

Warszawa 2014

Politechnika Warszawska
Wydział Elektroniki i Technik Informacyjnych
Instytut Automatyki i Informatyki Stosowanej



Łukasz Jendrzejek
Paweł Świokło

SZTUCZNA INTELIGENCJA W AUTOMATYCE - PROJEKT

**Sterowanie kaskadą zbiorników z wykorzystaniem modelu
rozmytego obiektu.**

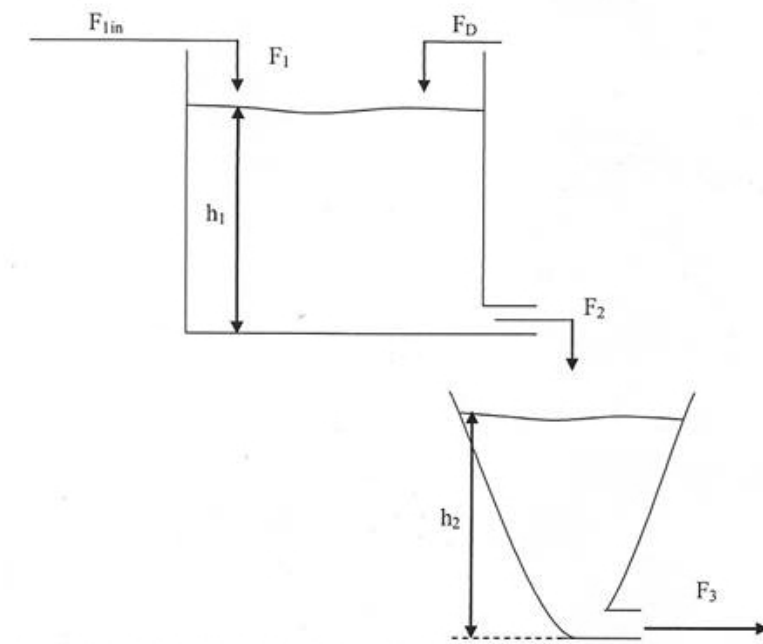
Spis treści

1	Opis obiektu	3
2	Modelowanie otrzymanego obiektu	4
2.1	Symulacja pełnego, nieliniowego obiektu	4
2.2	Liniowy model ciągły	6
2.3	Liniowy model dyskretny	8
2.4	Model rozmyty typu Takagi-Sugeno	9
2.5	Strojenie parametrów modelu rozmytego.	11

1 Opis obiektu

Obiektem rozważanym w niniejszym zadaniu jest kaskada dwóch zbiorników. Zadanie sterowania polega na odpowiednim ustalaniu dopływającego strumienia medium tak, aby stabilizować poziom tego medium w drugim zbiorniku. Schemat obiektu został ukazany na rysunku 1. Są na nim ukazane dwa strumienie dostarczające medium:

- strumień sterowany F_{1in} ,
- strumień zakłócający F_D .



Rysunek 1: Graficzny schemat struktury rozważanej kaskady zbiorników.

Poziom wypełnienia zbiorników reprezentowany jest przez parametry odpowiednio h_1 i h_2 . Ważnym elementem występującym niejawnie na schemacie jest opóźnienie strumienia wejściowego. Wartością sterowaną jest dopływ reprezentowany przez F_{1in} , a strumień dopływający do pierwszego strumienia w danej chwili określany jest przez wielkość F_1 .

Wraz z obiektem zostały dostarczone równania opisujące występujące zależności:

$$\begin{cases} \frac{dV_1}{dt} = F_1 + F_D - F_2(h_1) \\ \frac{dV_2}{dt} = F_2(h_1) - F_3(h_2) \\ F_2(h_1) = \alpha_1 \cdot \sqrt{h_1} \\ F_3(h_2) = \alpha_2 \cdot \sqrt{h_2} \\ V_1(h_1) = A_1 \cdot h_1 \\ V_2(h_2) = C_2 \cdot h_2^2 \\ F_1(t) = F_{1in}(t - \tau) \end{cases} \quad (1)$$

Stałe opisujące zbiornik wynoszą odpowiednio:

- $A_1 = 550 \text{ cm}^2$
- $C_2 = 0.8$
- $\alpha_1 = 28$
- $\alpha_2 = 16$

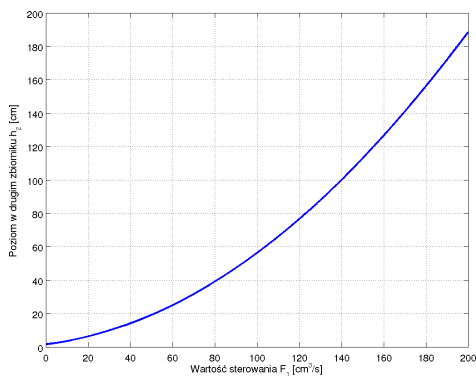
a punkt pracy, wokół którego odbywać się będzie linearyzacja i sterowanie obiektu dany jest parametrami o wartościach:

