

Układy Sterowania Inteligentnego.

Maciej Cebula
Marcin Kowalczyk
Daniel Rubak

Spis treści

1	Wstęp	2
1.1	Cel zajęć	2
1.2	Obiekt sterowania	2
1.3	Wskaźniki jakości	2
2	Model matematyczny	4
2.1	Model silnika elektrycznego prądu stałego	4
2.2	Model matematyczny obiektu	4
3	Regulator klasyczny - PID	7
3.1	Regulator PID	7
4	Regulator neuronowy	9
4.1	Optymalizacja nastaw regulatora	12
4.2	Porównanie wskaźników jakości	14
4.3	Projektowanie regulatora neuronowego z użyciem Neural Toolbox.	14
5	Regulator rozmyty	16
5.1	Pierwszy zestaw funkcji przynależności	17
5.2	Drugi zestaw funkcji przynależności	18
5.3	Porównanie	21
5.4	Regulator Takagi - Sageno	22
5.4.1	Manualny dobór struktury regulatora	22
5.4.2	Optymalizacja struktury regulatora	23
5.4.3	Porównanie	25
6	Regulator rozmyty dla modelu helikoptera	27
6.1	Wstęp	27
6.2	Model matematyczny	27
6.3	Regulator referencyjny	28
6.4	Regulator rozmyty	29
7	Wnioski	33

Wstęp

1.1 Cel zajęć

Celem projektu wykonywanego w ramach zajęć z przedmiotu *Układy Sterowania Inteligentnego* było zaprojektowanie regulatora dla manipulatora, którego zadaniem było zabranie szklanki z wodą z jednego miejsca i odstawienie jej w innym miejscu. Przemieszczał się on w płaszczyźnie poziomej. Napędzany był jednym silnikiem prądu stałego.

Projektowany regulator miał być układem sterowania inteligentnego. W ramach tego projektu należało również przeprowadzić porównanie inteligentnych algorytmów sterowania (np. sieć neuronowa lub regulator fuzzy) z regulatorami klasycznymi (np. PID, LQ lub czasoop-tymalny).

1.2 Obiekt sterowania

Obiektem sterowania był manipulator, który przynosił szklankę z wodą, a następnie powracał do położenia początkowego. Musiał on przenieść szklankę bez wylewania jej zawartości. Takie zadanie nie jest tożsame z samym pozycjonowaniem manipulatora. Wymaganie, by nie wylać wody narzuca ograniczenia na ruch obiektu. Musi poruszać się on z odpowiednio małym przyspieszeniem, gdy trzyma szklankę z wodą. Ograniczenie to nie jest jednak ważne, gdy powraca do położenia początkowego. Jest więc oczywiste, że dla ruchu w obie strony powinny zostać użyte inne regulatory. Pierwszy z regulatorów powinien zapewnić spełnienie następującego ograniczenia:

$$\ddot{\phi}(t) \leq \epsilon_{max} \quad (1.1)$$

gdzie:

ϕ jest położeniem kątowym manipulatora,

ϵ_{max} jest maksymalnym przyspieszeniem kątowym.

1.3 Wskaźniki jakości

Aby móc porównać ze sobą różne struktury regulatorów konieczne było zdefiniowanie wskaźników jakości, które minimalizować miał projektowany regulator. Zdecydowano, że będą one następującej postaci:

1. całka z kwadratu uchybu regulacji $J_1 = \int_0^{t_k} e(t)^2 dt \text{ [rad}^2 \cdot \text{s]}$

2. wskaźnik energetyczny $J_2 = \int_0^{t_k} u(t)^2 dt$ [$V^2 \cdot s$]
3. suma powyższych wskaźników $J_3 = J_1 + J_2$

(1.2)

gdzie:

$e(t) = r - \phi(t)$ to uchyb regulacji

ϕ jest położeniem kątowym manipulatora,

r jest zadaną pozycją manipulatora,

u jest sterowaniem podawanym na obiekt.

Należy zwrócić uwagę, że postanowiono zaniedbać opory ruchu. W związku z tym jedynym momentem siły działającym na manipulator był moment pochodzący od silnika prądu stałego. W dalszej części projektu możliwe jest zmodyfikowanie zadania w taki sposób, by wziąć pod uwagę opory związane z ruchem obrotowym manipulatora.

Model matematyczny

2.1 Model silnika elektrycznego prądu stałego

Silnik elektryczny odpowiedzialny za poruszanie ramieniem został zamodelowany jako obiekt inercyjny pierwszego rzędu. Równania 2.1 i 2.2 opisują zależność generowanego momentu obrotowego od prądu.

$$M(t) = k_e \cdot i(t) \quad (2.1)$$

$$U(t) = i(t) \cdot R + L \cdot \frac{di(t)}{dt} \quad (2.2)$$

gdzie:

$M(t)$ - moment generowany przez silnik,

$i(t)$ - prąd elektryczny,

k_e - stała elektryczna silnika,

L - indukcyjność silnika,

R - opór elektryczny silnika.

2.2 Model matematyczny obiektu

Równania mechaniczne opisujące dynamikę całego układu mają postać:

$$\frac{d^2\alpha(t)}{dt^2} \cdot J = k_e \cdot i(t) \quad (2.3)$$

$$U(t) = i(t) \cdot R + \frac{di(t)}{dt} \cdot L \quad (2.4)$$

gdzie:

α - kąt wychylenia,

J - moment bezwładności ramienia,

$U(t)$ - napięcie podawane na silnik,

W przyjętym modelu obiektu założono że wielkością sterującą jest napięcie podawane na silnik, a wyjściową kąt wychylenia ramienia.

Na podstawie równań 2.3 i 2.4 zapisano model matematyczny w postać równań stanu przyjmując następujące zmienne stanu:

x_1 - prąd silnika

x_2 - położenie katowe ramienia

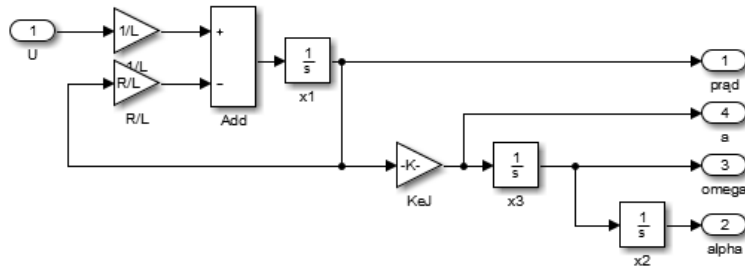
x_3 - prędkość kątowna ramienia

$$\dot{x}_1 = -x_1 \cdot \frac{R}{L} + \frac{U}{L} \quad (2.5)$$

$$\dot{x}_2 = x_3 \quad (2.6)$$

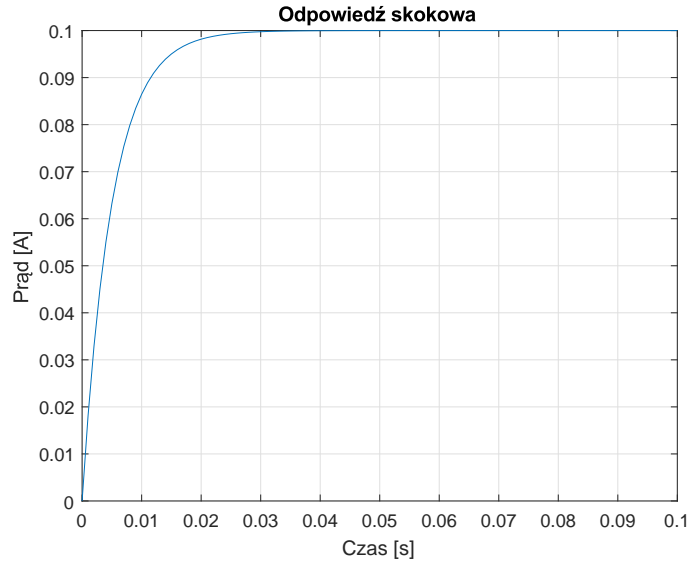
$$\dot{x}_3 = \frac{k_e}{J} \cdot x_1 \quad (2.7)$$

Schemat blokowy programu *Simulink* realizujący opisany powyższy model został przedstawiony na rysunku

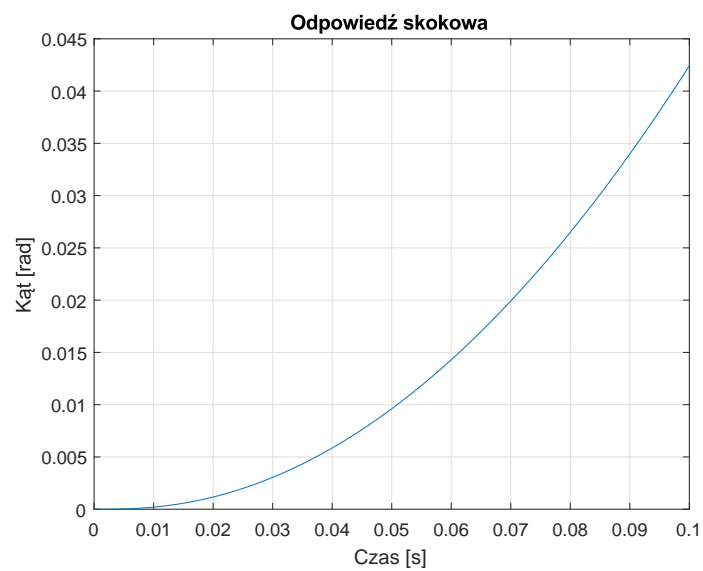


Rys. 2.1: Schemat blokowy programu *Simulink*.

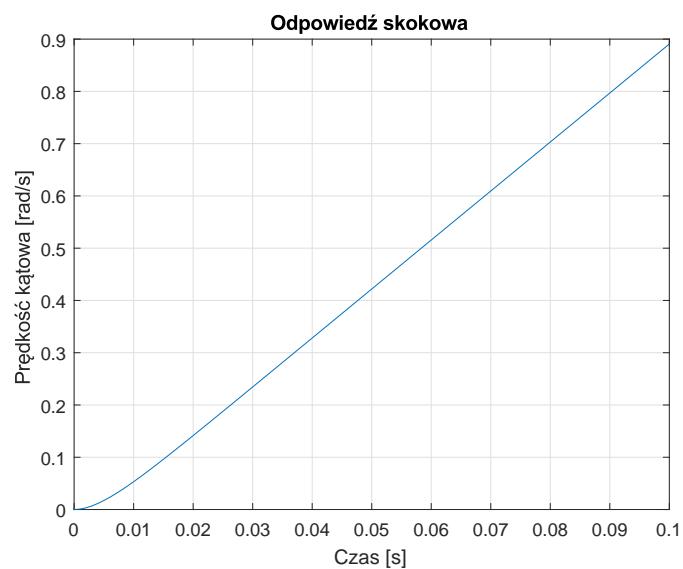
Na rysunkach przedstawiono odpowiedzi zmiennych stanu na skok napięcia.



Rys. 2.2: Odpowiedź prądu na skok napięcia.



Rys. 2.3: Odpowiedź kąta na skok napięcia.



