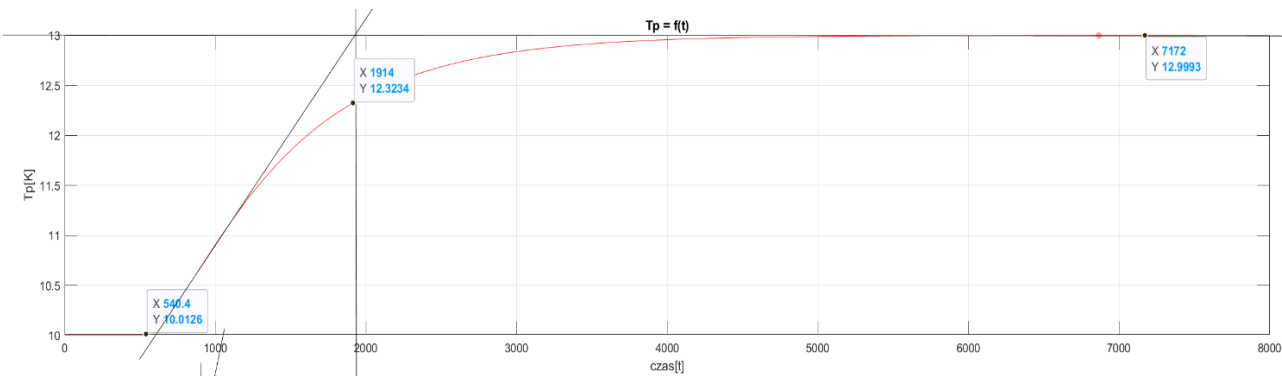


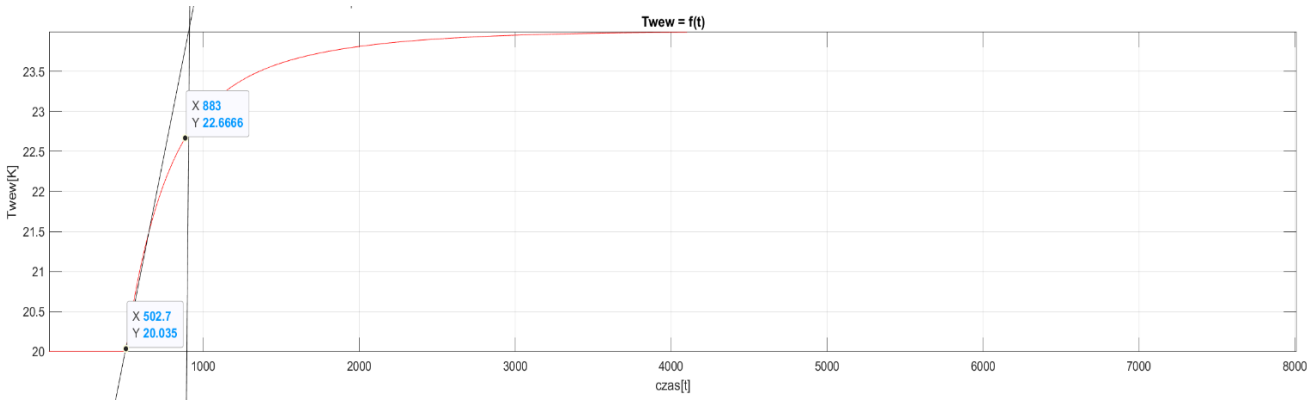
Autor Sprawozdania	Projektowanie układu Sterowania Semestr letni 2022/23	Termin: <i>Środa Godz. 18:55</i> <i>Grupa: Y02-44c</i>
Hubert Kowalczyk 259550	Sprawozdanie 2	Sprawozdanie z terminów 5,6,7,8,9,10
Prowadzący:		Data oddania sprawozdania: 27.05.2023
Mgr inż. Marta Lampasiak		Ocena:

1) Wyniki identyfikacji

1.1) Metoda stycznej

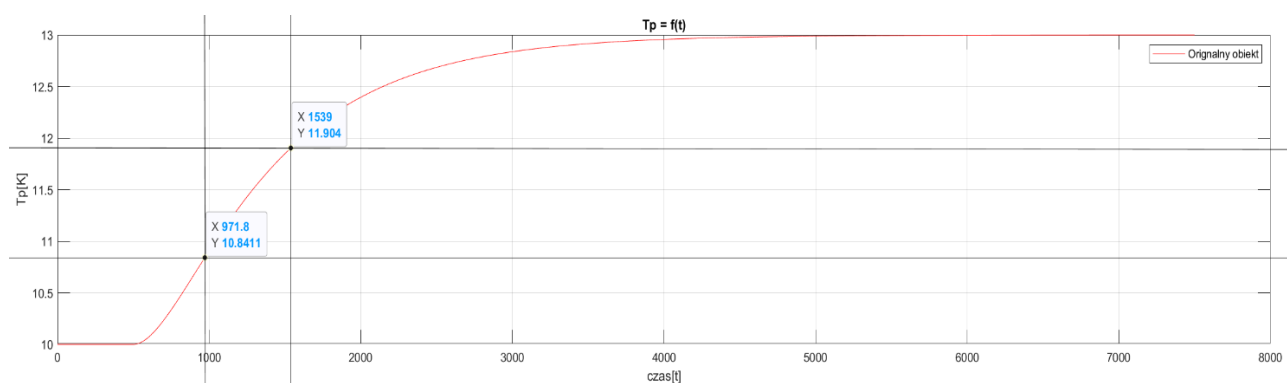


Wykres 1 Wykres Przedstawiający wykonanie identyfikacji za pomocą metody stycznej T_p

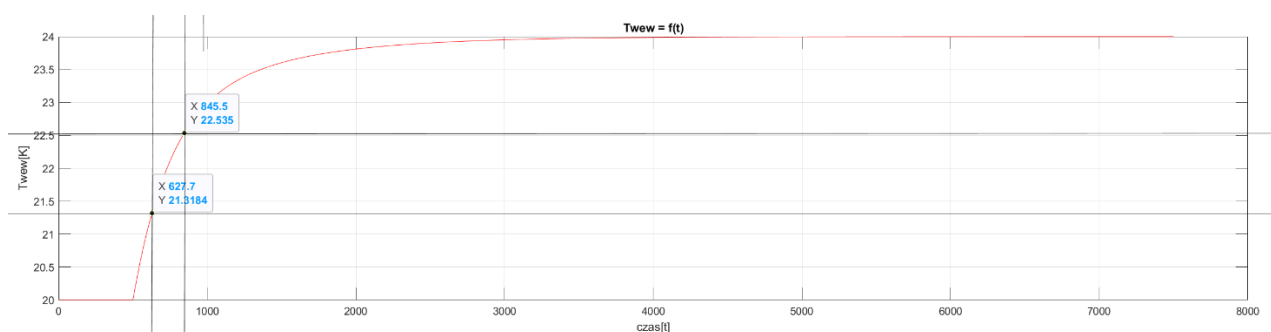


Wykres 2 Wykres Przedstawiający wykonanie identyfikacji za pomocą metody stycznej T_{wew}

1.2) Metoda dwupunktowa



Wykres 3 Wykres Przedstawiający wykonanie identyfikacji za pomocą metody dwupunktowej T_p



Wykres 4 Wykres Przedstawiający wykonanie identyfikacji za pomocą metody dwupunktowej T_{wew}

1.3) Parametry modelu FOTD

Tabela 1 Parametry modelu FOTD uzyskane za pomocą metody stycznej

Metoda stycznej					
	Δk	Δu	k	t_o	t
T_p	3,000	1000,000	0,003	540,400	1373,600
T_{wew}	4,000	1000,000	0,004	502,700	380,300

