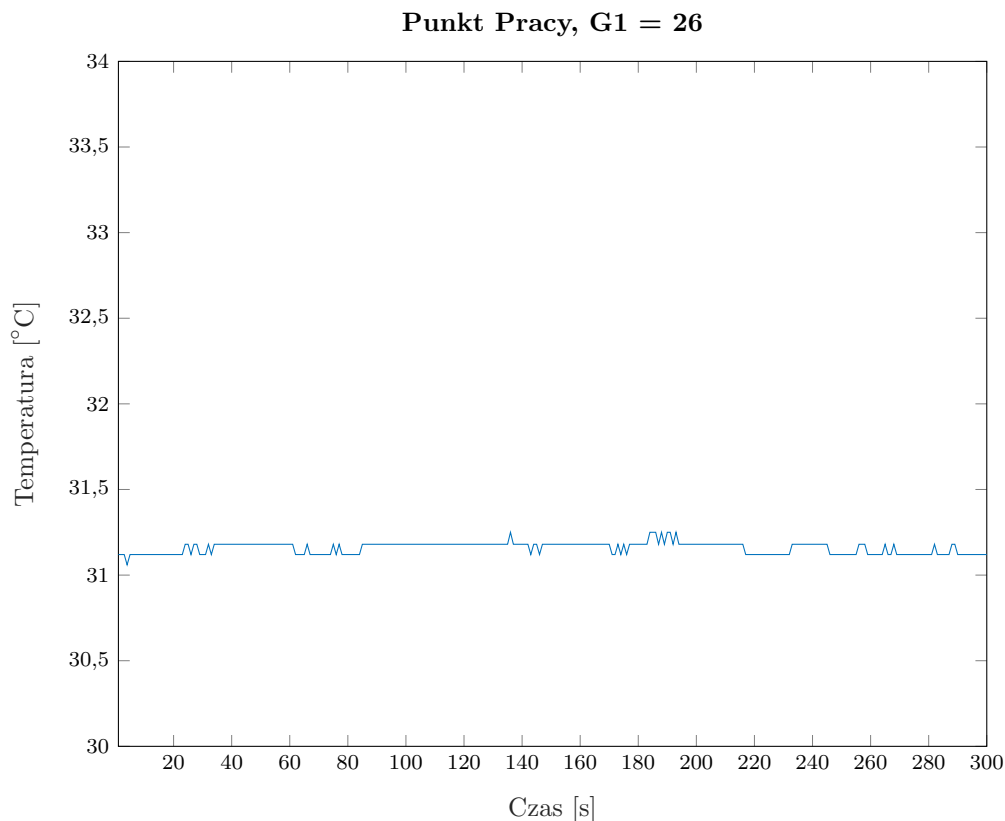
W pierwszej kolejności należało sprawdzić możliwość sterowania i pomiaru w komunikacji ze stanowiskiem. Następnie odczytać wartość temperatury termometru T1 w wyznaczonym punkcie pracy $G1=26\%$. Po ustawieniu mocy grzałki i odczekaniu, aż temperatura T1 ustabilizuje się, odczytana wartość termometru T1 wynosiła $31,12\text{ }^{\circ}\text{C}$. Wykres temperatury na termometrze T1 został przedstawiony na rys. 3