Rys. 2.4: Odpowiedź prędkości kątowej na skok napięcia.

Regulator klasyczny - PID

3.1 Regulator PID

Do pozycjonowania manipulatora zaproponowany został regulator składający się z dwóch równolegle połączonych regulatorów PID. Kiedy trzymana jest pełna szklanka, to na wyjście przekazywane jest sterowanie z pierwszego regulatora, a kiedy jest pusta to z drugiego. Regulatorom postanowiono zadać inne nastawy, takie, by ograniczyć przyspieszenie kątowe w sytuacji, gdy trzymana jest pełna szklanka. Ma to na celu spełnienie warunku, by przyspieszenie było małe, aby nie wylać wody. Struktura obu regulatorów jest identyczna, wyrażona następującym wzorem:

$$U = (P + I\frac{1}{s} + D\frac{sN}{S + N})E \quad (3.1)$$

gdzie:

U - sterowanie

E - uchyb regulacji

P, I, D - współczynniki odpowiednio od części proporcjonalnej, całkującej i różniczkującej.

Na podstawie przeprowadzonych symulacji przyjęto następujące nastawy regulatorów:

Regulator odpowiedzialny za pozycjonowanie ramienia z napełnioną szklanką:

$$P = 3$$

$$I = 0.2$$

$$D = 1.5$$

Regulator pozycjonujący ramię z pustą szklanką:

$$P = 3$$

$$I = 0.002$$

$$D = 1$$

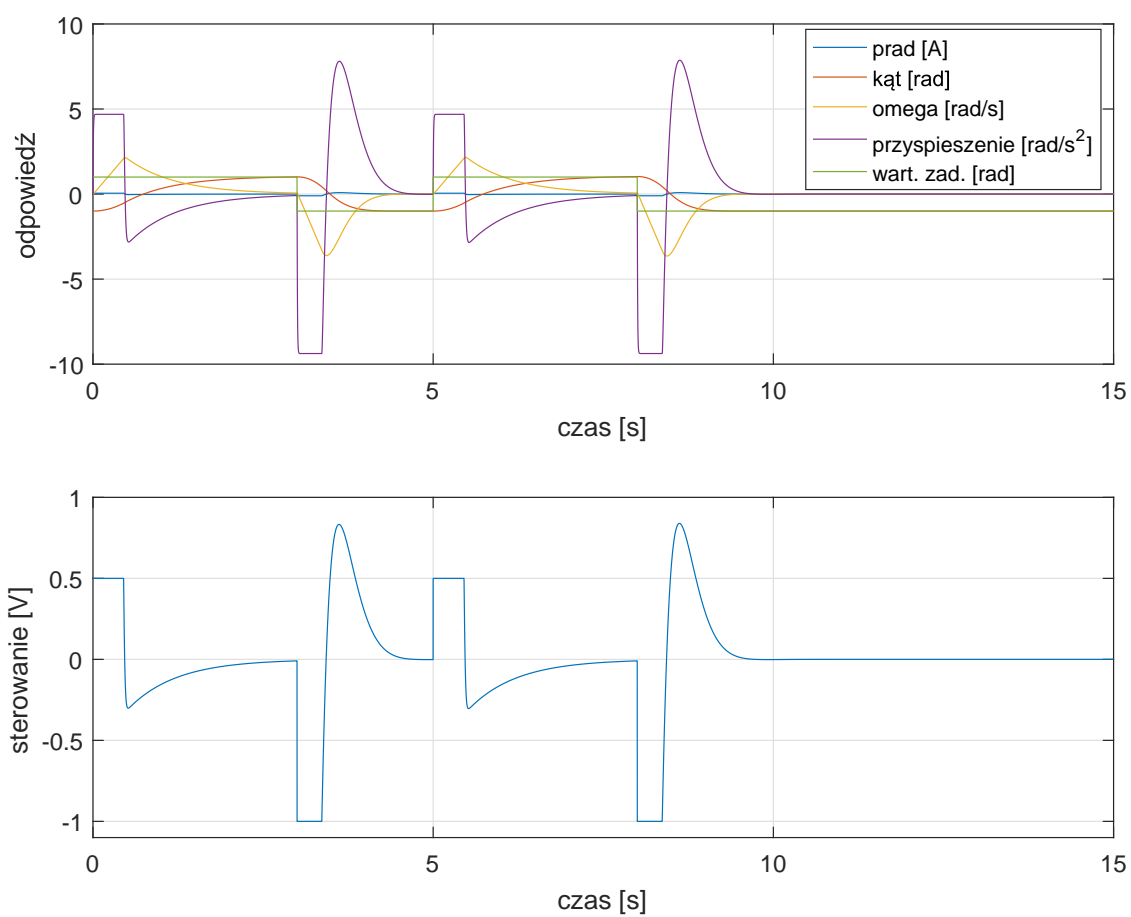
Pozycja zadana podawana na regulator manipulatora miała postać funkcji prostokątnej. Stwierdzono jednak, że z uwagi na ograniczenie przyspieszenia, czas pozycji zadanej dla ruchu z pełną szklanką powinien być dłuższy. Na tej podstawie przyjęto czas pozycjonowania ramienia z napełnioną szklanką na 3 s, a czas powrotu ramienia na pozycję początkową na 2 s.

Na rysunku 3.1 przedstawiono odpowiedź układ dla opisanych powyżej regulatorów. Dla tak przyjętych nastaw regulatorów otrzymano następujące wartości wskaźników jakości:

$$J_1 = 3.847 \text{ [rad}^2 \cdot \text{s]}$$

$$J_2 = 0.7587 \text{ [V}^2 \cdot \text{s]}$$

$$J_3 = 4.606$$



Rys. 3.1: Wartości zmiennych stanu i sterowania.

Regulator neuronowy

Regulator neuronowy został zaprojektowany w ten sposób aby otrzymać analogiczne przebiegi sygnałów jak w przypadku klasycznego regulatora PID, wykorzystując do tego celu jeden neuron. Przyjęto, że wektor sygnałów wejściowych będzie miał następującą postać:

$$x = \begin{bmatrix} \dot{e} \\ e \\ \int e \\ \frac{z - z_{min}}{z_{max} - z_{min}} \end{bmatrix} \quad (4.1)$$

gdzie:

e - uchyb regulacji

z - wart. zadana

z_{min} , z_{max} - odpowiednio minimalna i maksymalna wart. zadana

Współczynnik skalujący z racji na to, że w rozważanym przypfдку wymagane są dwa regulatory jest w postaci macierzy:

$$W = \begin{bmatrix} D_1 & P_1 & I_1 & 0 \\ D_2 & P_2 & I_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

Stała składowa dla tak przyjętej postaci regulatora jest trójelementowym wektorem:

$$b = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad (4.3)$$

Z racji na to że w zależności od tego czy przestawiamy pustą szklankę czy pełną należy zmieniać nastawy regulatora oraz saturację sygnału sterującego. Po uwzględnieniu tych wymagań przyjęto następującą postać funkcji:

$$f(u, z) = f_{sat1}((1 - z) \cdot u_1 + z \cdot u_2) \cdot (1 - z) + f_{sat2}((1 - z) \cdot u_1 + z \cdot u_2) \cdot z \quad (4.4)$$

gdzie:

z —przeskalowania wartość zadana do przedziału $[0,1]$

u_1 , u_2 —wartości sterowania odpowiednio od regulatorów dla pełnej i pustej szklanki.

f_{sat1} , f_{sat2} - funkcje saturacji dla pełnej i pustej szklanki.

Przeanalizowano dwie postaci funkcji saturacji:

1. klasyczna funkcja opisana równaniem :

$$f_{sat}(x) = \begin{cases} -K & x < y_{min} \\ x & x \in [y_{min}, y_{max}] \\ K & x > y_{max} \end{cases} \quad (4.5)$$

2. przybliżenie funkcją sigmoidalną postaci:

$$f_{sat}(x) = \left(\frac{2}{1 + \exp -\beta \cdot x} - 1 \right) \cdot K \quad (4.6)$$

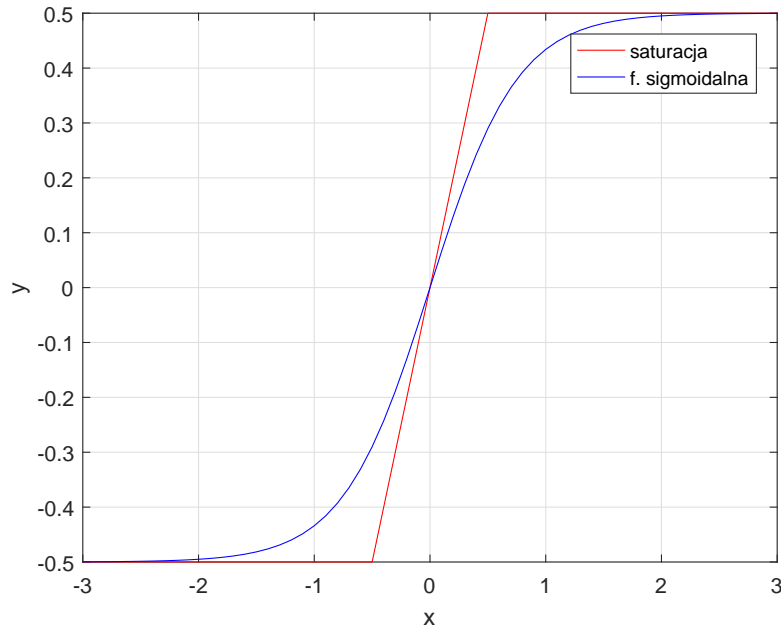
gdzie:

K - maksymalna dozwolona wartość sterowania podawanego na obiekt.