- $F_1 = 100 \frac{\text{cm}^3}{\text{s}}$
- $F_D = 20 \frac{\text{cm}^3}{\text{s}}$
- $\tau = 60 \text{ s}$
- $h_2 = 56.25 \text{ cm}$

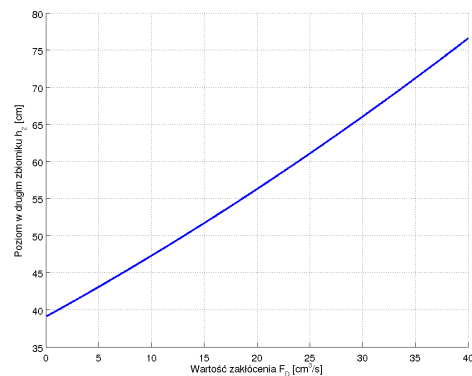
2 Modelowanie otrzymanego obiektu

Pierwszą częścią projektu było skupienie się na samym obiekcie i sposobach jego modelowania. W pierwszej kolejności obiekt zostanie wiernie odwzorowany, wykorzystane będą bezpośrednio podane równania różniczkowe dostarczone wraz z treścią projektu. Po analizie zachowania obiektu i sprawdzeniu jego odpowiedzi na skoki wartości sygnałów na wejściach dokonana zostanie linearyzacja w punkcie pracy, a następnie dyskretyzacja uzyskanego modelu liniowego w celu wykorzystania go do regulacji. Ostatnim etapem modelowania będzie utworzenie modelu rozmytego obiektu.

2.1 Symulacja pełnego, nieliniowego obiektu



(a) Charakterystyka statyczna toru wejście-wyjście



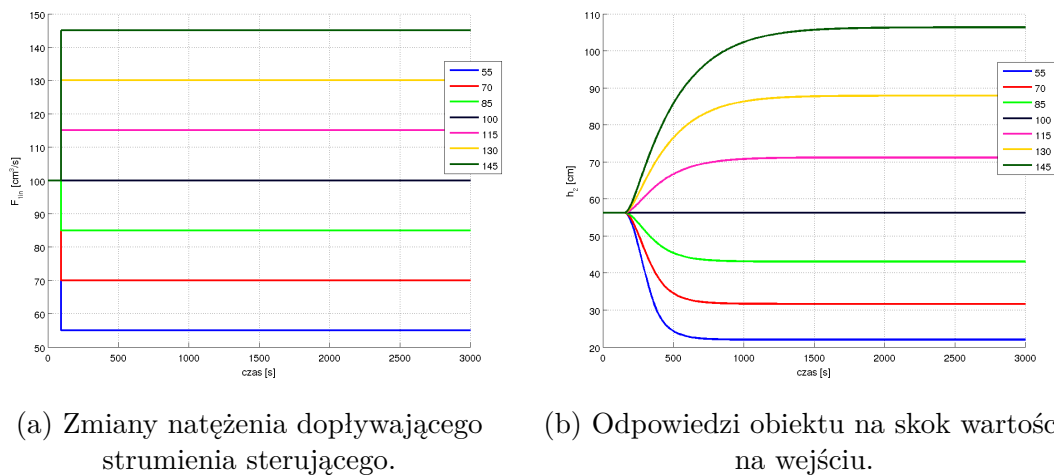
(b) Charakterystyka statyczna toru zakłócenie-wyjście.

Rysunek 2: Charakterystyki statyczne obiektu.