2.2. Zad 2

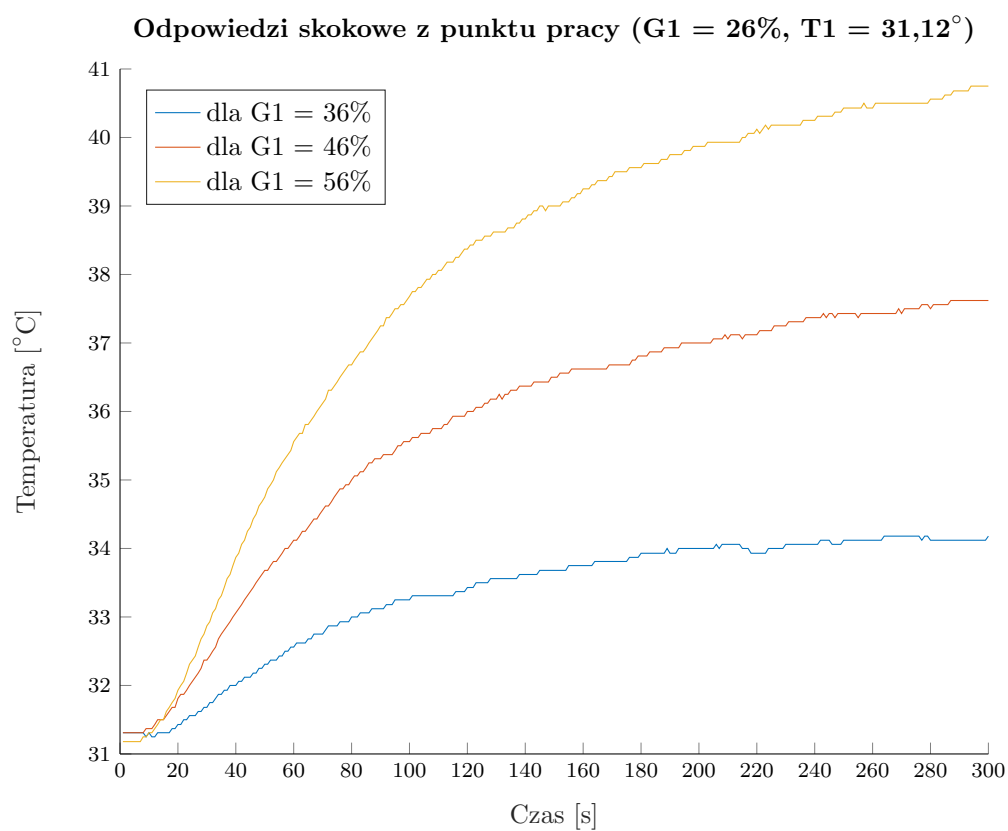
W tej części laboratorium należało przeprowadzić eksperyment dla 3 różnych wartości mocy grzałki G1. Rozpoczynając eksperyment z punktu pracy $G1=26\%$, wyznaczono odpowiedzi skokowe procesu. Eksperyment był wykonywany dla trzech różnych zmian sygnału sterującego, $G1=36\%$, $G1=46\%$ oraz $G1=56\%$. Wykresy przedstawiające zmiany temperatury przedstawiono na rys. 4

2.3. Zad 3

Wykonanie tego zadania polegało na przekształceniu jedną z odpowiedzi skokowych, tak aby otrzymać odpowiedź skokową używaną w algorytmie DMC. W tym celu wybrano drugą odpowiedź skokową, tj. skok G1 z mocy 26% do mocy 46%. Do przekształcenia zebranej



Rys. 3. Ustalanie się temperatury dla punktu pracy



Rys. 4. Odpowiedź skokowa procesu

odpowiedzi skokowej, na taką nadającą się do algorytmu DMC wykorzystano program TODO: "SkokDMC.m". Program ten wylicza potrzebną odpowiedź skokową przy użyciu prostego wzoru.

$$S(i) = \frac{Y(i) - Y_{pp}}{U_{skok} - U_{pp}} \quad (1)$$

gdzie:

- $S(i)$ - odpowiedź skokowa potrzebna do algorytmu DMC,
- $Y(i)$ - odpowiedź skokowa przed przekształceniem,
- Y_{pp} - wartość wyjścia w chwili $k=0$ (tutaj $Y_{pp} = 31,12$),
- U_{skok} - wartość sterowanie w chwili $k=0$ i później (tutaj $U_{skok} = 46$),
- U_{pp} - wartość sterowania przed chwilą $k=0$ (tutaj $U_{pp} = 26$)

W ten sposób przekształcona odpowiedź skokowa została zapisana do pliku TODO: "dane1.mat" i wykorzystana w dalszych częściach laboratoriów.