Tabela 2 Parametry modelu FOTD uzyskane za pomocą metody dwupunktowej

Metoda dwupunktowa								
	Δx	Δu	k	$0.283 \cdot \Delta x$	$0.632 \cdot \Delta x$	t_o	t	T_0/T
T_p	3,000	1000,000	0,003	10,850	11,132	689,000	850,800	0,447
T_{wew}	4,000	1000,000	0,004	11,896	12,528	326,700	850,800	0,616

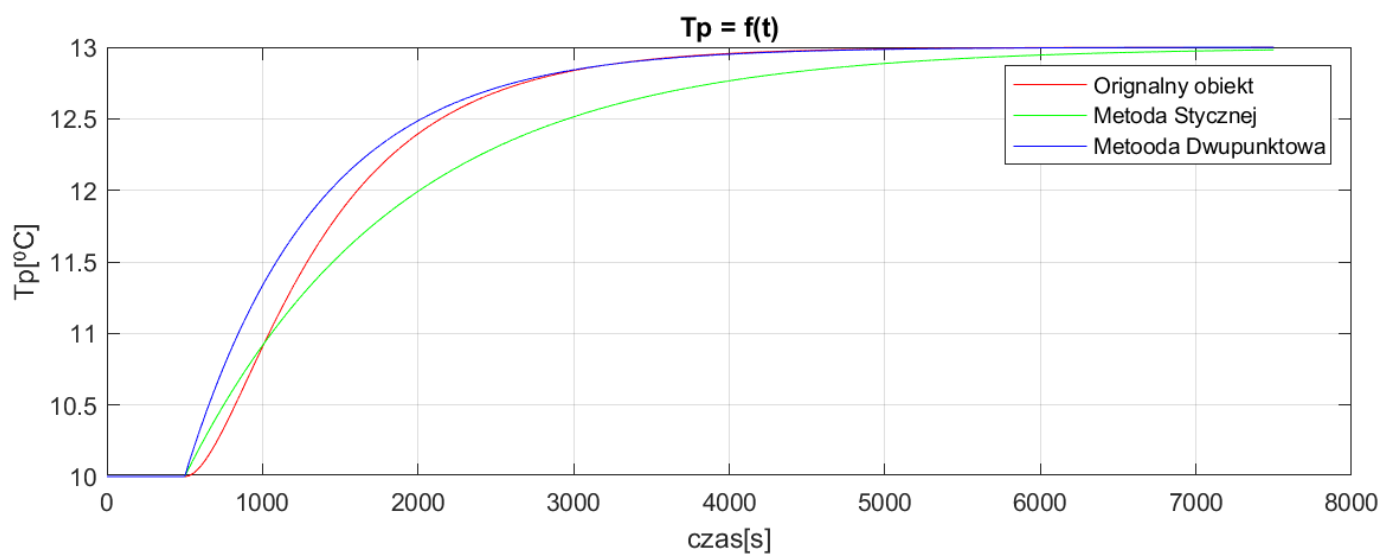
1.4) Wartości transmitancji w zależności od metody

Od parametru T_0 odjęto czas pojawienia się skoku

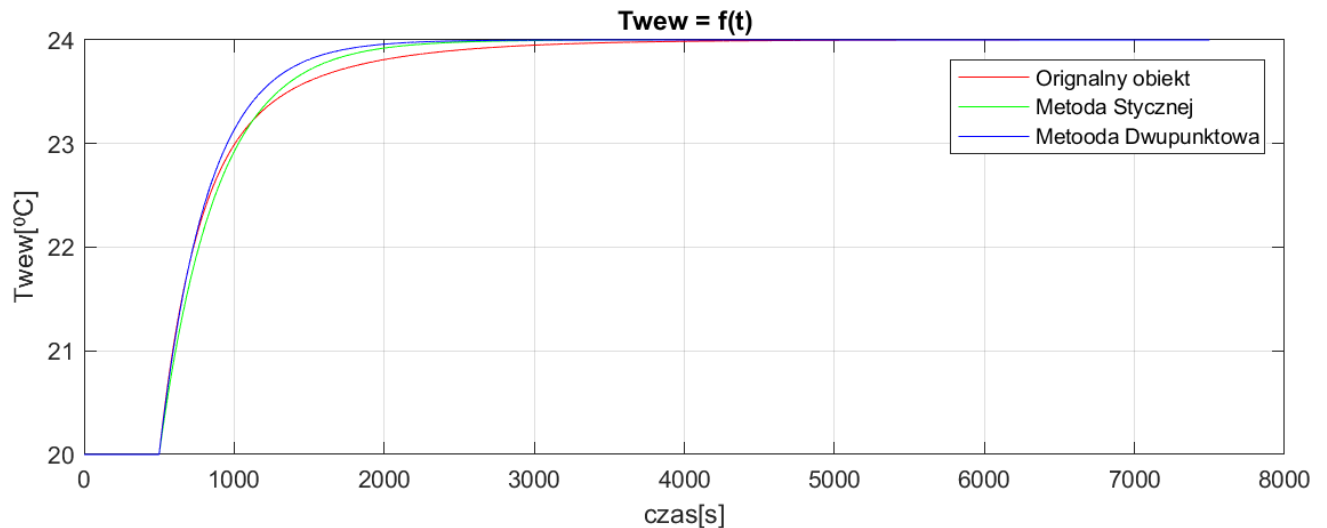
Tabela 3 Wartości transmitancji w zależności od metody

	Metoda Stycznej	Metoda dwupunktowa
T_p	$\frac{0.003}{1373.6s + 1} e^{-s40.4}$	$\frac{0.004}{Ts + 1} e^{-s689}$
T_{wew}	$\frac{0.003}{380.3s + 1} e^{-s502.7}$	$\frac{0.004}{Ts + 1} e^{-s518.8}$

2) Weryfikacja



Wykres 5 wykres $T_p = f(t)$ porównania metod identyfikacji z rzeczywistym obiektem

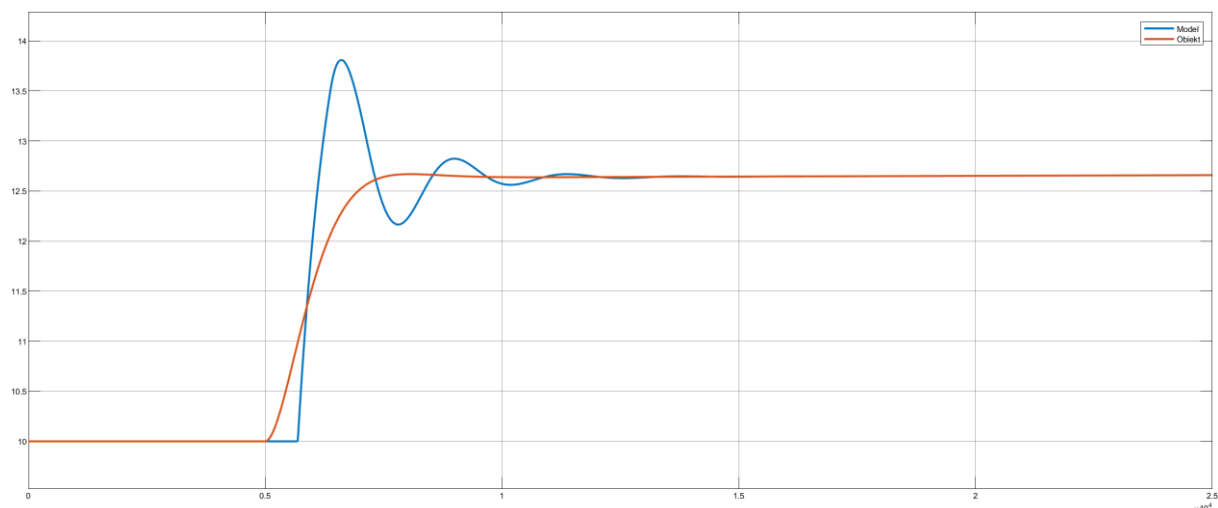


Wykres 6 wykres $T_{wew} = f(t)$ porównania metod identyfikacji z rzeczywistym obiektem