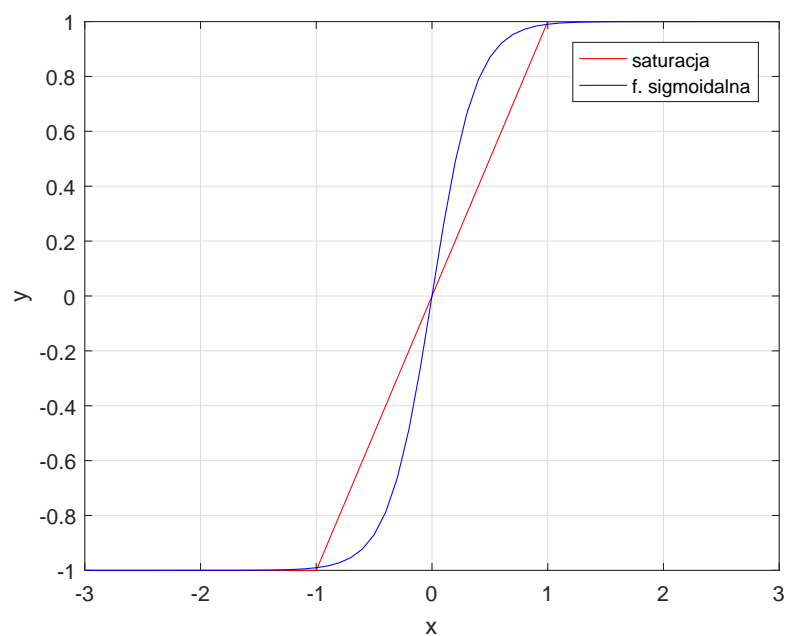
Parametry funkcji sigmoidalnych β zostały dobrane za pomocą funkcji *fmincon* tak aby zminimalizować różnice w stosunku do zależności opisanych równaniami 4.5. Finalnie otrzymano następujące wartości parametrów:

$$\beta_1 = -2.65$$

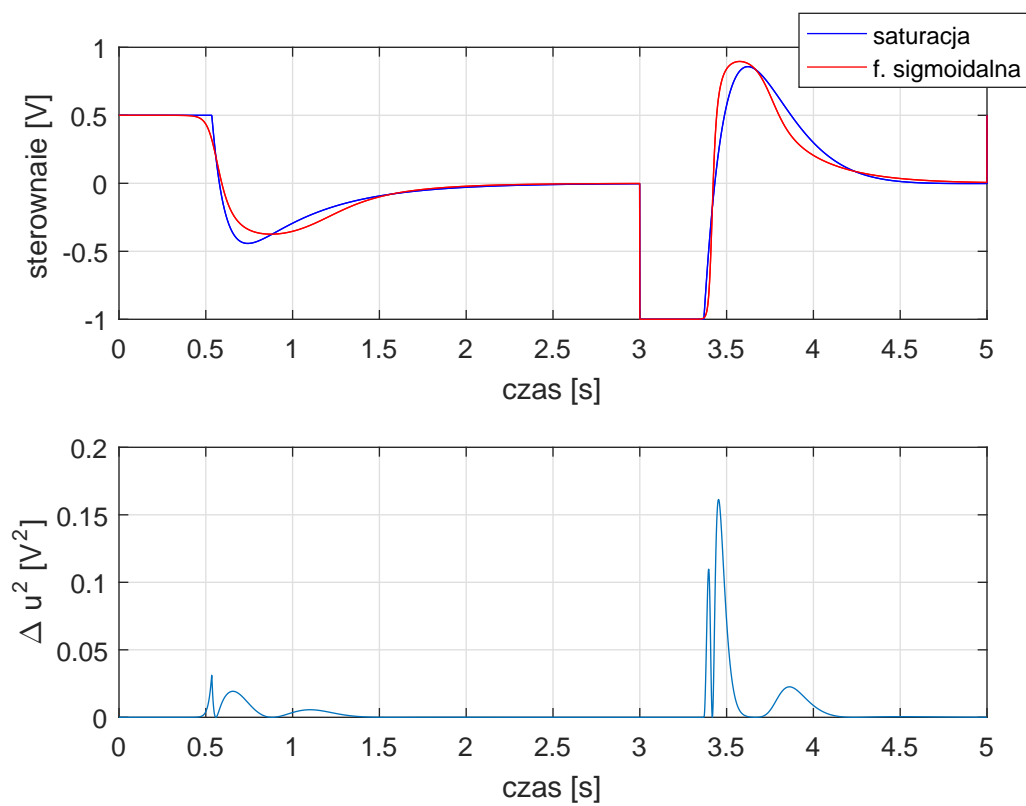
$$\beta_2 = -5.33$$



Rys. 4.1: Porównanie saturacji i funkcji sigmoidalnej - pełna szklanka.



Rys. 4.2: Porównanie saturacji i funkcji sigmoidalnej - pusta szklanka.



Rys. 4.3: Porównanie sterowania dla regulatora neuronowego.

4.1 Optymalizacja nastaw regulatora

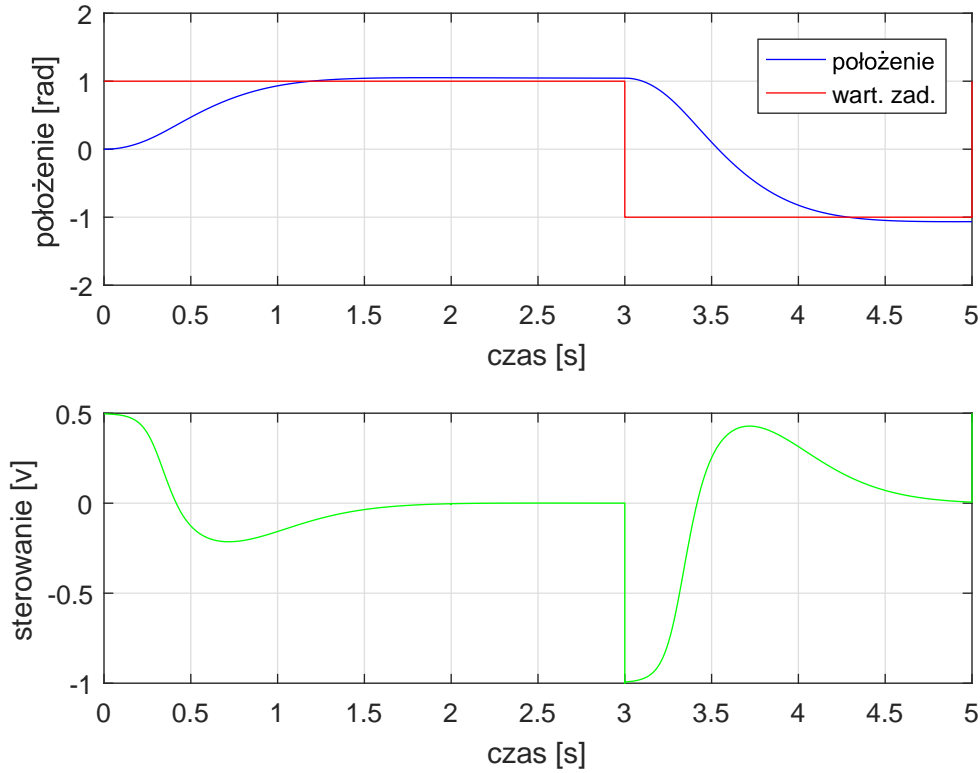
Wykorzystując funkcję optymalizacyjną *fmincon* środowiska MATLAB dobrano nastawy regulatora neuronowego opisanego w poprzedniej części tak aby minimalizować wskaźnik jakości J_3 . Poniżej zaprezentowano wartości poszczególnych parametrów regulatora oraz wartości wskaźników jakości.

$$J1 = 2.089 \text{ [rad}^2 \cdot \text{s]}$$

$$J2 = 0.438 \text{ [v}^2 \cdot \text{s]}$$

$$J3 = 2.527$$

$$W = \begin{bmatrix} D_1 & P_1 & I_1 & 0 \\ D_2 & P_2 & I_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.2402 & 2.6707 & 1.282 & 0 \\ 0.0661 & 0.7348 & 0.3541 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Rys. 4.4: Odpowiedź układu po optymalizacji $J = J_3$.

Z racji tej, że w układzie saturację zastąpiono funkcją sigmoidalną w wyniku działania regulatora otrzymano uchyb ustalony. Aby zniwelować ten efekt zmieniono postać wskaźnika jakości wykorzystywanego w funkcji optymalizującej na :

$$J = J_3 + 10 \cdot |z - \alpha_k| \quad (4.7)$$

gdzie:

z – wartość zadana,

α_k – położenie ramienia w stanie ustalonym.

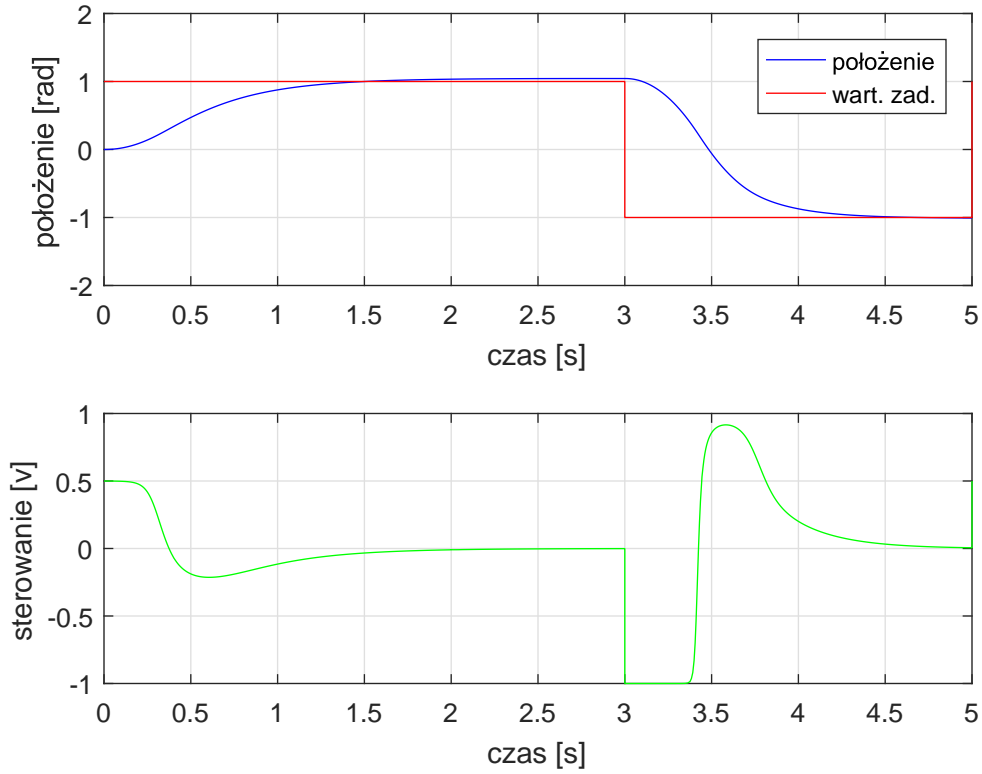
Dla tak zmodyfikowanego wskaźnika jakości otrzymano następujące parametry układu regulacji:

$$J1 = 1.963 \text{ [rad}^2 \cdot \text{s]}$$

$$J2 = 0.7543 \text{ [v}^2 \cdot \text{s]}$$

$$J3 = 2.718$$

$$W = \begin{bmatrix} D_1 & P_1 & I_1 & 0 \\ D_2 & P_2 & I_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0.2410 & 2.9867 & 1.5563 & 0 \\ 0.0635 & 3.0089 & 0.9648 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



Rys. 4.5: Odpowiedź układu po optymalizacji $J = J_3 + |z - \alpha_k|$.

4.2 Porównanie wskaźników jakości

Tabela 4.1: Porównanie wskaźników jakości regulator PID - neuronowy + saturacja - neuronowy + f. sigmoidalna.