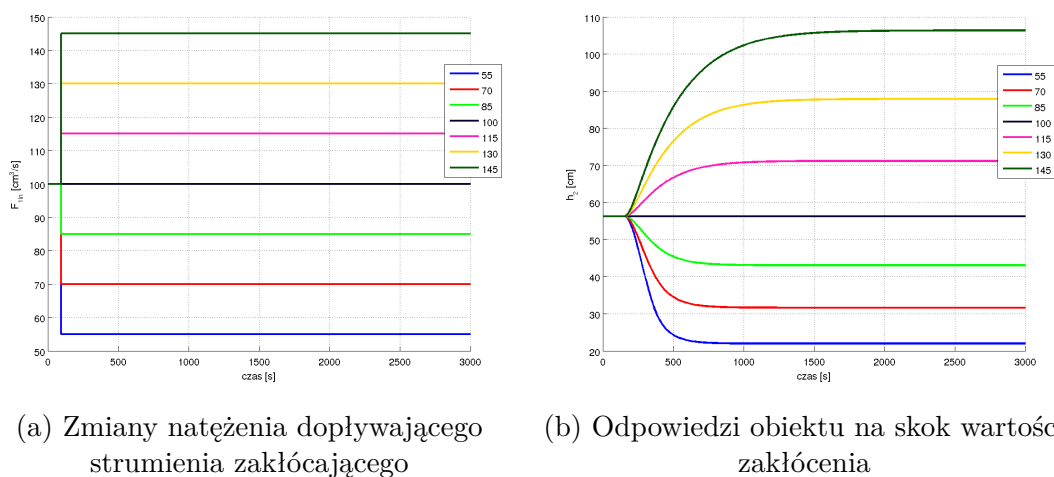
Opis matematyczny obiektu wyrażony w postaci równań różniczkowych został rozwiązany za pomocą funkcji `ode45` programu Matlab posługującej się metodą Rungego-Kutty. Pierwszym krokiem było wykreślenie charakterystyki statycznej obiektu. W tym celu obiekt był wielokrotnie symulowany aż do ustalenia się odpowiedzi na jego wyjściu na stałe sterowanie. Wyniki symulacji zostały przedstawione na rysunku 2.

Przebieg charakterystyki statycznej obrazuje zmianę wzmocnienia statycznego obiektu. Można zauważyć silną nieliniowość w torze wejście-wyjście, tak więc adekwatność modelu liniowego będzie silnie ograniczona do pewnego sąsiedztwa punktu pracy.

Następnie sprawdzona została nieliniowość dynamiczna obiektu. Rysunki 3 i 4 przedstawiają odpowiedzi obiektu na skoki wartości odpowiednio sterowania i zakłócenia. Wynika z nich, że obiekt wyraźnie szybciej reaguje na zmniejszanie dopływu do zbiorników.



Rysunek 3: Reakcja obiektu na zmianę wielkości dopływającego strumienia sterującego.



Rysunek 4: Reakcja obiektu na zmianę wielkości dopływającego strumienia zakłócającego.

Powyższe wykresy ilustrują poprawne działanie modelu opisanego równaniami różniczkowymi. Pełne modele nieliniowe opisujące rzeczywistość równaniami różniczkowymi ciężko jednak na ogół wykorzystać w algorytmie sterowania, dlatego zostaną sporządzone modele liniowe i rozmyte, które zostaną bezpośrednio wykorzystane do syntezy regulatorów obiektu.

2.2 Liniowy model ciągły

Zabieg linearyzacji polega na utworzeniu modelu liniowego rozważanego obiektu sterowania wokół określonego punktu pracy. Z racji tego, że model liniowy charakteryzuje się stałym wzmocnieniem, stanowi on dobre przybliżenie modelu nieliniowego jedynie w pewnym otoczeniu punktu pracy. Zabieg ten powoduje, że tracimy na dokładności modelu, którym się posługujemy, jednak zyskujemy na znacznym uproszczeniu reguł sterowania, ponieważ teoria sterowania stacjonarnych obiektów liniowych ma silne dowody stabilności.

Jedną z powszechnie stosowanych metod linearyzacji jest linearyzacja lokalna. Jeżeli funkcja w przestrzeni n wymiarowej R^n jest różniczkowalna i ma w otoczeniu danego punktu x_0 ciągle pochodne względem swoich argumentów, można dokonać lokalnego przybliżenia wartości $f(x)$ w punkcie x bliskim x_0 w postaci zależności:

$$f_{lin}(x) = f(x_0) + \left. \frac{df(x)}{dx} \right|_{x=x_0} \cdot (x - x_0) \quad (2)$$

Linearyzację należy rozpocząć od przekształcenia równań obiektu. Równania powinny zostać doprowadzone do postaci, w której po lewej stronie występuje pochodna zmiennej, a po prawej stronie zmienna ta występuje jawnie. Zatem dokonując odpowiednich przekształceń algebraicznych otrzymane zostały następujące nieliniowe równania stanu obiektu:

$$\begin{cases} \frac{dV_1}{dt} = F_1 + F_D - \alpha_1 \cdot \sqrt{\frac{V_1}{A_1}} \\ \frac{dV_2}{dt} = \alpha_1 \cdot \sqrt{\frac{V_1}{A_1}} - \alpha_2 \cdot \sqrt[4]{\frac{V_2}{C_2}} \end{cases} \quad (3)$$