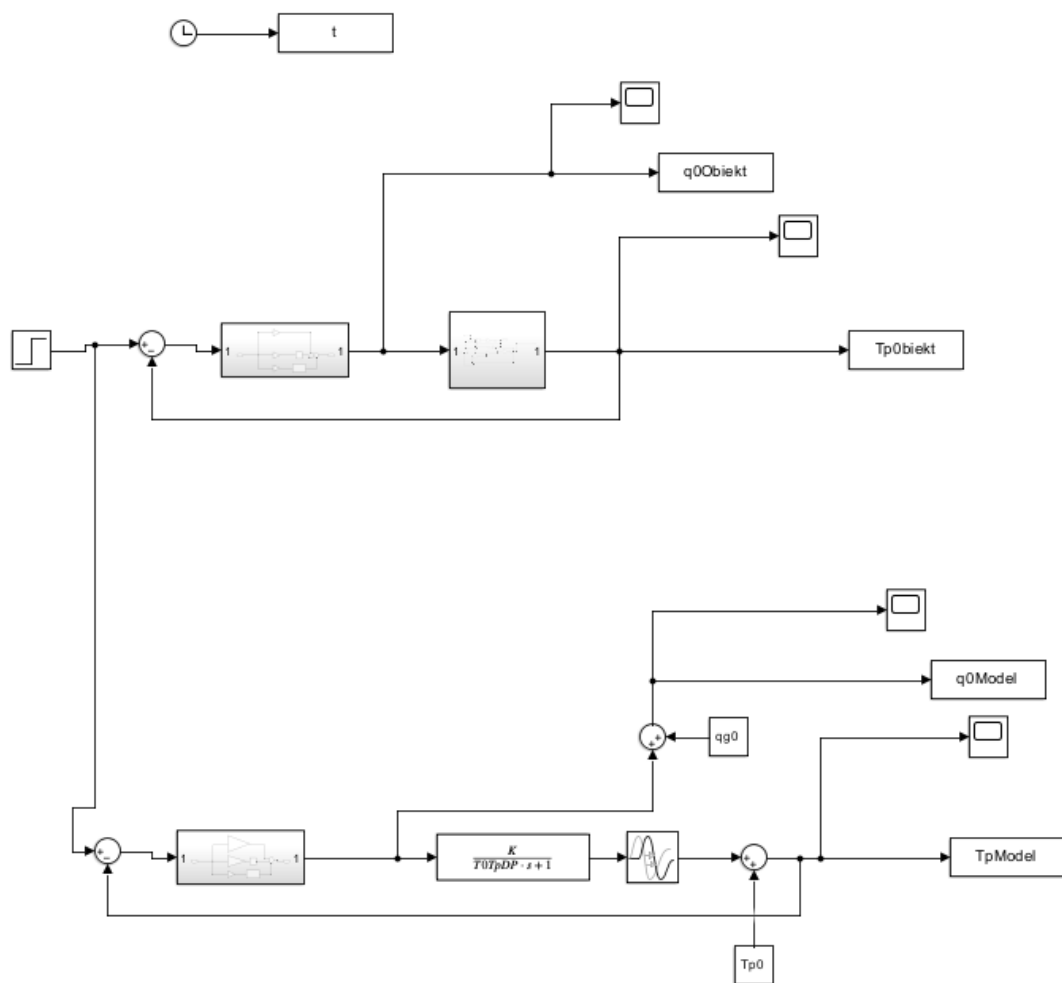
- Na podstawie wykresów 5 i 6 stwierdzono, że metoda dwupunktowa w bardziej poprawny sposób odzwierciedla reakcje na skok oryginalnego obiektu. Widoczne jest to w szczególności na wykresie 5 gdzie wyjście metody dwupunktowej odbiega w mniejszym stopniu od metody stycznej. Na wykresie 6 ten sam efekt nie jest aż, tak widoczny jednak po wystąpieniu skoku wyjście metody dwupunktowej pokrywa się bardziej z oryginalnym obiektem.

W dalszej części sprawozdania

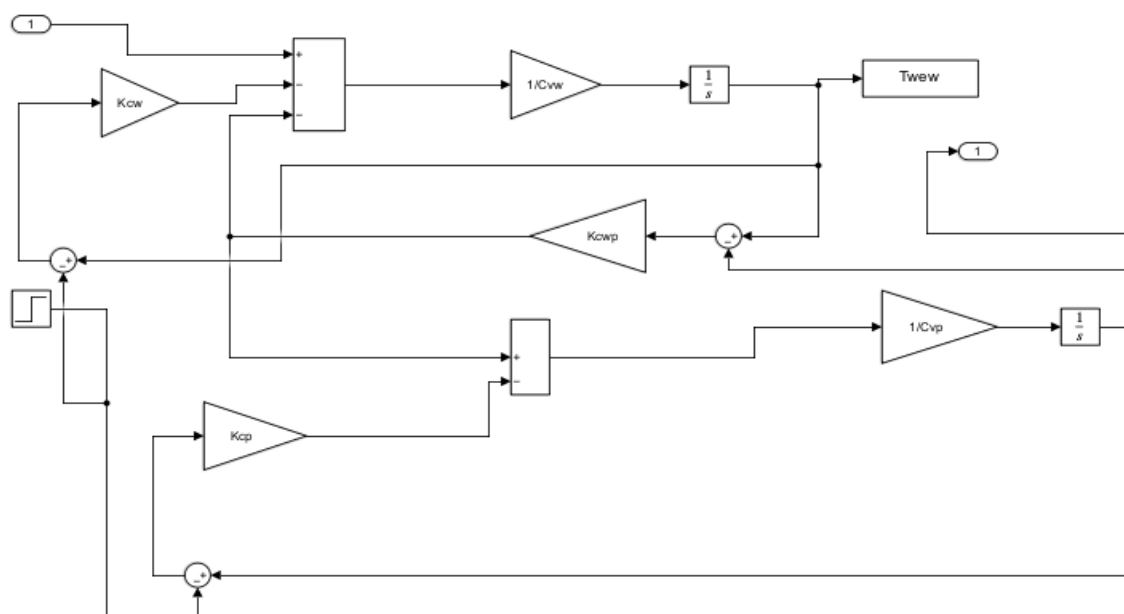
3) Wprowadzenie układu regulacji – na modelu (URM) i na obiekcie (URO) – schemat