Poza przekształceniem odpowiedzi skokowej należało ją jeszcze przybliżyć używając w tym celu członu inercyjnego drugiego rzędu z opóźnieniem.

$$G(s) = \frac{K}{(sT_1 + 1)(sT_2 + 1)} e^{-T_d s} \quad (2)$$

Po dyskretyzacji danej transmitancji otrzymujemy

$$G(z) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} z^{-T_d} \quad (3)$$

gdzie

$$\begin{aligned} a_1 &= -\alpha_1 - \alpha_2 \\ a_2 &= \alpha_1 \alpha_2 \\ \alpha_1 &= e^{-\frac{1}{T_1}} \\ \alpha_2 &= e^{-\frac{1}{T_2}} \\ b_1 &= \frac{K}{T_1 - T_2} [T_1(1 - \alpha_1) - T_2(1 - \alpha_2)] \\ b_2 &= \frac{K}{T_1 - T_2} [\alpha_1 T_2(1 - \alpha_2) - \alpha_2 T_1(1 - \alpha_1)] \end{aligned} \quad (4)$$

Z wykresu odpowiedzi skokowej procesu zostało odczytane opóźnienie. W naszym przypadku $T_d = 9$. Aby wyznaczyć wartości pozostałych współczynników użyto dostępnej w matlabie funkcji `ga`, która minimalizuje wartość zadanej funkcji z wykorzystaniem algorytmu generycznego. Funkcja minimalizowana, to funkcja wyliczająca sumę kwadratów błędów pomiędzy odpowiedzią skokową, a transmitancją przybliżającą.

```
% aproksymacja odpowiedzi skokowej
```

```
function ERR = AproksSkokDMC(X)
```

```
data = load('dane1.mat');
S = data.S;
time = data.time;
```

```
T1 = X(1);
```

```

T2 = X(2);
K = X(3);
Td = 9;
y(1:time) = 0;

alpha1 = exp(-1/T1);
alpha2 = exp(-1/T2);
a1 = -alpha1-alpha2;
a2 = alpha1*alpha2;
b1 = K*(T1*(1-alpha1)-T2*(1-alpha2))/(T1-T2);
b2 = K*(alpha1*T2*(1-alpha2)-alpha2*T1*(1-alpha1))/(T1-T2);

for k = Td+3:time
    y(k) = b1 + b2 - a1*y(k-1) - a2*y(k-2);
end

e = S - y';

ERR = (norm(e))^2;
end

```

Następnie używając skryptu TODO:”Optymalizacja.m” zostały wyznaczone pozostałe parametry transmitancji przybliżającej odpowiedź skokową. Ostateczne wartości parametrów to

$$\begin{aligned}
 K &= 0.330938 \\
 T_1 &= 0.000907 \\
 T_2 &= 82.104622 \\
 T_d &= 9
 \end{aligned}
 \tag{5}$$

Wykres zarówno odpowiedzi skokowej, jak i transmitancji ją przybliżającej został zamieszczony na rys. 5

2.4. Zad 4

Kolejnym podpunktem zadań laboratoryjnych było zaimplementowanie algorytmu regulacji PID oraz DMC w języku MATLAB.

Regulator PID

```

% implemrntacja PID

function U = PID(e)

    persistent Upop
    persistent e0
    persistent e1
    persistent e2
    persistent K
    persistent Ti
    persistent Td
    persistent Tp
    persistent r2
    persistent r1

```

```

persistent r0

% Ograniczenia sterowania
Gmax = 100;
Gmin = 0;

%      Upp = 26;
%      Ypp = 31.12;

if isempty(e0)
    Upop = 0;           % sterowanie w punkcie pracy
    e0=0;
    e1=0;
    e2=0;

    % Nastawy regulatora
    K = 0.5 * 43 * 1.5; %Kk = 43, Tk = 36
    Ti = 0.5 * 36*2;% * 4; %inf;      10
    Td = 0.125 * 36;% * 0.6; %      0.4
    Tp = 1;

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

% przesunięcie uchybów
e2 = e1;
e1 = e0;
e0 = e;

U = r2*e2 + r1*e1 + r0*e0 + Upop;

if U > Gmax
    U = Gmax;
end

if U < Gmin
    U = Gmin;
end

Upop = U;

end

```

Regulator DMC

```

function U = DMC(yzad, y, D, N, Nu, lambda)

persistent init
persistent S