Pełny opis obiektu należy uzupełnić jeszcze o równanie wyjścia, które jest następujące:

$$h_2 = \sqrt{\frac{V_2}{C_2}} \quad (4)$$

Oznaczając powyższe funkcje w następujący sposób:

$$\begin{cases} f_1(F_1, V_1, V_2) = \frac{dV_1}{dt} \\ f_2(F_1, V_1, V_2) = \frac{dV_2}{dt} \\ g(F_1, V_1, V_2) = h_2 \end{cases} \quad (5)$$

zostały wyznaczone analitycznie pochodne cząstkowe funkcji:

$$\begin{aligned} \frac{df_1}{dF_1} &= 1 \\ \frac{df_1}{dV_1} &= -\frac{\alpha_1}{2} \cdot \sqrt{\frac{1}{A_1 \cdot V_1}} \\ \frac{df_1}{dV_2} &= 0 \end{aligned} \quad (6)$$

$$\begin{aligned}
\frac{df_2}{dF_1} &= 0 \\
\frac{df_2}{dV_1} &= \frac{\alpha_1}{2} \cdot \sqrt{\frac{1}{A_1 \cdot V_1}} \\
\frac{df_2}{dV_2} &= -\frac{\alpha_2}{4} \cdot \sqrt[4]{\frac{1}{C_2 \cdot V_2^3}} \\
\frac{dg}{dV_1} &= 0 \\
\frac{dg}{dV_2} &= \frac{1}{2} \cdot \sqrt{\frac{1}{C_2 \cdot V_2}}
\end{aligned} \tag{6}$$

Model liniowy przyrostowy w postaci równania stanu - równanie wyjścia jest postaci:

$$\begin{aligned}
\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \\
\mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}
\end{aligned} \tag{7}$$

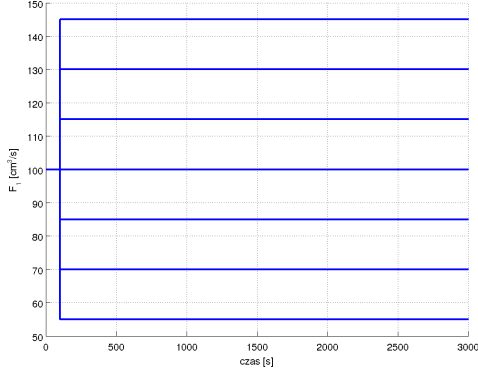
gdzie:

$$\begin{aligned}
\mathbf{x} &= \begin{bmatrix} V_1 - V_{10} \\ V_2 - V_{20} \end{bmatrix}, & \mathbf{u} &= \begin{bmatrix} F_1 - F_{10} \end{bmatrix} \\
\mathbf{y} &= \begin{bmatrix} h_2 - h_{20} \end{bmatrix}
\end{aligned} \tag{8}$$

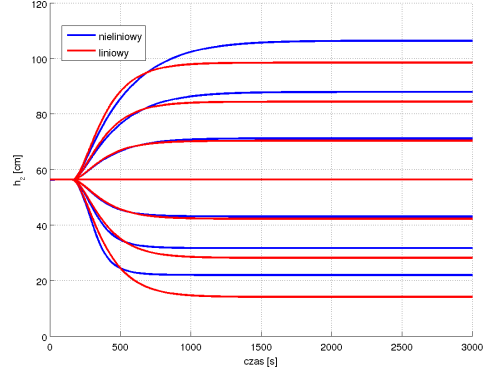
Macierze \mathbf{A} , \mathbf{B} i \mathbf{C} składają się więc z następujących elementów (dla uproszczenia zapisu przyjęto oznaczenie $p.p.$ jako warunek zdefiniowany przez punkt pracy podany w treści polecenia):

$$\begin{aligned}
\mathbf{A} &= \begin{bmatrix} \left. \frac{\partial f_1}{\partial V_1} \right|_{p.p.} & \left. \frac{\partial f_1}{\partial V_2} \right|_{p.p.} \\ \left. \frac{\partial f_2}{\partial V_1} \right|_{p.p.} & \left. \frac{\partial f_2}{\partial V_2} \right|_{p.p.} \end{bmatrix} \\
\mathbf{B} &= \begin{bmatrix} \left. \frac{\partial f_1}{\partial F_1} \right|_{p.p.} \\ \left. \frac{\partial f_2}{\partial F_1} \right|_{p.p.} \end{bmatrix} \\
\mathbf{C} &= \begin{bmatrix} \left. \frac{\partial g}{\partial V_1} \right|_{p.p.} & \left. \frac{\partial g}{\partial V_2} \right|_{p.p.} \end{bmatrix} \\
\mathbf{D} &= \begin{bmatrix} 0 & 0 \end{bmatrix}
\end{aligned} \tag{9}$$