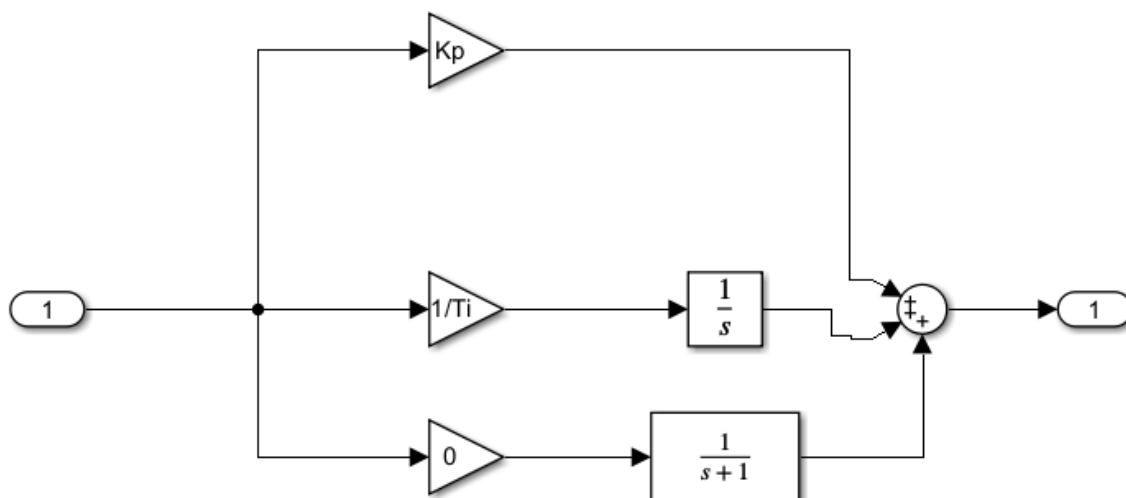
Wykres 7 Przykładowy Rysunek Scope $T_pModel = f(t)$



Rys 1 Schemat wprowadzenia układu regulacji



Rys 2 Wnętrze subsystemu Obiektu Wprowadzenia układu regulacji



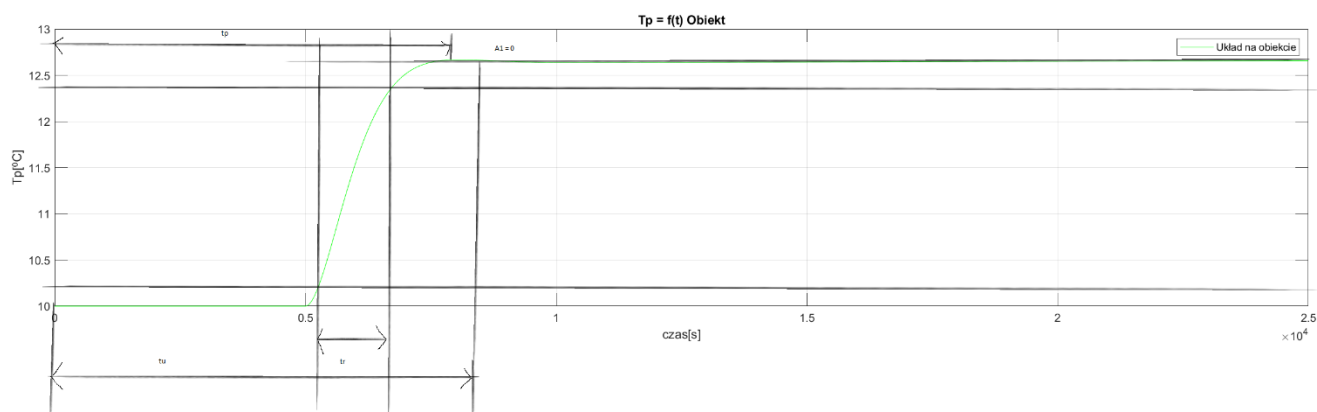
Rys 3 Wnętrze subsystemu regulatora PID Wprowadzenia układu regulacji

4) Nastawy obliczone metodą Zieglera-Nicholsa

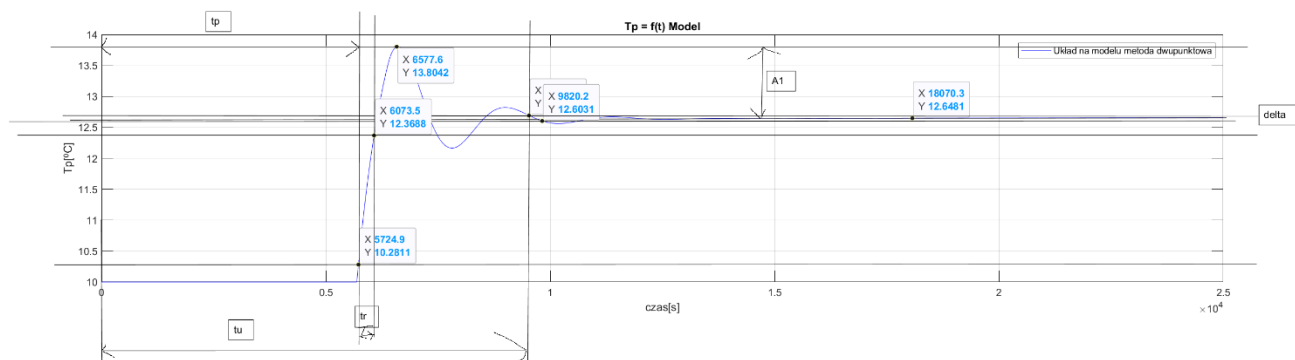
Tabela 4 Obliczone nastawy za pomocą metody Zieglera-Nicholsa oraz parametry potrzebne do ich obliczenia

	T0	T	K	Kp	Ti	Ki
Metoda Stycznej	540,4000	1374,0000	0,0030	763.3333	1,798E+03	5,5561E-04
Metoda Dwupunktowa	689,0000	850,0000	0,0030	369,5652	2,298E+03	4,3522E-04