```



```

persistent M
persistent Mp
persistent K
persistent dUP
persistent Upop

if isempty(init)

    % Wczytanie macierzy S z pliku dane1.mat
    data = load('dane1.mat');
    S = data.S;

    % Odpowiedź skokowa aproksymowana
    % data = load('Sapro.mat');
    % S = data.Sapro;

    % przedłużenie wektora S
    for i = D+1:D+N
        S(i) = S(D);
    end

    % Inicjalizacja macierzy
    M = zeros(N, Nu);
    for i = 1:Nu
        M(i:N, i) = S(1:N-i+1);
    end

    Mp = zeros(N, D-1);
    for i = 1:(D-1)
        Mp(1:N, i) = S(i+1:N+i) - S(i);
    end

    I = eye(Nu);

    K = ((M*M + lambda*I)^(-1))*M';
    dUP = zeros(D-1,1);
    Upop = 26;
    init = 1;
end

% Ograniczenia sterowania
Gmax = 100;
Gmin = 0;

Y0 = zeros(N,1);
dU = zeros(Nu,1);

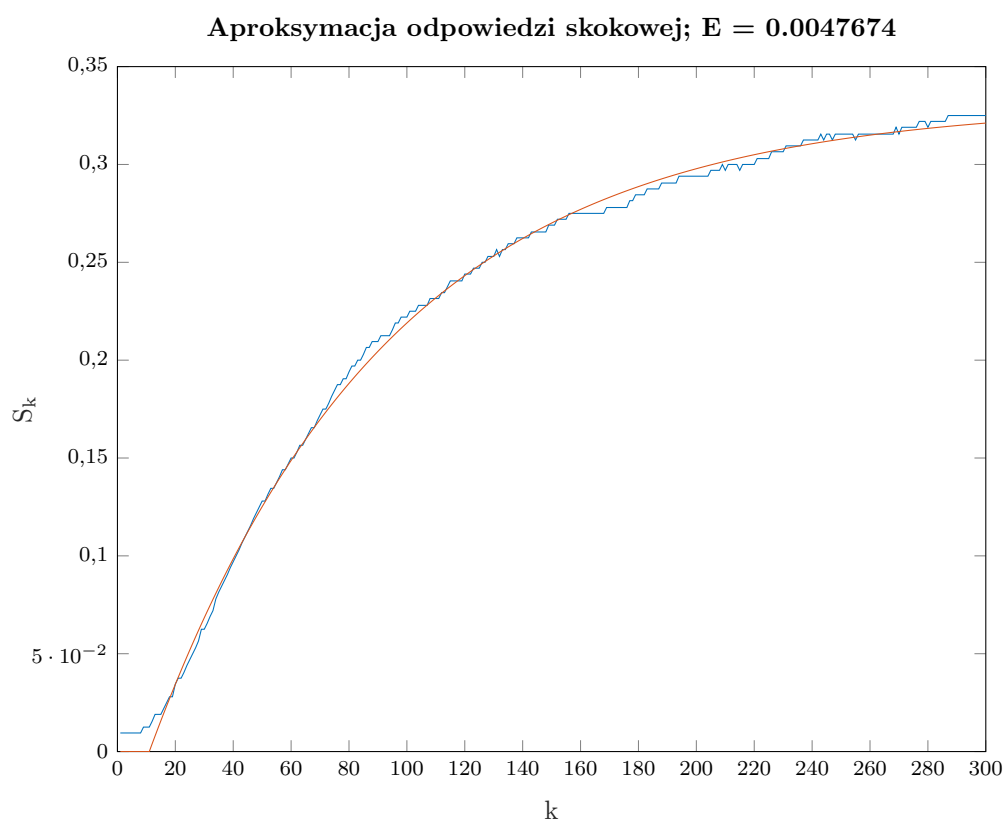
% liczone online
Yzad = yzad*ones(N,1);
Y = y*ones(N,1);

```

```
Y0 = Y + Mp*dUP;  
dU = K*(Yzad - Y0);  
du = dU(1);  
  
for n=D-1:-1:2  
    dUP(n) = dUP(n-1);  
end  
dUP(1) = du;  
  
U = Upop + du;  
  
if U > Gmax  
    U = Gmax;  
end  
  
if U < Gmin  
    U = Gmin;  
end  
  
Upop = U;  
end
```

2.5. Zad 5

Ostatnim zadaniem był dobór nastawów obu algorytmów regulacji.



Rys. 5. Odpowiedź skokowa procesu, oraz transmitancja ją aproksymująca