Rysunek 5 przedstawia wynik symulacji modelu liniowego na tle oryginalnych równań różniczkowych obiektu. Przy niewielkim oddalaniu się od punktu pracy model liniowy wciąż stosunkowo dokładnie oddaje zachowanie się obiektu, jednak przy znacznym oddaleniu się od punktu pracy nieliniowość obiektu jest coraz bardziej widoczna, a model liniowy coraz bardziej odbiega od rzeczywistości.



(a) Zmiany natężenia dopływającego strumienia sterującego.



(b) Odpowiedzi obiektu na skok wartości na wejściu.

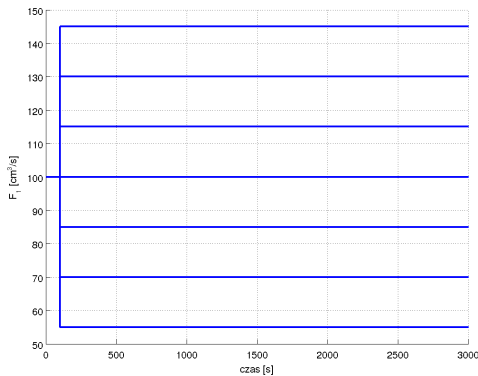
Rysunek 5: Porównanie modelu liniowego z oryginalnymi równaniami różniczkowymi obiektu.

2.3 Liniowy model dyskretny

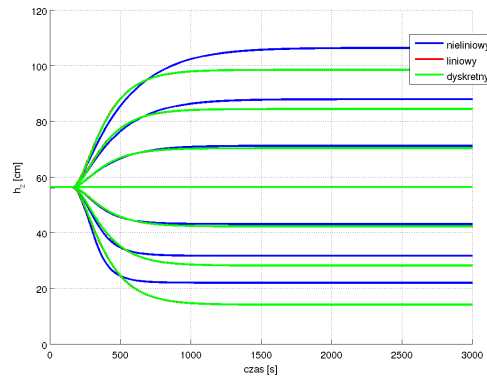
Kolejnym celem zadania będzie sterowanie obiektem za pomocą algorytmu predykcyjnego GPC, dlatego model musi zostać przekształcony do postaci dyskretnego równania różnicowego. Ponieważ obiekt jest dość wolny, przyjęty został dłuższy okres próbkowania: $T_p = 5s$. Model dyskretny został uzyskany za pośrednictwem narzędzi wbudowanych Matlaba, tj. zestawu funkcji `ss2tf` i `c2d`. Wykorzystana została metoda dyskretyzacji bazująca na ekstrapolatorze zerowego rzędu. Uzyskane równanie różnicowe (o przybliżonych wartościach numerycznych współczynników) jest następujące:

$$y(k) = 1.913202047891722 \cdot y(k-1) - 0.914885618196487 \cdot y(k-2) + 0.000800873041280923 \cdot u(k-13) + 0.000777474119435667 \cdot u(k-14) \quad (10)$$

Wyniki tak uzyskanego modelu dyskretnego zostały przedstawione na tle równań różniczkowych obiektu i ciągłego modelu liniowego na wykresie 6.



(a) Zmiany natężenia dopływającego strumienia sterującego.



(b) Odpowiedzi obiektu na skok wartości na wejściu.

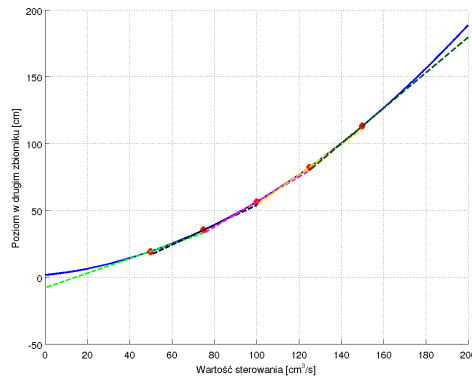
Rysunek 6: Porównanie modelu dyskretnego z liniowym ciągłym i oryginalnymi równaniami różniczkowymi obiektu.