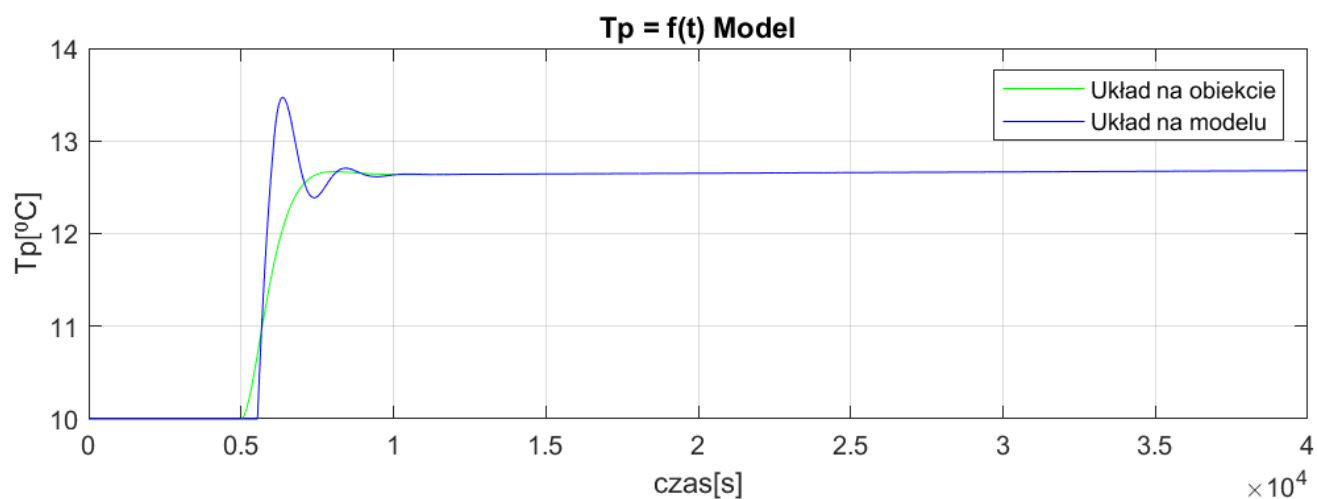
5) Porównanie URM i URO



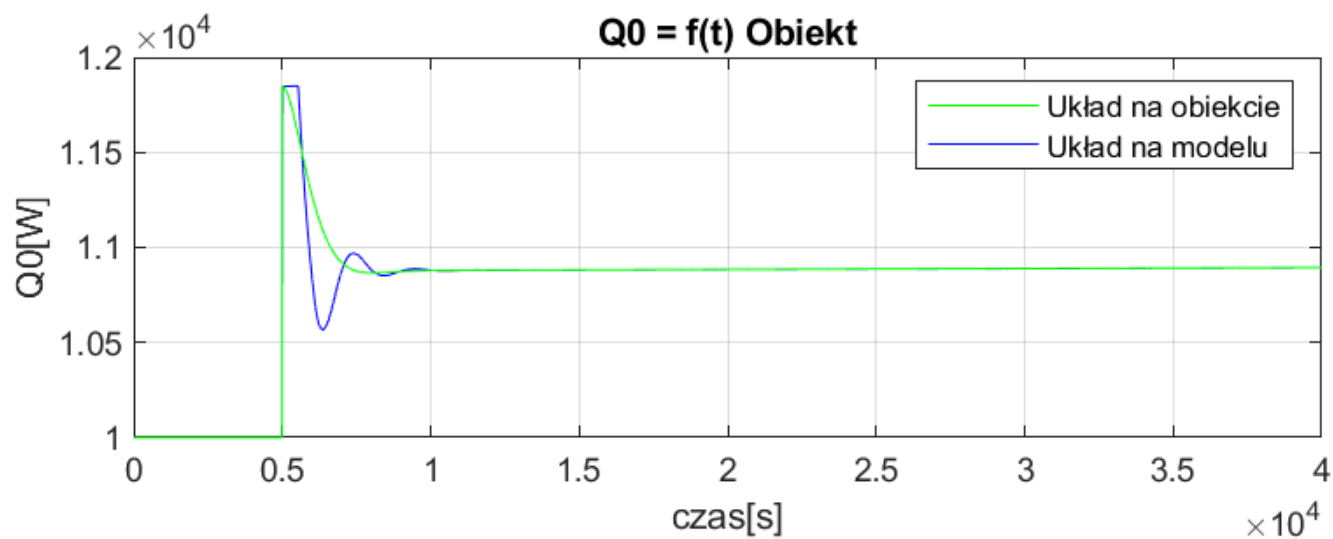
Wykres 8 Odczytane wskaźniki jakości Tp obiekt



Wykres 9 Odczytane wskaźniki jakości model metoda dwupunktowa



Wykres 10 URM i URO porównanie na jednym wykresie



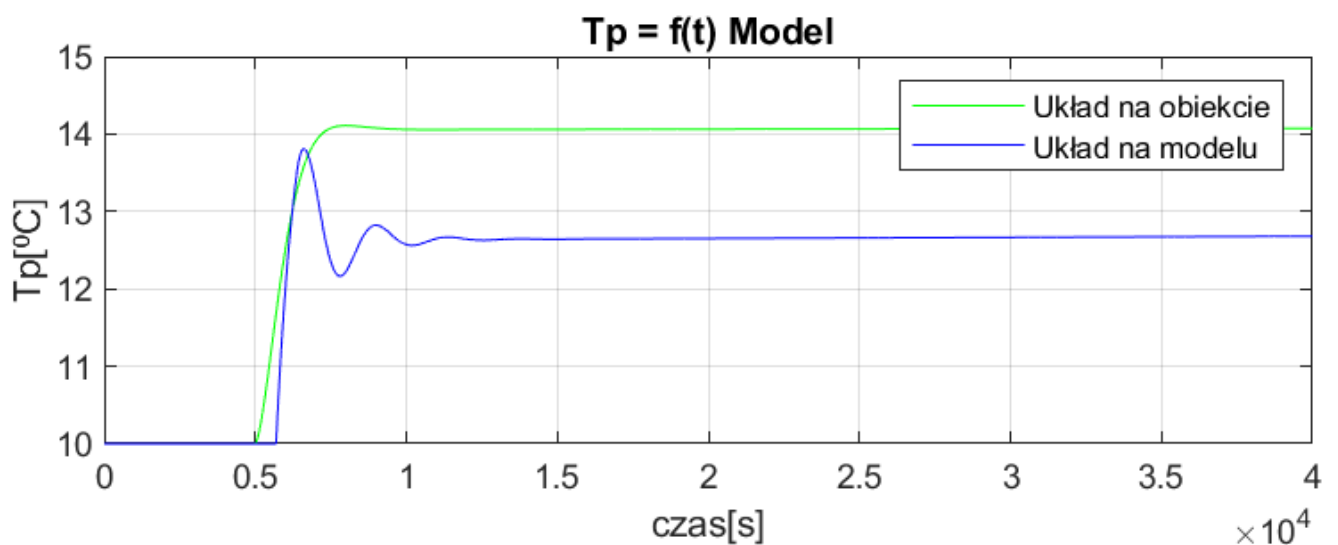
Wykres 11 Porównanie wielkości sterującej na jednym wykresie URO i URM

Tabela 5 Tabela wartości odczytanych wskaźników jakości w zależności

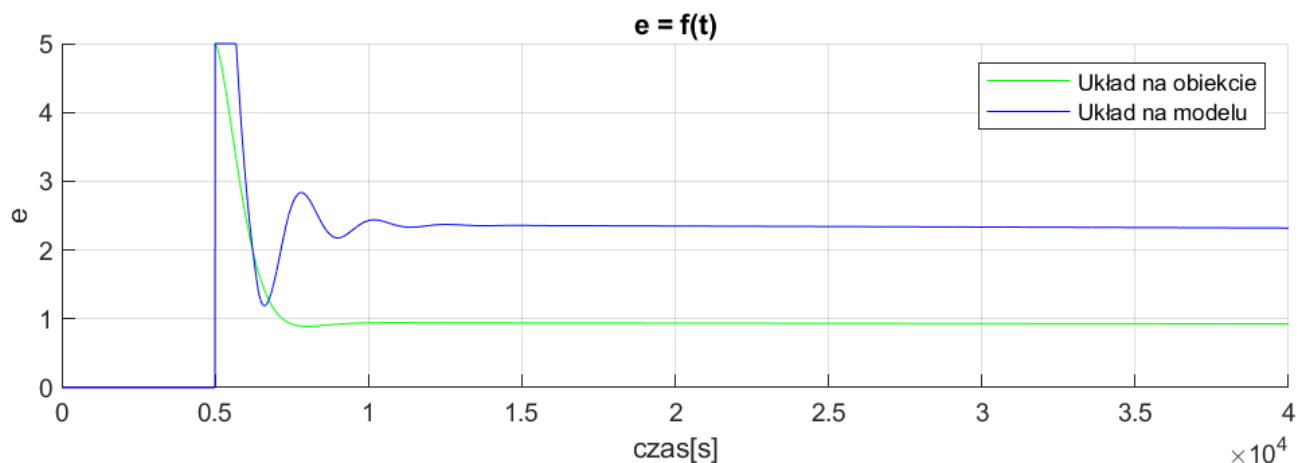
t_skok = 5000						
t_simulation = 25000	A1	tr	tp	tu	Mp	lise
Obiekt	0,21	1860	8128,6	9400	7,777	1,27E+05
Model Dwupunktowa	1,1561	348,6	6577,6	9526,2	43,6264	1,26E+05

- Czas regulacji, czas pierwszego przeregulowanie są znacznie wyższe w przypadku regulacji obiektu.
- Przeregulowanie jest znacznie wyższe w przypadku regulowanie modelu opartego na metodzie dwupunktowej.
- Wartości wskaźnika całki kwadratowej mają bardzo podobną wielkość.
- Na powyższych wykresach (10,11) widoczne jest to, że regulator lepiej działa na obiekcie. Może być to spowodowane pewną aproksymacją modelu uzyskanego za pomocą metody dwupunktowej.