Regulator	J_1	J_2	J_3
PID	3.739	0.841	4.580
Neuronowy + saturacja	3.739	0.841	4.580
Neuronowy + f. sigmoidalna	3.741	0.857	4.597
OPTYMALIZACJA			
Neuronowy1	2.08	0.438	2.527
Neuronowy2	1.963	0.7543	2.718

4.3 Projektowanie regulatora neuronowego z użyciem Neural Toolbox.

Zadanie polegało na zbadaniu jaka struktura regulatora neuronowego najlepiej odwzoruje pracę układu z regulatorem PID. Na podstawie sygnału sterującego wygenerowanego przez zaprojektowany we wcześniejszej części regulator PID badano która z rozpatrywanych struktur regulatora jest najlepsza pod względem minimalizacji całki z różnicy pomiędzy sterowaniem referencyjnym i sygnałem pochodzącym z otrzymanego regulatora. Do przeprowadzenia tej części ćwiczenia wykorzystano przybornik Neural Network ze środowiska MATLAB.

Rozpatrywano różne postaci sygnałów wejściowych

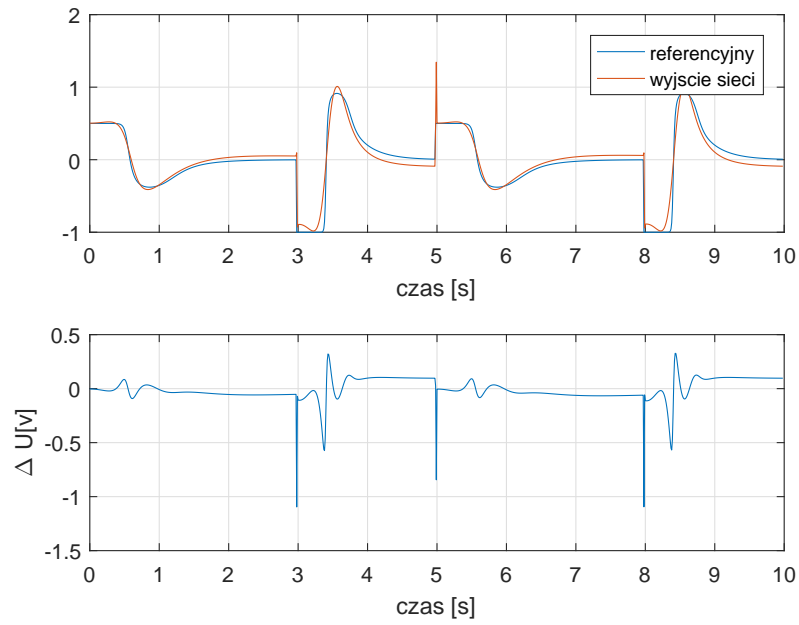
1. trzy ostatnie wartości uchybu regulacji,
2. trzy ostatnie wartości uchybu regulacji plus ostatnia wartość referencyjnego sygnału sterującego

Dodatkowo sprawdzono jak liczba neuronów wpływa na wyniki eksperymentu. Rozpatrywano odpowiednio 10 i 20 neuronów w strukturze regulatora.

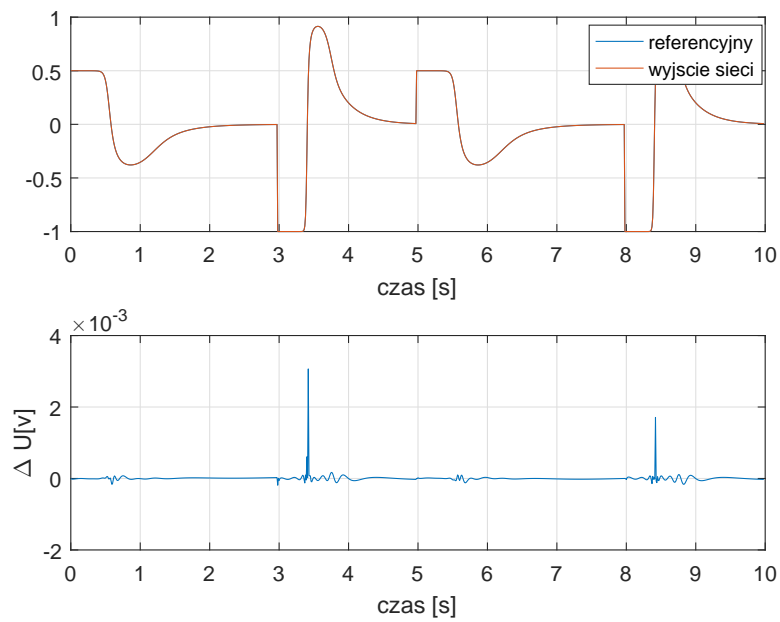
Otrzymane wyniki przedstawione są w tabeli 4.2

Tabela 4.2: Porównanie różnych struktur regulatora neuronowego w stosunku do regulatora PID.

wskaźnik jakości	10 neuronów	20 neuronów	10 neuronów + sterowanie	20 neuronów + sterowanie
$e [V^2 \cdot s]$	11.7862	1.5767	0.1073	$1.391 \cdot 10^{-5}$



Rys. 4.6: Porównanie sterowanie referencyjnego z wyjściem regulatora neuronowego l. neuronów = 10.



Rys. 4.7: Porównanie sterowanie referencyjnego z wyjściem regulatora neuronowego l. neuronów = 20, + sterowanie.

Regulator rozmyty

W tej sekcji przedstawione zostaną wyniki symulacji działania przyjętego układu z regulatorem rozmytym Mamdaniego. Cały proces projektowania struktury regulatora został przeprowadzony w toolbox-ie *Fuzzy* środowiska MATLAB. Głównym celem przeprowadzonych badań było zapoznanie się z zasadą działania regulatora rozmytego i jak najlepsze odwzorowanie działania oryginalnego regulatora PD.

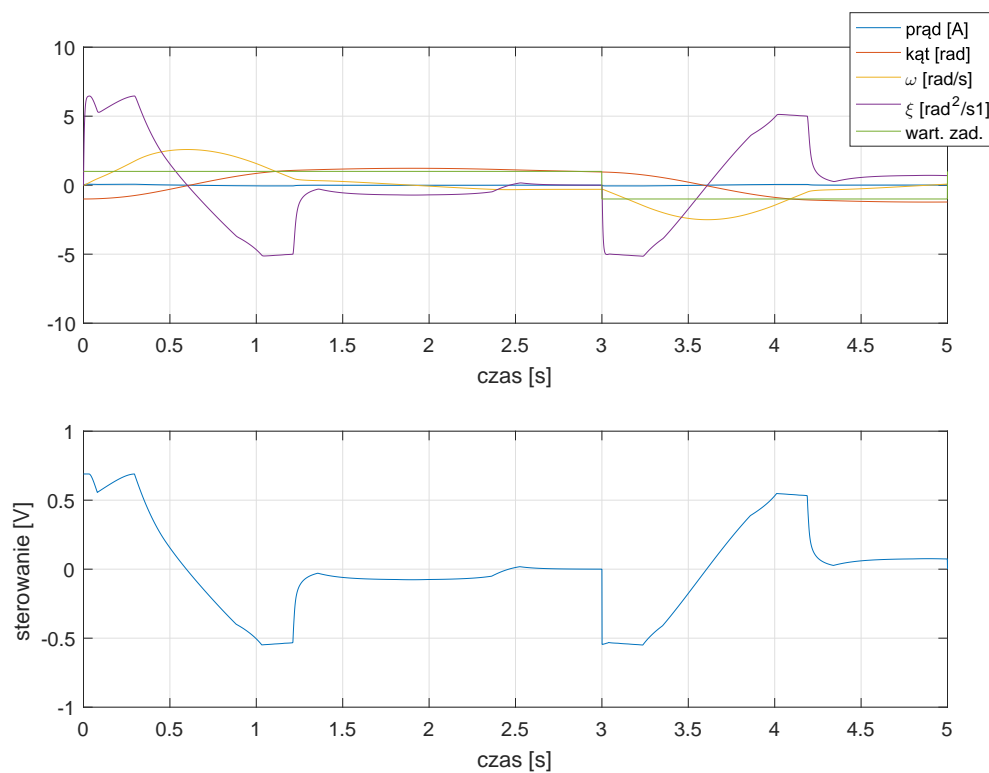
Zdecydowano, że sygnałami wejściowymi, na których bazował regulator będą uchyb i pochodna uchybu regulacji. Na bazie przebiegów wcześniej wspomnianych sygnałów została zaprojektowana baza reguł. Ze względu na różną dynamikę układu, zależną od wypełnienia szklanki, wprowadzono różne reguły dla sterowania. Wszystkie reguły w zależności od wartości uchybu regulacji i jego pochodnej podane są w tabeli 5.1. Symbole P i P_p oznaczają odpowiednio regułę "dodatnią" dla pełnej i pustej szklanki.

W ramach przeprowadzonych badań symulacyjnych porównano działania regulatora na bazie wcześniej przyjętych wskaźników jakości dla różnych postaci funkcji przynależności.

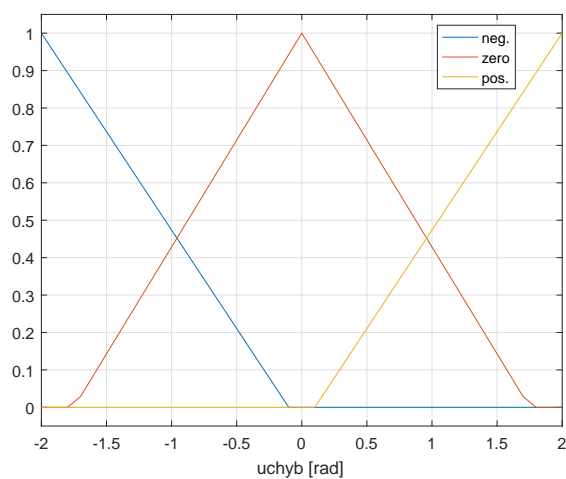
Tabela 5.1: Tabla reguł regulatora fuzzy. N - wart. ujemna, Z - wart. zerowa, P - wart. dodatnia, P_p - wart dodatnia dla pustej szklanki

$e/\frac{de(t)}{dt}$	N	Z	P
N	Z	N	N
Z	N	Z	P
P	P_p	P_p	Z

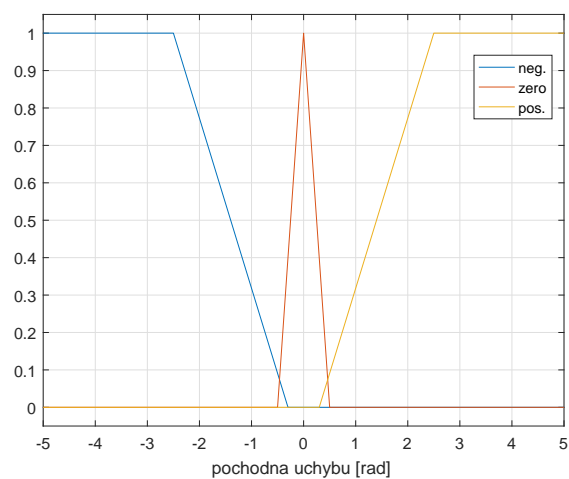
5.1 Pierwszy zestaw funkcji przynależności



Rys. 5.1: Odpowiedź obiektu dla regulatora rozmytego.