2.4 Model rozmyty typu Takagi-Sugeno

Jednym z podejść do tworzenia modeli nieliniowych jest koncepcja modeli rozmytych Takagi-Sugeno. Regulatory oparte o modele rozmyte okazują się silnym narzędziem, zwłaszcza w aplikacjach, gdzie trudno jest uzyskać pełny, analityczny opis obiektu sterowanego. W modelach rozmytych określa się stopień przynależności elementu x do zbiorów rozmytych. W przeciwieństwie do podejścia klasycznego, w którym element należał do jakiegoś zbioru lub nie (przynależność ostra), tutaj element może należeć do kilku zbiorów jednocześnie w sposób określony funkcją przynależności. Zmienna pośrednia między zmienną numeryczną a zmienną symboliczną określana jest mianem *zmiennej lingwistycznej*.

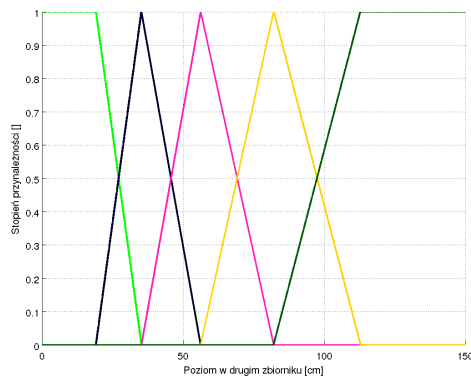
Modele rozmyte typu Takagi-Sugeno charakteryzują się regułami z następnikami funkcyjnymi, z których najczęściej wykorzystuje się wielomiany pierwszego rzędu. Dzięki takiej postaci można w całkiem dokładny sposób modelować obiekt wykorzystując niewielki zbiór reguł. Ponadto dodatkowe korzyści płyną z faktu, że istnieje szereg metod identyfikacji i konstrukcji modeli liniowych, co pozwala w stosunkowo łatwy sposób stworzyć model rozmyty, który z powodzeniem można wykorzystać w regulatorze.

W pracy przyjęty został zakres zmian wartości sterowania $F_1 \in < 50, 150 >$, wobec tego przyjętych zostało pięć modeli lokalnych, ułożonych w punktach odpowiadających wartościom sterowania $F_1 \in \{50, 75, 100, 125, 150\}$. Zmienną lingwistyczną jest poprzednie wyjście obiektu, czyli $y(k-1)$. Ułożenie modeli lokalnych wraz z ich wzmocnieniami przedstawiono na tle charakterystyki statycznej obiektu na rysunku 7.



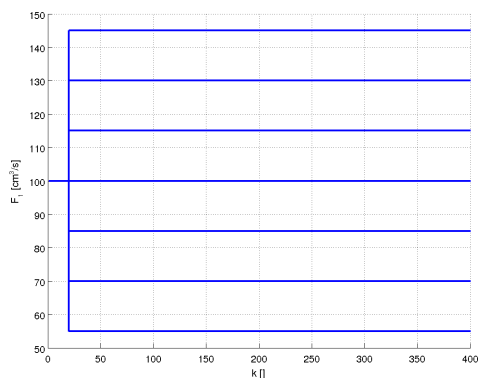
Rysunek 7: Charakterystyka statyczna obiektu wraz z charakterystykami modeli lokalnych modelu rozmytego.

Początkowo funkcje przynależności zostały wybrane jako proste funkcje trapezowe i trójkątne. Ich postaci przedstawia rysunek 8. Wzory funkcji zostały dopasowane tak, aby modele lokalne przełączane były w sposób liniowy, wartość 1 jest osiągnięta w punkcie pracy modelu lokalnego. W ten sposób co najwyżej dwa modele lokalne są aktywne jednocześnie.

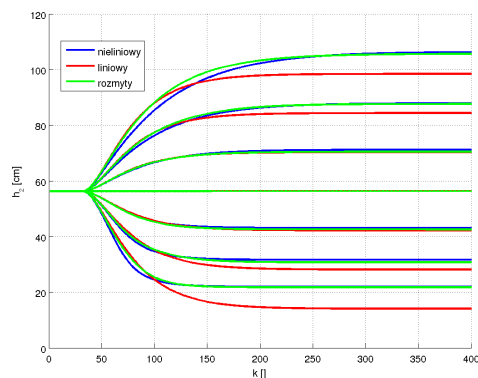


Rysunek 8: Funkcje przynależności wstępnego modelu rozmytego.