6) Ocena jakości URO

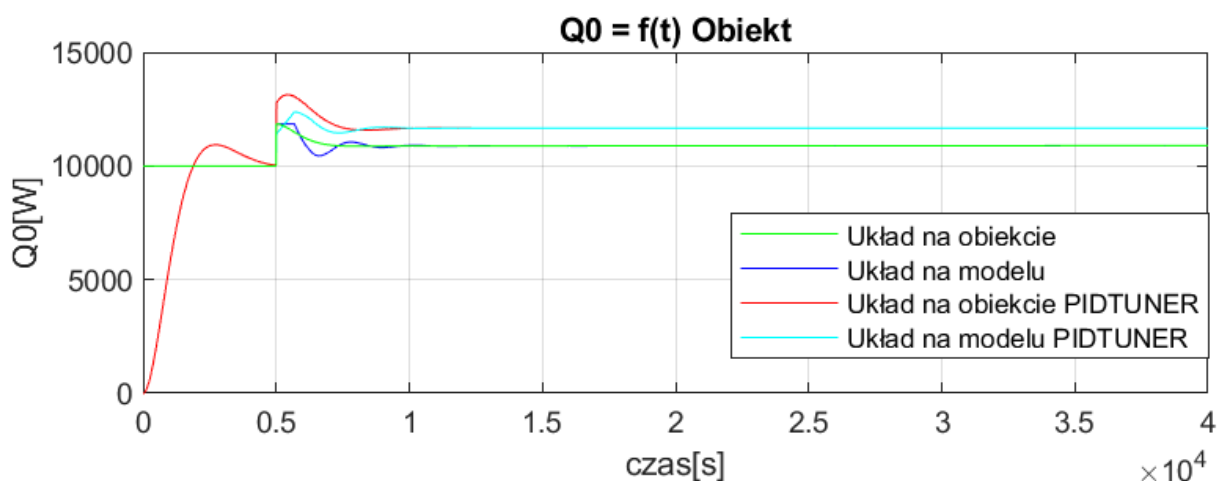


Wykres 12 Skok SP wykres PV

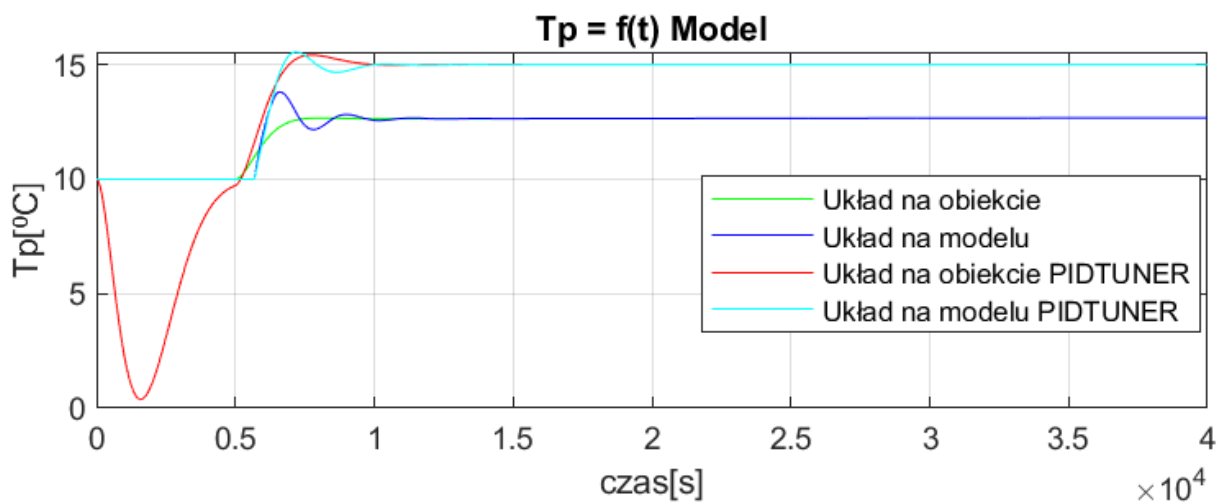


Wykres 13 Skok SP wykres uchybu

7) Porównanie różnych metod doboru nastaw – badania na URM i URO



Wykres 14 Porównanie Q_0 na obiekcie i modelu z wykorzystaniem nastaw z PIDTUNER i bez



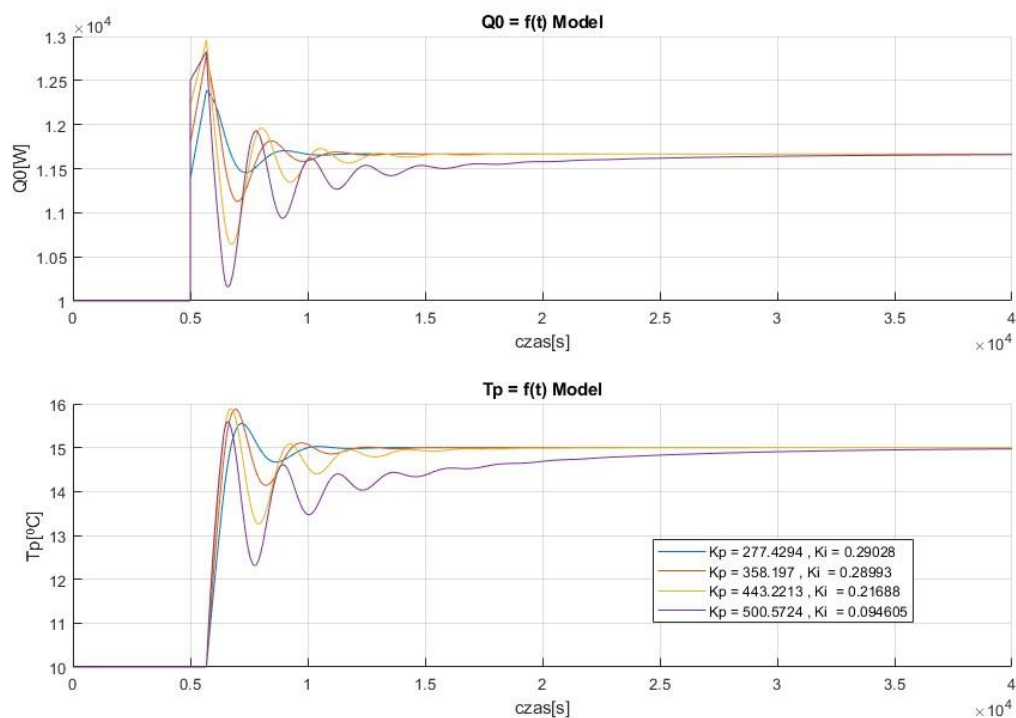
Wykres 15 Porównanie T_p na obiekcie i modelu z wykorzystaniem nastaw z PIDTUNER i bez

Tabela 6 Nastawy, Wskaźniki jakości, zapasy dla obliczonego regulatora oraz PID TUNERA