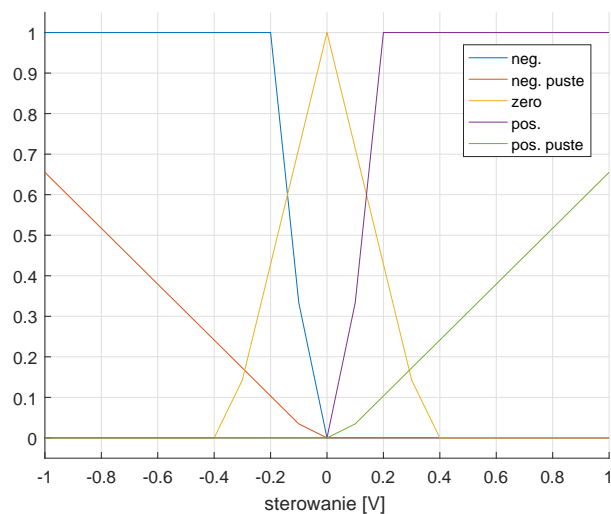


(a) Reguły dla uchybu regulacji.



(b) Reguły dla pochodnej uchybu.

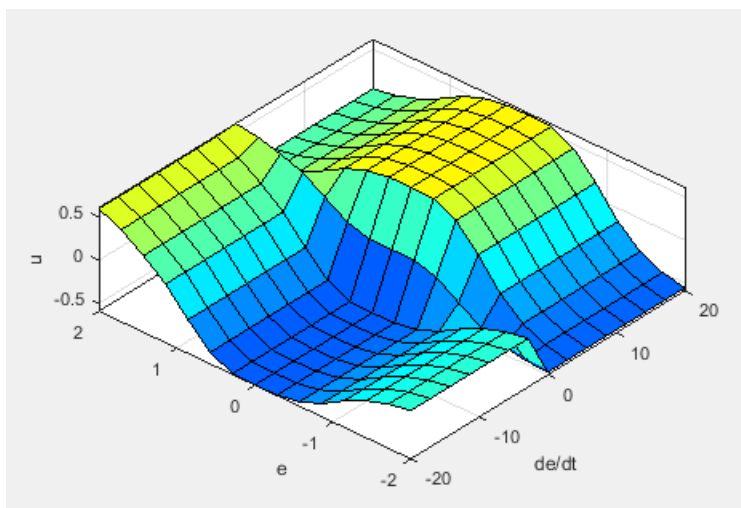
Rys. 5.2: Reguły dla pierwszego zestawu funkcji przynależności.



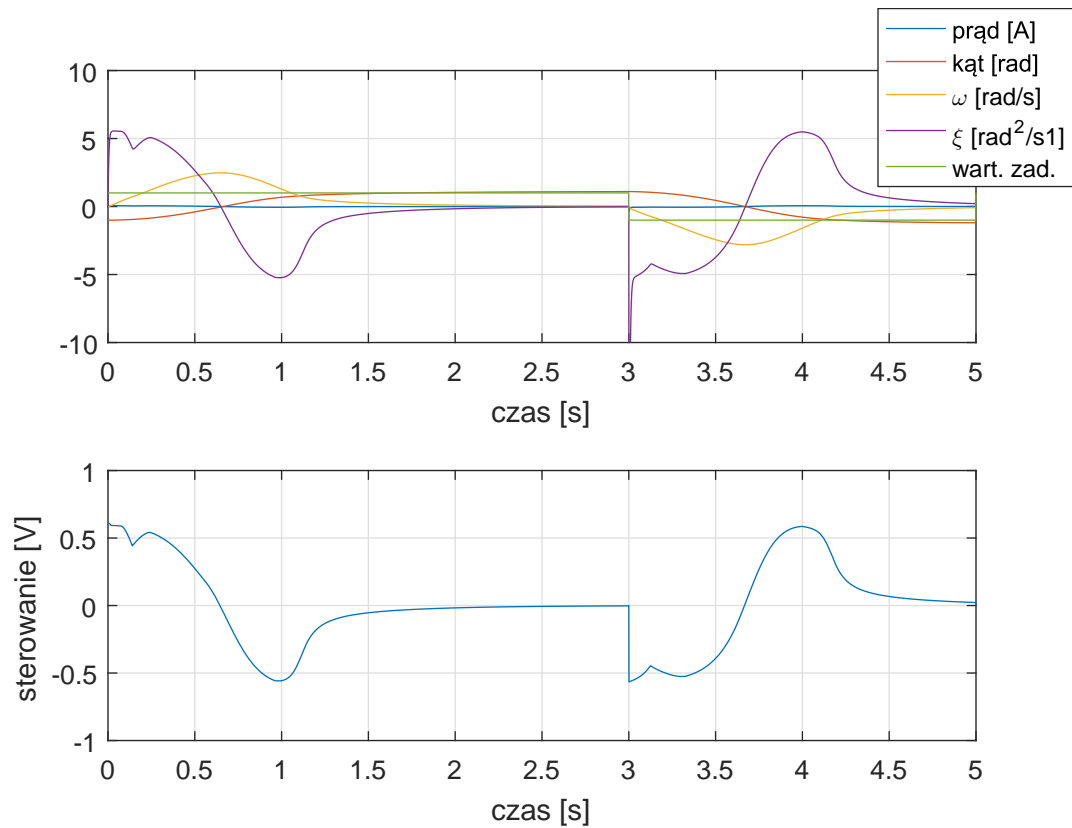
Rys. 5.3: Reguły dla sterowania - I zestaw funkcji przynależności.

5.2 Drugi zestaw funkcji przynależności

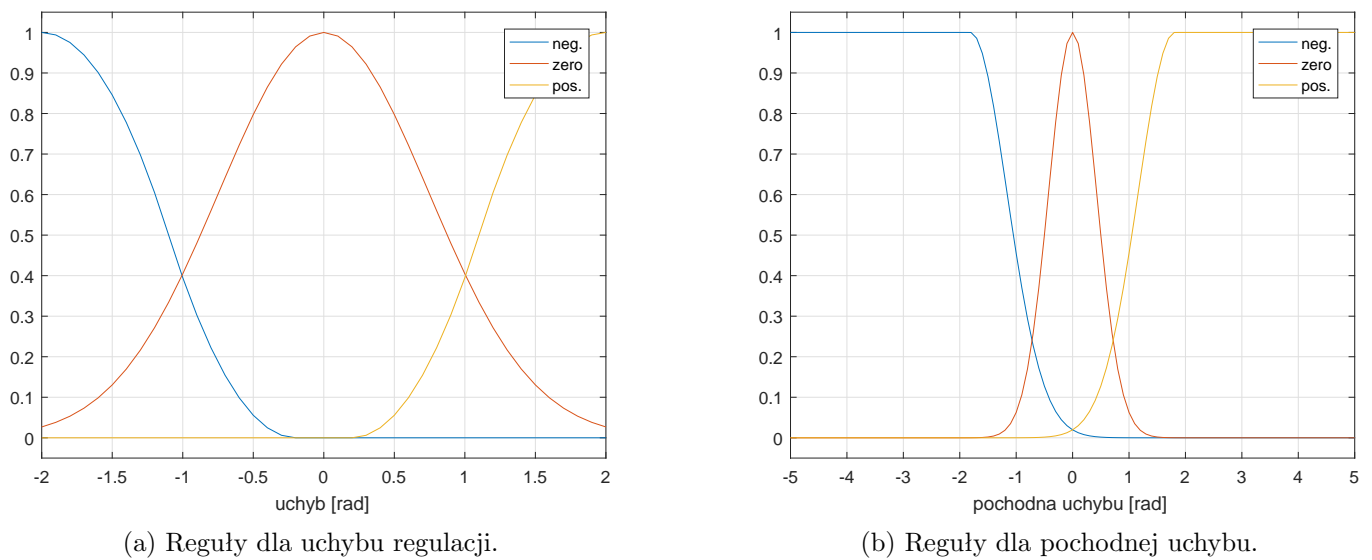
Na rysunku 5.4 zaprezentowano płaszczyznę wyznaczoną przez przyjęte reguły.



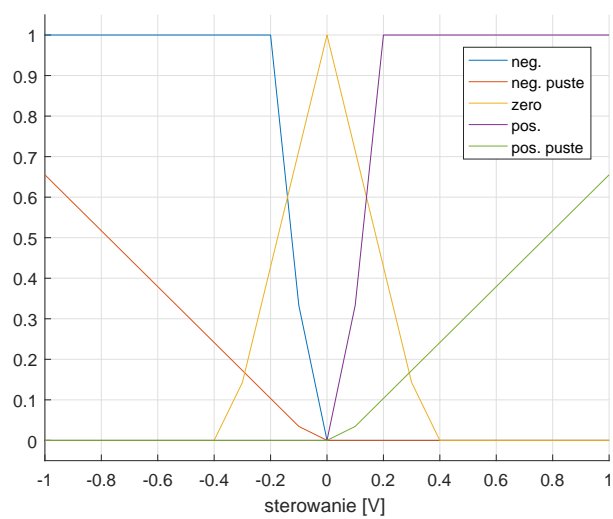
Rys. 5.4: Płaszczyzna sterowań wyznaczona przez reguły zestawu nr II.



Rys. 5.5: Odpowiedź obiektu dla regulatora rozmytego, II zestaw funkcji przynależności.



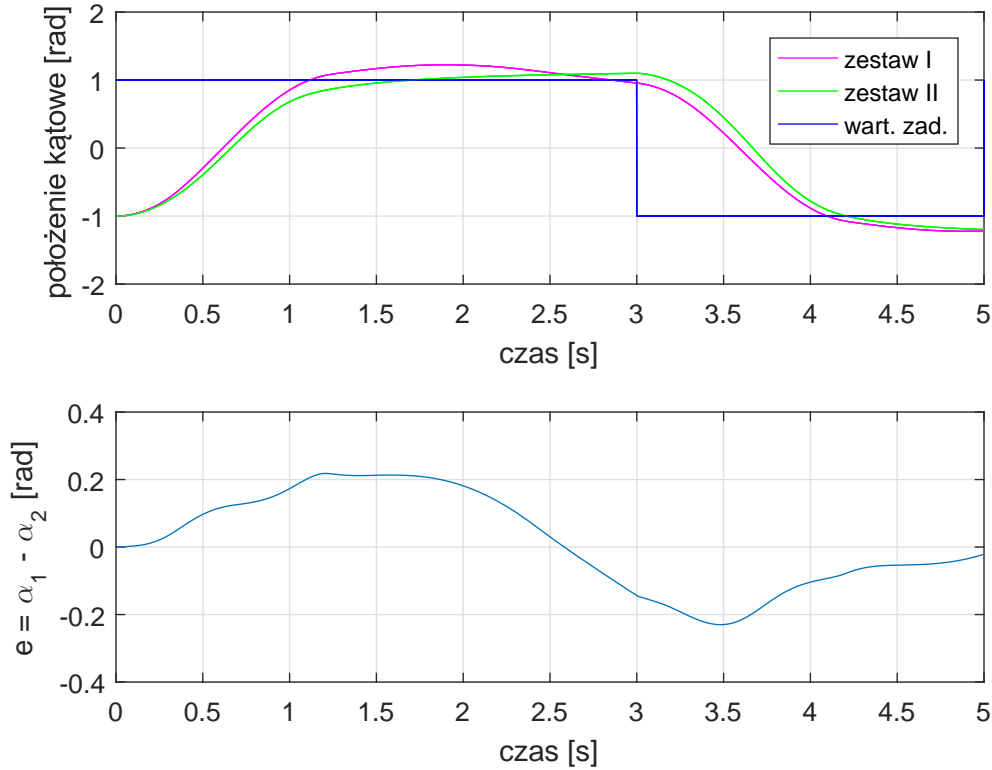
Rys. 5.6: Reguły dla drugiego zestawu funkcji przynależności.