Wyniki symulacji modelu rozmytego zostały przedstawione na tle modelu liniowego i rzeczywistych danych obiektu na rysunku 9. W modelu rozmytym wykorzystywane są informacje z innych obszarów, stąd występują zdecydowanie mniejsze uchyby ustalone w porównaniu z modelem liniowym. Ponadto zmienna jest także dynamika modelu rozmytego, stąd model rozmyty lepiej odwzorowuje obiekt rzeczywisty. Dopasowanie tak uzyskanego modelu rozmytego do obiektu w zakładanym przedziale działania powinno być wystarczające do wykorzystania w syntezie regulatorów rozmytych.



(a) Zmiany natężenia dopływającego strumienia sterującego.



(b) Odpowiedzi obiektu i modeli na skok wartości na wejściu.

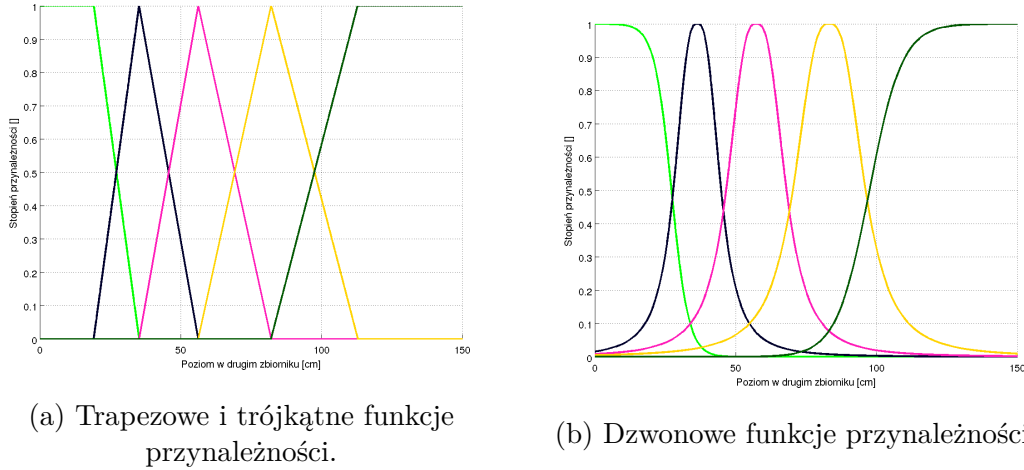
Rysunek 9: Porównanie modelu rozmytego z liniowym dyskretnym i oryginalnymi równaniami różniczkowymi obiektu.

2.5 Strojenie parametrów modelu rozmytego.

Warto sprawdzić, czy sposób przełączania modeli lokalnych może zostać poprawiony stosując inny kształt funkcji przynależności. Do tego celu zastosowano funkcje dzwonowe, które opisane są wzorem:

$$\mu(x) = \frac{1}{1 + \left| \frac{x-c}{b} \right|^{2a}} \quad (11)$$

w której parametr c odpowiada za środek symetrii dzwonu, a parametry a i b za kształt funkcji. Dzwonowe funkcje przynależności zostały dopasowane do przebiegów trójkątnych i prostokątnych za pomocą narzędzia `cftool` programu Matlab. Rysunek 10 przedstawia porównanie zbiorów funkcji przynależności trójkątnych do dzwonowych. Warto wspomnieć, że model rozmyty w wersji z dzwonowymi funkcjami przynależności dalej dysponuje tymi samymi modelami lokalnymi.

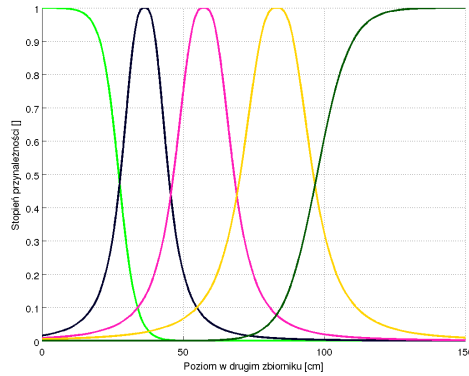


Rysunek 10: Porównanie dwóch zestawów funkcji przynależności o innych kształtach.