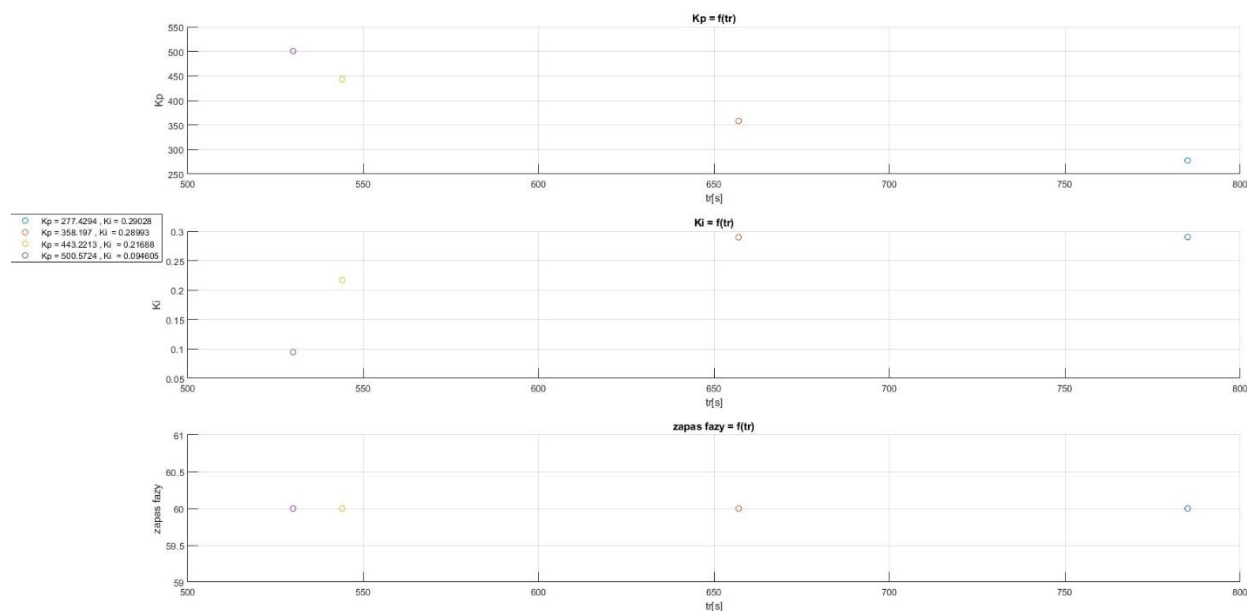
	Ziegler-Nichols		PID TUNER	
Nastawy	Kp	369,5652	Kp	544,8728
	Ki	4,3522E-04	Ki	0,42897
Wskaźniki jakości	tr	1860	tr	1,27E+03
	tu	9400	tu	4,33E+03
	Mp	7,777	Mp	9,33
	A1	0,21	A1	1,09
	lise	1,27E+05	lise	1,78E+05
Zapasy	$\Delta\phi$	94,1	$\Delta\phi$	60
	$\Delta\lambda$	23,8	$\Delta\lambda$	inf

- Na wykresie 15 widoczny jest brak stanu ustalonego dla obiektu z pid tunera oznacza to błąd. Natomiast dla modelu nie zachodzi takie zjawisko i wszystko wygląda prawidłowo.
- Nastawy pochodzące od PIDTUNERA mają inną wartość a wskaźniki jakości takie jak czas narastanie oraz regulacji mają lepszą wartość gdyż są krótsze.
- Przeregulowanie oraz amplituda lepsze są dla obliczonych nastaw.
- Natomiast zapasy mają lepszą wartość dla nastaw pochodzących z PIDTUNERA.
- Wartości końcowe dla nastaw z PIDTUNERA różnią się o pewną wartość od nastaw uzyskanych za pomocą metody dwupunktowej.

8) Związek pomiędzy stabilnością i jakością regulacji



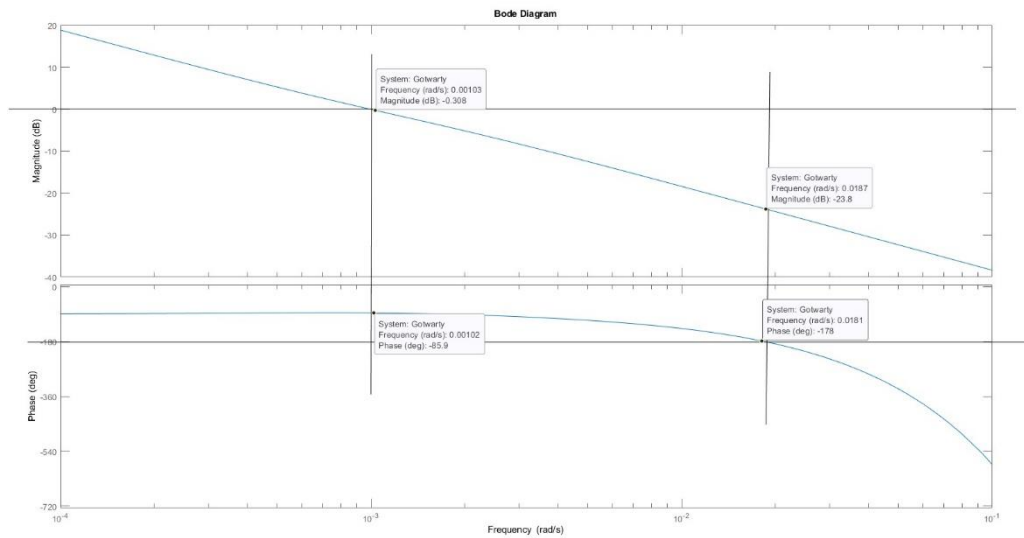
Rys 4 Wykresy PV w zależności od nastaw PID TUNERA



Rys 5 Wykresy zależności między nastawami a czasami narastania oraz między zapasem fazy a czasem narastania

- Na pierwszych dwóch wykresach na rysunku 5 widoczna jest to że im wyższa wartość nastawy K_p tym mniejszy czas regulacji badanego obiektu.
- Wraz ze wzrostem nastawy K_i można zauważyć że wraz ze zwiększeniem jej wartości wzrasta też czas regulacji obiektu.
- Nie stwierdzono zależności pomiędzy zapasem fazy a czasem regulacji gdyż zapas fazy ma cały czas tą samą niezmienną wartość