Rys. 5.7: Reguły dla sterowania - II zestaw funkcji przynależności.

5.3 Porównanie

Na rysunku 5.8 przedstawiono porównanie przebiegów odpowiedzi obiektu dla każdego z rozpatrywanych zestawów reguł. W tabeli 5.2 zapisano wartości wskaźników jakości dla obu rozpatrywanych przypadków.



Rys. 5.8: Porównanie działania regulatora dla oby zestawów reguł.

Tabela 5.2: Wskaźniki jakości dla regulatora PD - fuzzy.

Funkcje przynależności	J_1	J_2	J_3
I zestaw	3.624	1.96	5.583
II zestaw	4.201	1.667	5.868

Z przebiegów zamieszczonych na rysunku 5.8 i danych z tabeli 5.2 wynika, że w przypadku drugiego zestawu funkcji przynależności odpowiedź obiektu ma gorszą dynamikę i mniejsze przeregulowanie niż w przypadku zestawu nr 1. W przypadku wykorzystania funkcji gaussowskich otrzymano dużo lepszy wskaźnik jakości J_2 , który jest odpowiednikiem energii dostarczonej do układu.

5.4 Regulator Takagi - Sageno

Główną różnicą pomiędzy regulatorem rozmytym typu Takagi-Sageno (1985 r.), a strukturą regulatora zaproponowaną przez Mamdaniego jest liniowa funkcja wyjścia.

5.4.1 Manualny dobór struktury regulatora

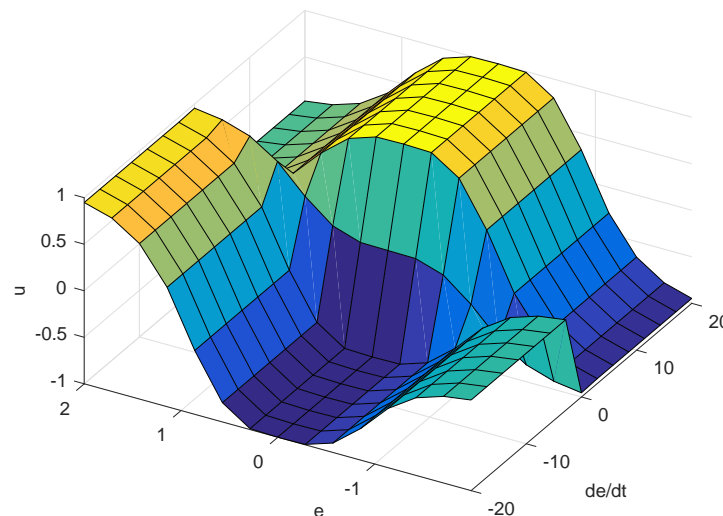
W rozpatrywanym przypadku reguły przynależności do zbiorów rozmytych dla sygnału wejściowego są identyczne jak te opisane w podrozdziale 5.2. Wielkościami poszukiwanymi w tym przypadku były wartości funkcji wyjściowej dla każdego z rozpatrywanych przypadków. Przyjęto, że będzie ona przyjmować wartości ze zbioru $-1, 0, 1$. W tabeli 5.3 zamieszczone są wszystkie reguły zaprojektowanego regulatora.

Tabela 5.3: Tabelka reguł regulatora typu Takagi-Sageno.

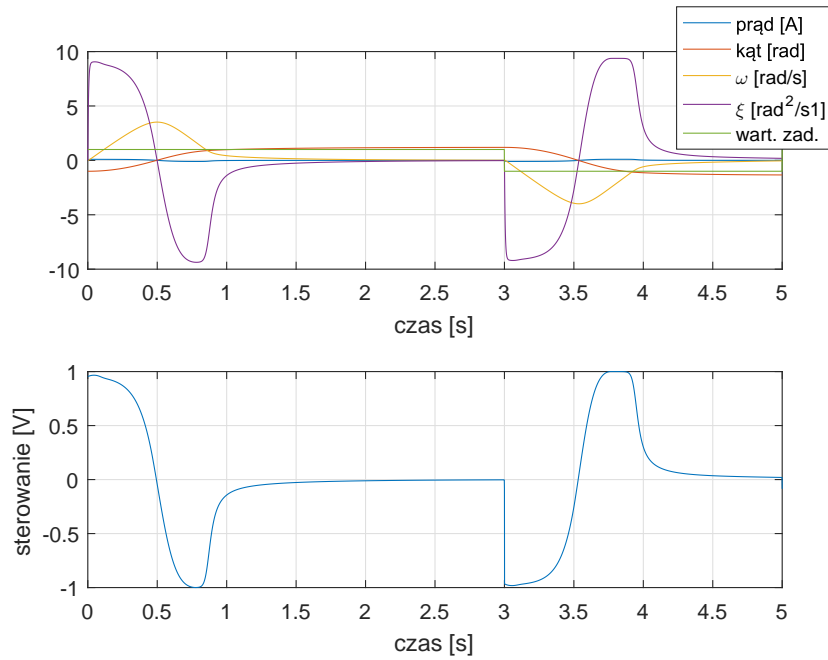
$e/\frac{de(t)}{dt}$	N	Z	P
N	0	-1	-1
Z	-1	0	1
P	1	1	0

Jak można zauważyć, tabela 5.3 jest analogiczna do tabeli 5.1 z tą różnicą, że nieliniowe funkcje przynależności zastąpiono liczbami.

Na rysunku 5.9 zaprezentowano powierzchnię sterowania dla tak dobranej struktury regulatora, a na rysunku 5.10 odpowiedź układu.



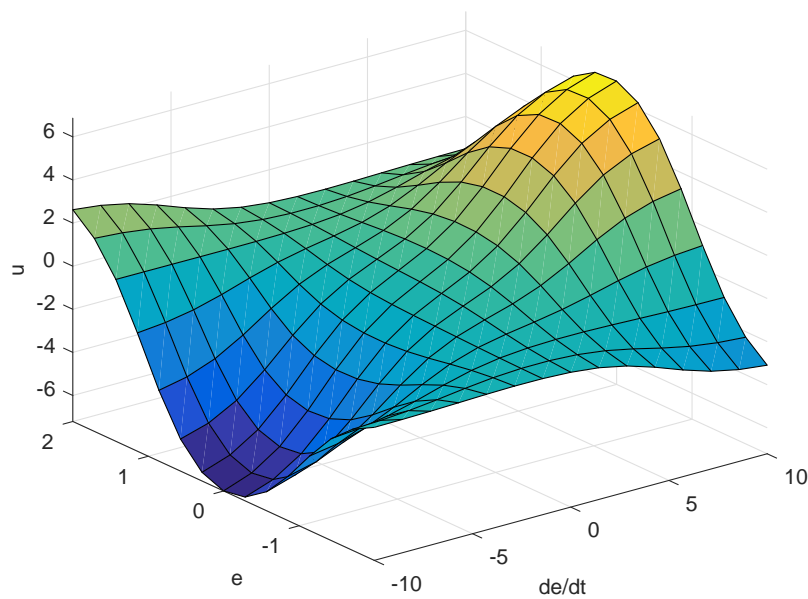
Rys. 5.9: Powierzchnia sterowania dla regulatora rozmytego typu Takagi-Sageno.



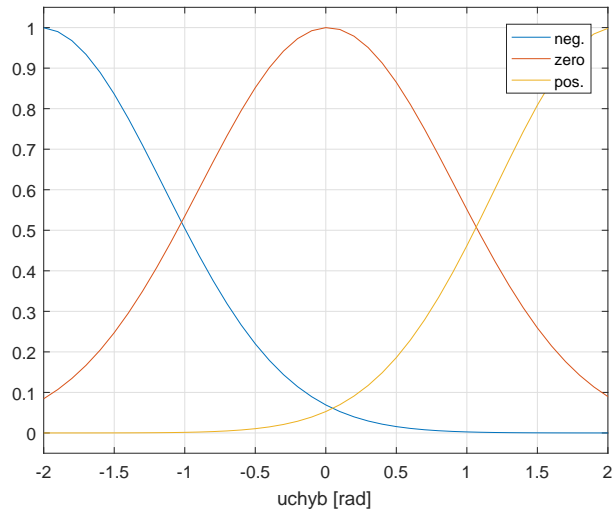
Rys. 5.10: Odpowiedź obiektu dla regulatora rozmytego typu Takagi-Sageno.

5.4.2 Optimalizacja struktury regulatora

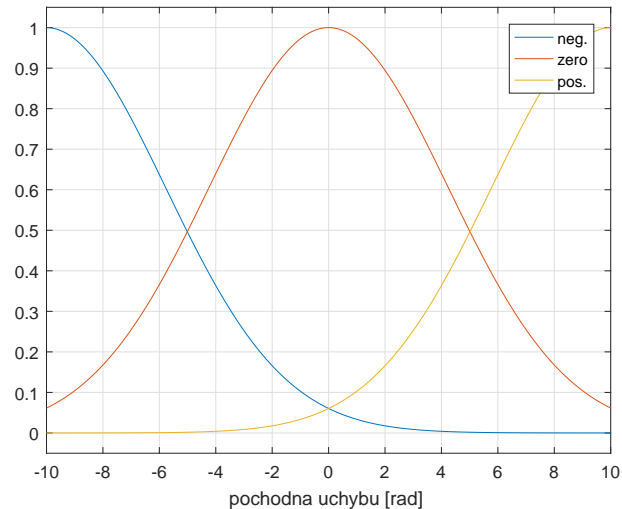
W kolejnym kroku wykorzystano funkcje środowiska Matlab *genfis1* i *anfis* do wyznaczenia struktury regulatora. W procesie konfiguracji ustawiono gaussowskie funkcje przynależności zarówno dla uchybu regulacji jak i dla jego pochodnej. Na rysunkach 5.11 - 5.12a zaprezentowano płaszczyznę sterowania oraz funkcje przynależności zwrócone przez wcześniej wymienione funkcje.



Rys. 5.11: Powierzchnia sterowania dla regulatora rozmytego typu Takagi-Sageno po optymalizacji.