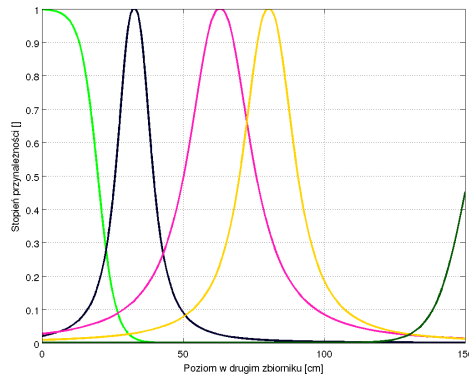
Kryterium jakości zadania optymalizacji stanowiła suma kwadratów błędów na wygenerowanej trajektorii wyjścia obiektu, gdzie przez błąd rozumiana była różnica pomiędzy wyjściem z obiektu i wyjściem z modelu rozmytego. Wektor zmiennych decyzyjnych zbudowany był z parametrów wszystkich kolejnych funkcji przynależności, a ograniczenia na zmienne decyzyjne zostały nałożone w następujący sposób:

$$\begin{aligned} LB &= \left[\min\{0.8a^i, 1.2a^i\}; \min\{0.8b^i, 1.2b^i\}; \min\{0.9c^i, 1.1c^i\} \right], \quad i = 1, \dots, 5 \\ UB &= \left[\max\{0.8a^i, 1.2a^i\}; \max\{0.8b^i, 1.2b^i\}; \max\{0.9c^i, 1.1c^i\} \right], \quad i = 1, \dots, 5 \end{aligned} \quad (12)$$

Wynikowe kształty funkcji przynależności po optymalizacji parametrów są przedstawione na rysunku 11. Krzywe dzwonowe uległy zarówno przesunięciom, jak i zmienił się ich kształt. Trajektorie wyjść obiektu i wyjść obydwu modeli rozmytych (wykorzystującego startowe postaci funkcji dzwonowych i postaci po optymalizacji) są przedstawione na rysunku 12. Można zauważyć, że przebieg wyjścia z modelu zoptymalizowanego nie zmienił się znacznie w porównaniu do modelu z początkowymi wartościami parametrów.

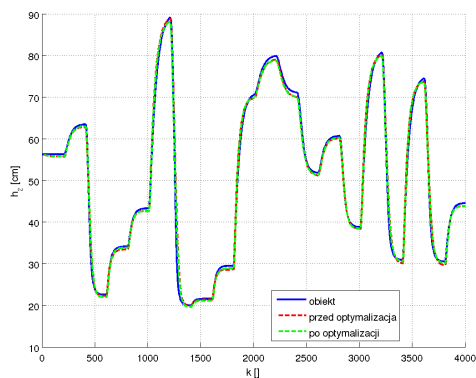


(a) Funkcje przynależności przed optymalizacją.



(b) Funkcje przynależności po optymalizacji.

Rysunek 11: Porównanie zestawów funkcji przynależności po optymalizacji.

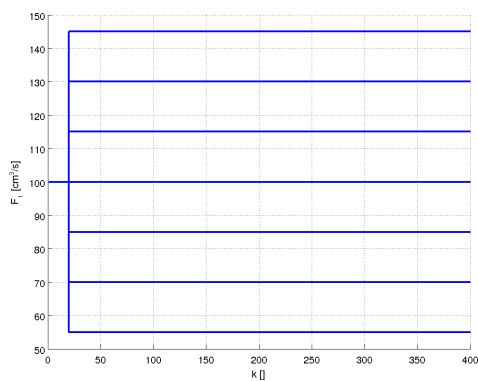


Rysunek 12: Porównanie zestawów funkcji przynależności po optymalizacji.

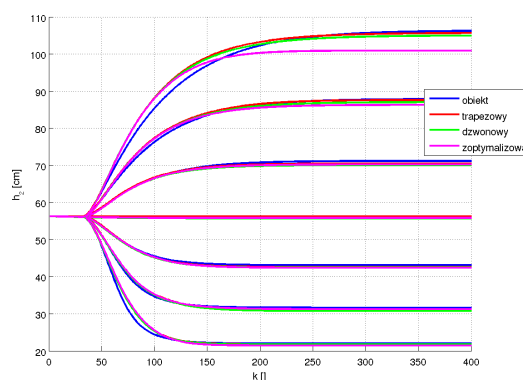
Ostatnim testem było porównanie wszystkich trzech modeli rozmytych:

- wykorzystującego trójkątne i trapezowe funkcje przynależności;
- wykorzystującego dzwonowe funkcje przynależności;
- wykorzystującego dzwonowe funkcje przynależności, po optymalizacji ich parametrów.

Symulacja zachowania wszystkich modeli na skoki wartości wejściowej przedstawiono na rysunku 13. Najbardziej wyraźną różnicę w działaniu można zaobserwować dla największej wielkości skoku. Najlepiej odwzorowuje obiekt model oparty na trapezowych funkcjach przynależności i ten model zostanie wykorzystany do syntezy regulatorów rozmytych.



(a) Skoki wartości podane na wejścia obiektu.



(b) Odpowiedzi obiektu i modeli.

Rysunek 13: Porównanie modeli rozmytych.