9) Dodatek



Kod programu

```
close all;  
clear;
```

```
a = 0.25;
```

```
TzewN = -20;  
TwewN = 20;  
TpN = 10;  
qgN = 10000;  
Tzad = TpN;
```

```
a_p = 5;  
b_p = 5;  
c_p = 5;  
Vw = a_p * b_p * c_p;
```

```
Vp = 0.6*Vw;
```

```
cp = 1000;  
rop = 1.2;
```

```

Kcw = qgN/(TwewN * (1 + a) - TzewN - a * TpN);
Kcp = (qgN*a*(TwewN-TpN))/((TwewN * (1 + a) - TzewN - a*TpN) * (TpN - TzewN));
Kcwp = a*Kcw;

Cvw = cp * rop * Vw;
Cvp = cp * rop * Vp;

zmianaTzew = 0;
procent_zmiany_qg = 1;
qg0 = qgN;
Tzew0 = TzewN;
dqg = 0.1*qgN;
Twew0 = (qg0*(Kcwp + Kcp) + Tzew0*(Kcw*Kcwp + Kcw * Kcp + Kcwp * Kcp))/(Kcw*(Kcwp
+ Kcp) + Kcwp * Kcp);
Tp0 = ((Kcw + Kcwp)*Kcp*Tzew0 + Kcwp*(qg0 + Kcw*Tzew0))/(Kcw*(Kcwp + Kcp) +
Kcwp*Kcp);
qg0 =Twew0*(Kcw+Kcwp)-Tp0*Kcwp-Kcw*((Tp0*(Kcp+Kcwp)-Twew0*Kcwp)/Kcp);

M = [(Cvw*Cvp) (Kcwp*Cvp + Kcp*Cvw + Kcw*Cvp + Kcwp*Cvp) (Kcw*Kcwp + Kcw*Kcp +
Kcp*Kcwp)];

L11 = [ Cvp (Kcwp + Kcp)];
L21 = Kcwp;
L12 = [ (Cvp*Kcw) (Kcw*Kcwp + Kcp*Kcw + Kcwp*Kcp)];
L22 = [ (Cvw*Kcp) (Kcw*Kcp + Kcp*Kcwp + Kcwp*Kcw)];

t_simulation = 40000;
t0 = 5000;

dgq = 0;
%dgq = 0;
dTzew = 0;
dTzad = 5;

Lm11 = L11;
Lm21 = L21;

Mm = M;

T0TpDP = 690;
TTpDP = 850;
K0TpDP = 0.003;

T0TwewDP = 520;
TTwewDP = 327;
K0TwewDP = 0.004;

LTwew11DP = K0TwewDP;
MTwew11DP = [TTwewDP 1];

```

```

LTp11DP = K0TpDP;
MTp11DP = [TTpDP 1];

T0Tp = 540;
TTp = 1374;
K0Tp = 0.003;
trasport_delay = 540.4;
trasport_delayDP = 689;

T0Twew = 502;
TTwew = 381;
K0Twew = 0.004;

K = 0.003
Kp = (0.9*TTpDP)/(K*T0TpDP)
Ti = 3.33 * T0TpDP
tp = 3.33 * 189
Ki = 1/Ti

%K = 0.003
%Kp = (0.9*TTp)/(K*T0Tp)
%Ti = 3.33 * T0Tp
%tp = 3.33 * 40.4
%Ki = 1/Ti

KpTunerObjekt = 544.8728;
KiTunerObjekt = 0.42897;
KpTunerModel = 277.4294;
KiTunerModel = 0.29028;

LTwew11 = K;
MTwew11 = [TTwewDP 1];

LTp11 = K;
MTp11 = [TTpDP 1];

sim("termin7simulink.slx");

figure();

hold on;
subplot(211)
plot(t,q00biekt,"g");
grid on
title("Q0 = f(t) Obiekt");
ylabel("Q0[W]");
xlabel("czas[s]");

hold on;
subplot(212)
plot(t,Tp0biekt,"g");
title("Tp = f(t) Obiekt");
ylabel("Tp[°C]");

```

```

xlabel("czas[s]");
hold on;
grid on;

hold on;
subplot(211)
plot(t,q0Model,"b");
grid on
title("Q0 = f(t) Model");
ylabel("Q0[W]");
xlabel("czas[s]");

hold on;
subplot(212)
plot(t,TpModel,"b");
title("Tp = f(t) Model");
ylabel("Tp[°C]");
xlabel("czas[s]");
grid on;

hold on;
subplot(211)
plot(t,q0ObiektPIDTUNER,"r");
grid on
title("Q0 = f(t) Obiekt");
ylabel("Q0[W]");
xlabel("czas[s]");

hold on;
subplot(212)
plot(t,TpObiektPIDTUNER,"r");
title("Tp = f(t) Obiekt");
ylabel("Tp[°C]");
xlabel("czas[s]");
hold on;
grid on;

hold on;
subplot(211)
plot(t,q0ModelPIDTUNER,"c");
grid on
title("Q0 = f(t) Model");
ylabel("Q0[W]");
xlabel("czas[s]");

hold on;
subplot(212)
plot(t,TpModelPIDTUNER,"c");
title("Tp = f(t) Model");
ylabel("Tp[°C]");
xlabel("czas[s]");
grid on;

legend("Układ na obiekcie","Układ na modelu","Układ na obiekcie PIDTUNER","Układ
na modelu PIDTUNER")

hold on;
subplot(211)

```



```

plot(t,q0Obiekt,"g");
grid on
title("Q0 = f(t) Obiekt");
ylabel("Q0[W]");
xlabel("czas[s]");

legend("Układ na obiekcie","Układ na modelu","Układ na obiekcie PIDTUNER","Układ
na modelu PIDTUNER")

figure(2)

hold on;

plot(t,e_obiekt,"g");
title("Tp = f(t) Obiekt");
ylabel("Tp[°C]");
xlabel("czas[s]");
hold on;
grid on;

hold on;

plot(t,e_model,"b");
grid on
title("e = f(t)");
ylabel("e");
xlabel("czas[s]");

legend("Układ na obiekcie","Układ na modelu")
%legend("Układ na modelu metoda dwupunktowa")

figure(3)

KpTunerModelloop = [277.4294 , 358.197 , 443.2213, 500.5724];
KiTunerModelloop = [0.29028, 0.28993 , 0.21688 , 0.094605];

for i = 1:length(KpTunerModelloop)
KpTunerModel = KpTunerModelloop(i);
KiTunerModel = KiTunerModelloop(i);

sim("termin7simulink.slx");

subplot(211)
hold on;
plot(t,q0ModelPIDTUNER);
grid on
title("Q0 = f(t) Model");
ylabel("Q0[W]");
xlabel("czas[s]");

subplot(212)
hold on;
plot(t,TpModelPIDTUNER);

```

```

title("Tp = f(t) Model");
ylabel("Tp[°C]");
xlabel("czas[s]");
grid on;

end

legend("Kp = 277.4294 , Ki = 0.29028", "Kp = 358.197 , Ki = 0.28993", "Kp = 443.2213 , Ki = 0.21688", "Kp = 500.5724 , Ki = 0.094605")

figure()
t_raise = [785 , 657, 544, 530];
zapas_fazy = [60 , 60, 60, 60];

figure(4)
for i = 1:length(KpTunerModelloop)

figure(4)

subplot(311)
hold on;
plot(t_raise(i),KpTunerModelloop(i),"o");
grid on
title("Kp = f(tr)");
ylabel("Kp");
xlabel("tr[s]");

subplot(312)
hold on;
plot(t_raise(i),KiTunerModelloop(i),"o");
grid on
title("Ki = f(tr)");
ylabel("Ki");
xlabel("tr[s]");

subplot(313)
hold on;
plot(t_raise(i),zapas_fazy(i),"o");
title("zapas fazy = f(tr)");
ylabel("zapas fazy");
xlabel("tr[s]");
grid on;

end
legend("Kp = 277.4294 , Ki = 0.29028", "Kp = 358.197 , Ki = 0.28993", "Kp = 443.2213 , Ki = 0.21688", "Kp = 500.5724 , Ki = 0.094605")

%s = tf('s');

%G = (K/(T0TpDP*s + 1))*exp(-86*s);

```

```
%R = KpTunerModel + KiTunerModel/s;
```

```
%Gotwarty = G*R;
```

```
%figure()  
%nyquist(Gotwarty)  
%th = 0:pi/50:2*pi;  
%xunit = cos(th) + pi/2;  
%yunit = sin(th) + pi/2;  
%hold on  
%plot(xunit,yunit);
```

```
%figure()  
%bode(Gotwarty)  
%th = 0:pi/50:2*pi;  
%xunit = cos(th) + pi/2;  
%yunit = sin(th) + pi/2;  
%hold on  
%plot(xunit,yunit);
```