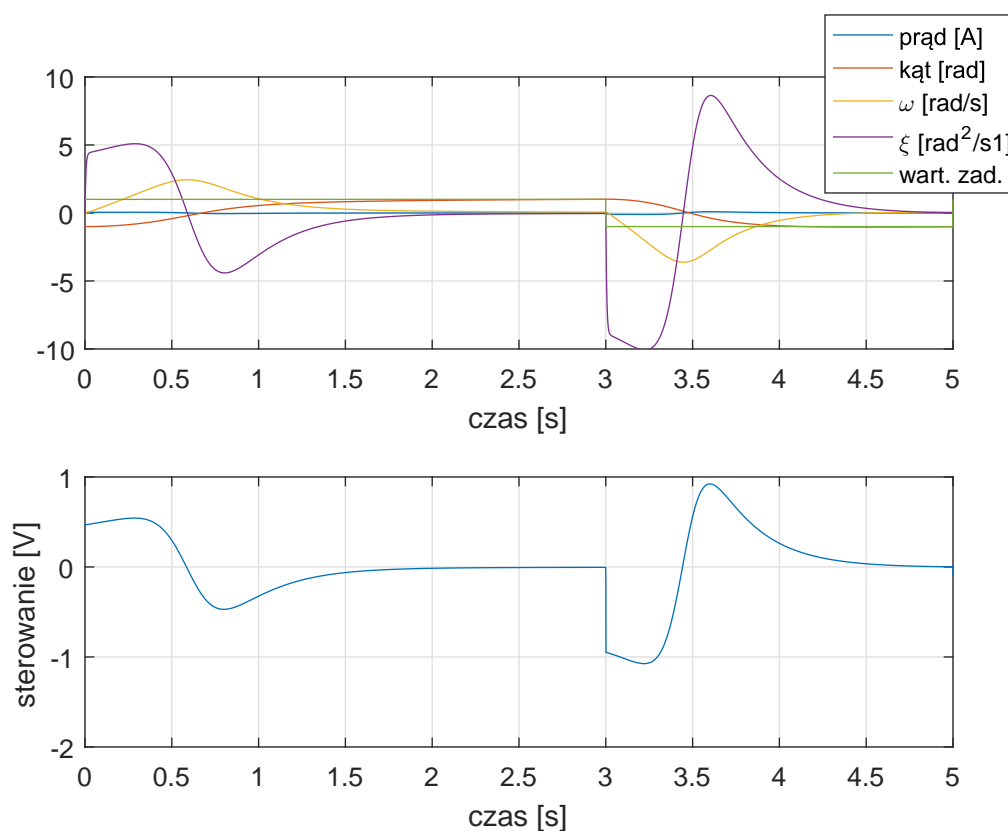
(a) Reguły dla uchybu regulacji.



(b) Reguły dla pochodnej uchybu.

Rys. 5.12: Reguły dla regulatora T-S po optymalizacji.

Rysunek 5.13 przedstawia przebiegi wszystkich zmiennych stanu rozważanego układu.



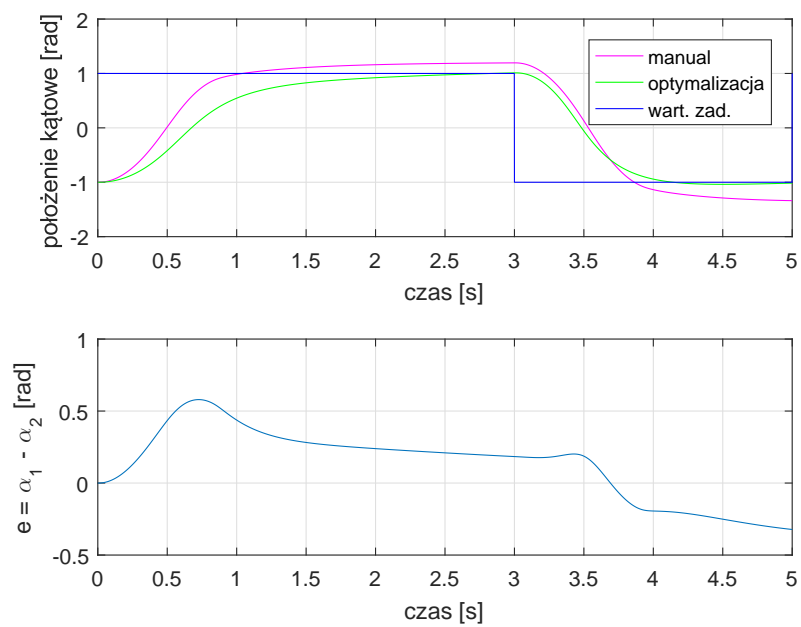
Rys. 5.13: Odpowiedź obiektu dla regulatora rozmytego typu Takagi-Sageno po optymalizacji.

5.4.3 Porównanie

Porównując wartości w tabeli 5.4 i przebiegi na rysunku 5.14 można stwierdzić, że odpowiedź obiektu dla regulatora otrzymanego w wyniku przeprowadzenia procedury optymalizującej ma gorszą dynamikę niż w przypadku ręcznego strojenie. W zamian za to otrzymaliśmy znacząco lepszą wartość wskaźnika J_2 oraz mniejsze przeregulowanie.

Tabela 5.4: Porównanie wskaźników jakości dla regulatora rozmytego typu Takagi-Sageno.

Sposób projektowania	J_1	J_2	J_3
manualny	3.497	2.639	6.136
optymalizacja	3.642	0.8707	4.513



Rys. 5.14: Porównanie działania regulatora przed i po optymalizacji.

Regulator rozmyty dla modelu helikoptera

6.1 Wstęp

Celem przedstawionych w niniejszym rozdziale badań było sprawdzenie jak regulator rozmyty współpracuje z rzeczywistym obiektem. Do eksperymentów wykorzystany został model helikoptera oraz jego model matematyczny sporządzony na potrzeby Laboratorium Problemowego. Zadaniem zaprojektowanego regulatora była stabilizacja układu w zadanym położeniu w osi poziomej.

6.2 Model matematyczny

Obiekt opisany jest następującymi równaniami:

$$\begin{aligned} J_v \frac{d^2 \alpha_v}{dt^2} &= -f_v \frac{d\alpha_v}{dt} + a \cdot \sin(\alpha_v - \alpha_{v0}) + M_v(\omega_v) \\ I_v \frac{d\omega_v}{dt} &= u_v - H_v^{-1}(\omega_v) \end{aligned} \tag{6.1}$$

gdzie:

α_v jest kątem obrotu w płaszczyźnie pionowej,

J_v jest momentem bezwładności względem osi obrotu w płaszczyźnie pionowej,

f_v jest współczynnikiem tarcia lepkiego,

a jest momentem sił grawitacji,

α_{v0} jest kątem równowagi układu w płaszczyźnie pionowej,

I_v jest momentem bezwładności dużego śmigła,

$H_v^{-1}(\omega_v)$ jest charakterystyką statyczną układu silnik-śmigło dla silnika głównego,

ω_v jest prędkością obrotową silnika głównego,

$M_v(\omega_v)$ jest momentem sił generowanym przez silnik główny,

u_v jest współczynnikiem wypełniania sygnału PWM dla silnika głównego.

Równania stanu sporządzone na podstawie równań 6.1 mają postać:

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\frac{f_v}{J_v}x_2 + \frac{a}{J_v}\sin(\alpha_v - \alpha_{v0}) + \frac{M_v(\omega_v)}{J_v} \\ \dot{x}_3 &= \frac{u_v}{I_v} - \frac{H_v^{-1}(\omega_v)}{I_v}\end{aligned}\tag{6.2}$$

Proces identyfikacji parametrów równań 6.2 opisany jest w pracy [1].

6.3 Regulator referencyjny

Aby dobrać strukturę regulatora rozmytego typu Sageno zdecydowano wykorzystać regulator LQI. Podejście to wymagało zlinearyzowania równań 6.2 oraz zaprojektowania obserwatora Luengergera ([1]) w celu estymacji pełnego stanu obiektu.

Zlinearyzowany model w położeniu poziomym opisany jest równaniem

$$\begin{aligned}\dot{x} &= Ax + B \\ y &= Cx + D\end{aligned}\tag{6.3}$$

gdzie:

$$\begin{aligned}A &= \begin{bmatrix} 0 & 1 & 0 \\ -4.4005 & -0.0695 & 0.0244 \\ 0 & 0 & -2.8870 \end{bmatrix} \\ B &= \begin{bmatrix} 0 \\ 0 \\ 577.5771 \end{bmatrix} \\ C &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ D &= 0\end{aligned}\tag{6.4}$$

Przyjęto następujące wartości własne obserwatora:

$$\begin{aligned}\lambda_1 &= -3 \\ \lambda_2 &= -6 \\ \lambda_3 &= -9\end{aligned}\tag{6.5}$$

Obserwator Luenbergera pełnego rzędu opisany jest równaniem różniczkowym

$$\dot{w} = Aw + L(y - Cw) + Bu\tag{6.6}$$

gdzie:

A jest macierzą stanu obserwowanego układu,

B jest macierzą sterowania obserwowanego układu,

C jest macierzą wyjścia obserwowanego układu,

G jest macierzą wybraną tak, by wartości własne macierzy $A - LC$ miały ujemne części rzeczywiste,

w estymuje obserwowany stan.

Wybrana macierz L miała następującą postać:

$$L = \begin{bmatrix} 15.0434 \\ 49.9218 \\ 87.9693 \end{bmatrix} \quad (6.7)$$

Regulator LQI opisany jest zależnością

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt$$

$$Q = \begin{bmatrix} 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (6.8)$$

$$R = 1$$

gdzie:

x_4 jest całką z uchybu kąta.

Wektor wzmocnień 6.9 regulatora został wyliczony za pomocą funkcji *lqi* środowiska *Matlab*.

$$K = \begin{bmatrix} 0.3154 & 0.7105 & 0.0042 & -1 \end{bmatrix} \quad (6.9)$$

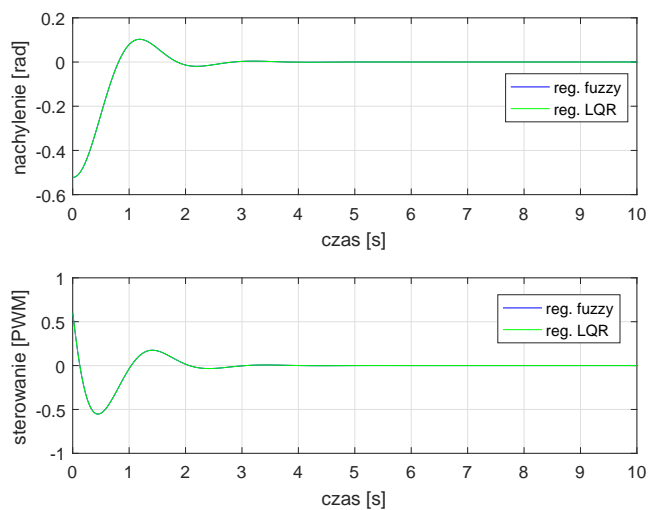
6.4 Regulator rozmyty

Do doboru struktury regulatora rozmytego typu Sageno wykorzystano funkcję *genfis* środowiska *Matlab*. W pierwszym etapie sprawdzono jakość regulacji projektując regulator *fuzzy* na bazie modelu matematycznego. Na rysunku 6.1a przedstawiono odpowiedzi modelu obiektu w przypadku działania regulatora LQI i fuzzy.

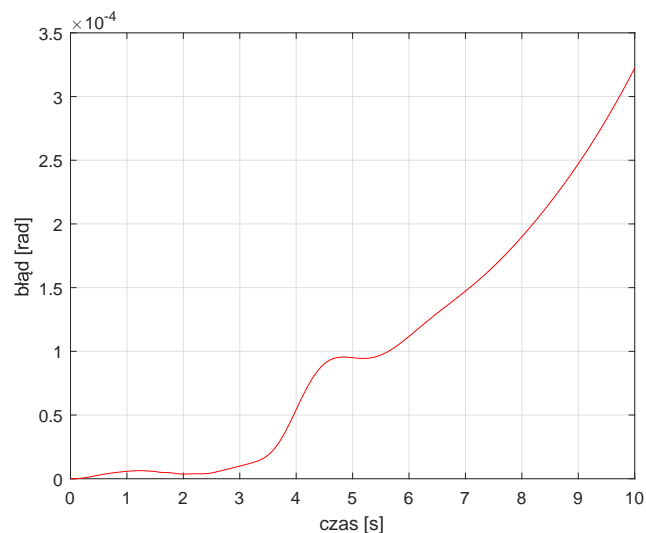
Tabela 6.1: Porównanie wskaźników jakości regulator LQI i fuzzy dla modelu.

Regulator	J_1
LQI	0.102
fuzzy	0.102

Wartości wskaźników jakości zaprezentowane w tabeli 6.1 oraz przebiegi zaprezentowane na rysunkach 6.1a i 6.1b pokazują że w przypadku matematycznego modelu rozważanego obiektu klasyczny regulator LQI może być z powodzeniem zastąpiony regulatorem rozmytym bez pogorszenia jakości sterowania.

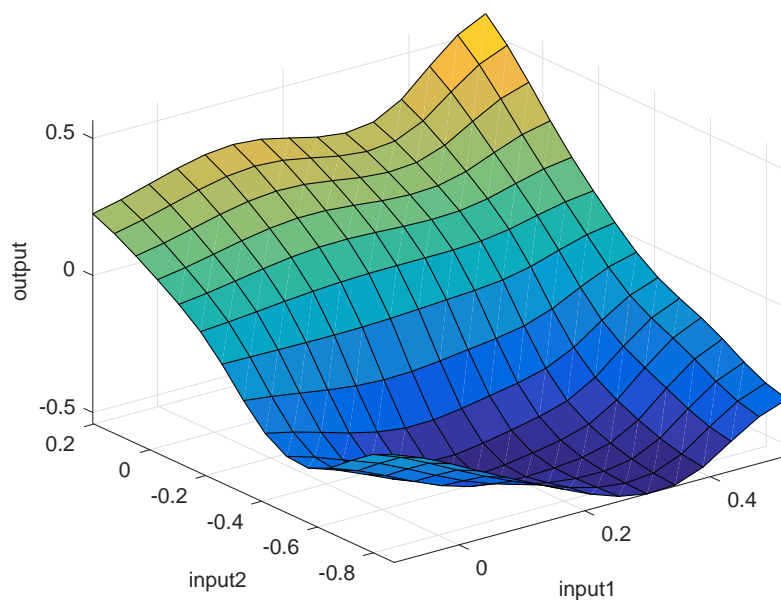


(a) Porównanie działania regulatora LQR i fuzzy.



(b) Różnica działania regulatora LQR i fuzzy.

Rys. 6.1: Działanie regulatora dla modelu obiektu.



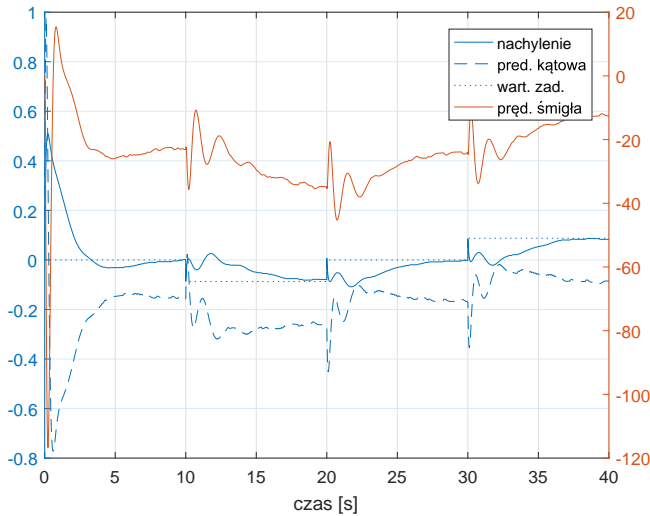
Rys. 6.2: Powierzchnia sterowania dla modelu obiektu rzutowana na pierwszą i drugą zmienną stanu.

Następnie postanowiono sprawdzić działanie regulatora na rzeczywistym obiekcie. W tym celu na wejście obiektu podano sygnał pokazany na rysunku 6.3a. Podczas działania układu rejestrowano wartości zmiennych stanu (wyjście z obserwatora Luenbergera), całki uchybu regulacji oraz sterowania.

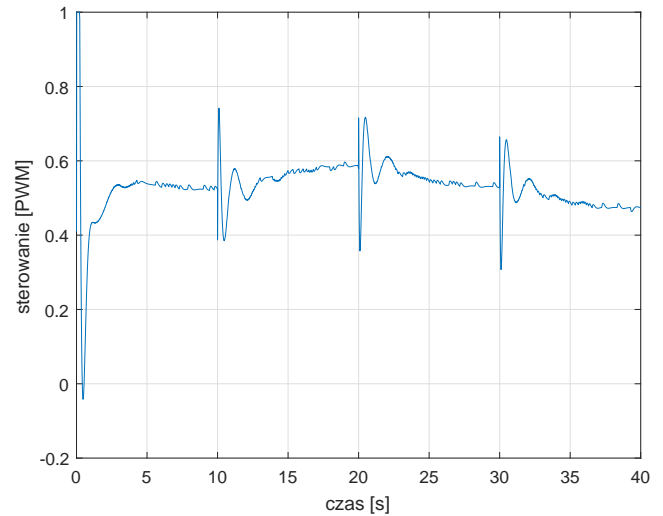
Zdecydowano, że do procedury *genfis* podane zostaną przebiegi zarejestrowanych zmiennych

między 20 a 40 sekundą. Decyzja ta wynikała z chęci uniknięcia optymalizacji struktury regulatora na danych z początkowego etapu działania obserwatora gdy estymowane wartości zmiennych stanu znacznie odbiegały od rzeczywistych.

Niestety nie udało się przetestować działania tak zaprojektowanego regulatora na obiekcie

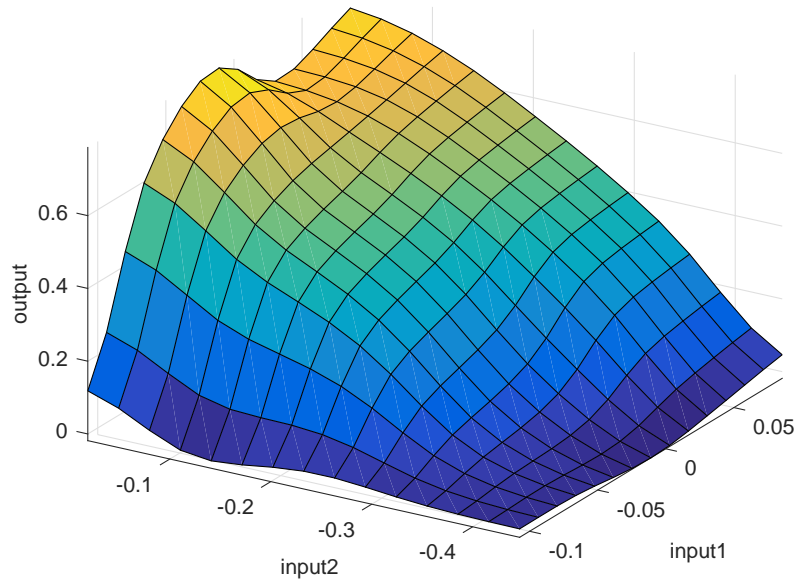


(a) Przebieg zmiennych stanu.



(b) Sterowanie podawane na obiekt.

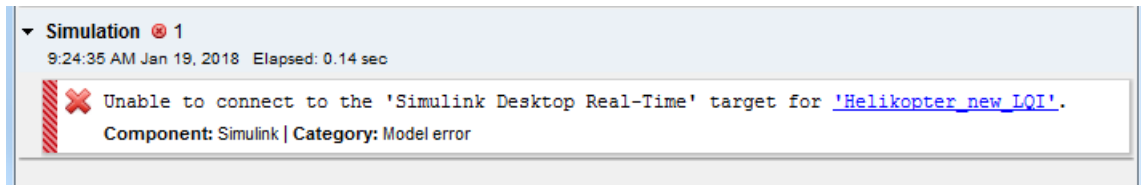
Rys. 6.3: Dane wykorzystane do optymalizacji struktury reg. rozmytego.



Rys. 6.4: Powierzchnia sterowania dla obiektu rzutowana na pierwszą i drugą zmienną stanu.

ponieważ *Simulink* nie potrafił uruchomić modelu w czasie rzeczywistym gdy był wykorzy-

stywany w nim bloczek *Fuzzy Logic Controller*.



Rys. 6.5: Błąd zwracany przez *Simulink*.

Na podstawie powierzchni sterowania zaprezentowanej na rysunku 6.4 można stwierdzić że struktura regulatora jest podobna do uzyskanej na podstawie danych symulacyjnych rysunek 6.2.

Wnioski

Przeprowadzenie badań opisanych w niniejszym sprawozdaniu pozwoliło zapoznać się, z innymi niż dotychczas poznane, rodzajami regulatorów. Zaprojektowanie od podstaw regulatora neuronowego jak i również rozmytego pozwoliło lepiej zrozumieć zasadę ich działania. Dzięki wybraniu, na potrzeby symulacji, prostego obiektu drugiego rzędu mogliśmy w łatwy sposób badać wpływ wprowadzanych w strukturze regulatora modyfikacji na działanie całego systemu. Zamieszczone w sprawozdaniu porównania klasycznych regulatorów z regulatorem neuronowym oraz rozmytym pokazało, że każde z rozważanych podejść daje porównywalne wyniki i może być zastosowane w praktyce.

Podsumowując, regulator neuronowy jak i rozmyty spełnił swoją rolę, jednak aby uzyskać zbliżone wyniki do regulatorów klasycznych należało poświęcić więcej czasu na dobraniu odpowiedniej struktury i parametrów regulatora. Wykorzystanie sieci neuronowych bądź też logiki rozmytej może mieć swoje uzasadnienie w przypadku bardziej złożonych systemów. Dla prostych obiektów pierwszego i drugiego rzędu może być ciężko uzyskać lepszą jakość regulacji niż w przypadku regulator PID lub LQ bez wykorzystania dedykowanych narzędzi *Matlab-a*.

Bibliografia

- [1] Cebula M., Kowalczyk M., Rubak D.: *Model helikoptera. Laboratorium Problemowe 2*, Kraków 2018