

Wydział Elektroniki i Technik Informatycznych  
Politechnika Warszawska

Sztuczna Inteligencja w Automatyce

Projekt 1

Kacper Marchlewicz, Adam Wróblewski

Warszawa, 2023

# Spis treści

<b>1. Zadanie 1</b>	2
1.1. Symulacja obiektu	2
1.2. Linearyzacja modelu	5
1.3. Dyskretyzacja modeli	8
1.4. Konwencjonalny regulator DMC w wersji analitycznej	11
<b>2. Zadanie 2</b>	14
2.1. Opracowanie modeli rozmytych Takagi-Sugeno	14
2.2. Implementacja rozmytego regulatora predykcyjnego	19
2.3. Nastrojony regulator DMC analityczny rozmyty	27
<b>3. Zadanie 3</b>	31
3.1. Regulator SL	31
3.2. Porównanie działania	34

# 1. Zadanie 1

## 1.1. Symulacja obiektu

Na podstawie podanych równań obiektu otrzymaliśmy nieliniowy model obiektu:

$$\frac{dh_1}{dt} = \frac{F1_{in}(t - \tau) + F_d - \alpha_1 \sqrt{h_1(t)}}{2C_1 h_1(t)} \quad (1.1)$$

$$\frac{dh_2}{dt} = \frac{\alpha_1 \sqrt{h_1(t)} - \alpha_2 \sqrt{h_2(t)}}{3C_2 h_2(t)^2} \quad (1.2)$$

Gdzie:

$F1_{in}$  - wielkość sterująca

$h_2$  - wielkość regulowana

$C_1 = 0,75$ .

$C_2 = 0,55$ .

$\alpha_1 = 20$ .

$\alpha_2 = 20$ .

Punkt pracy:

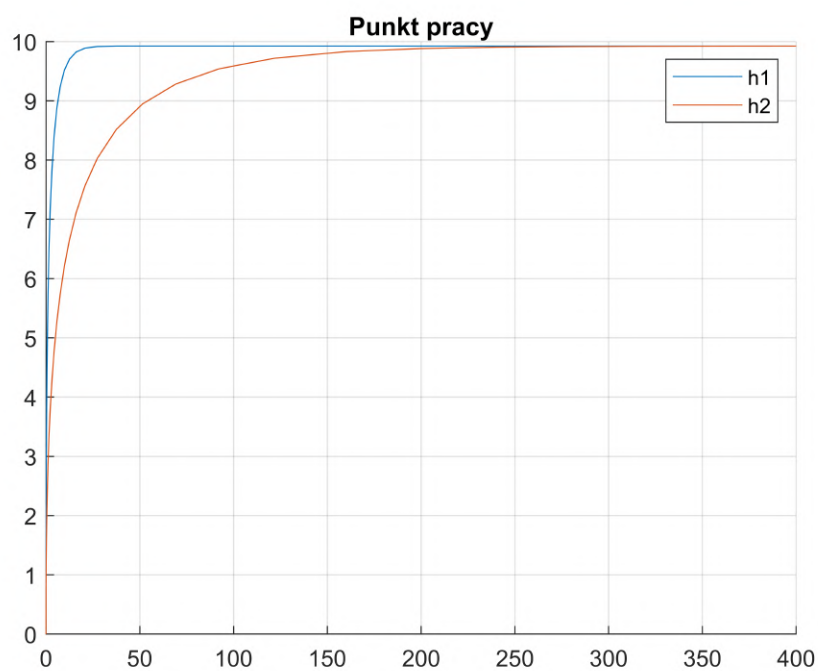
$F_1 = 52 cm^3/s$

$F_D = 11 cm^3/s$

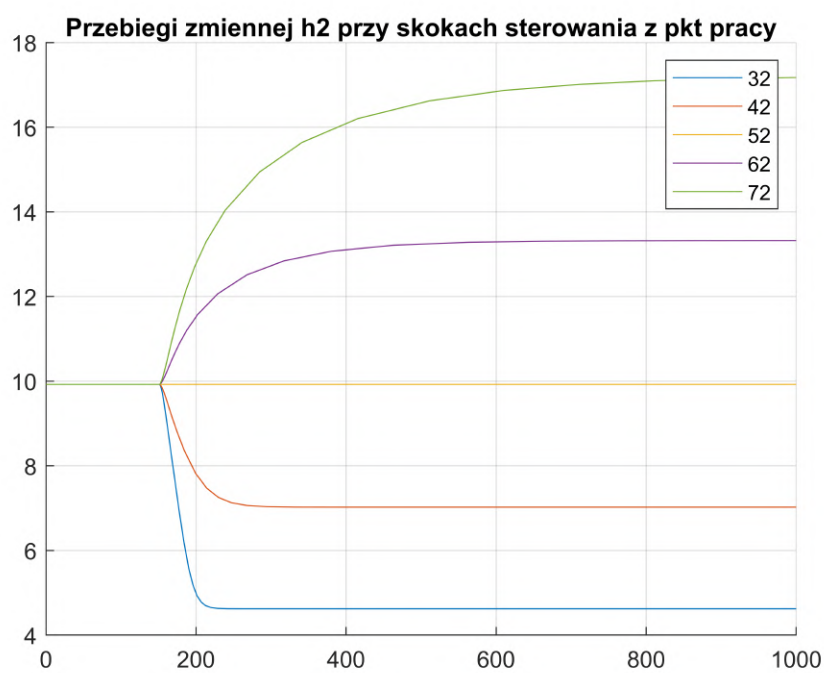
$\tau = 50s$

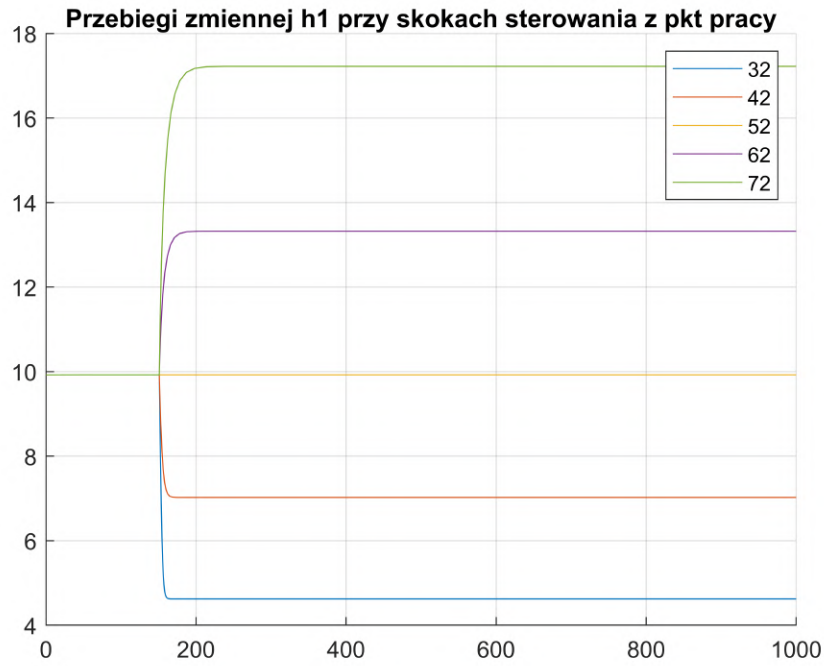
$h_2 = 9,9225 cm$

Symulację obiektu ciągłego przeprowadziliśmy wykorzystując funkcję *ode23s*. W punkcie pracy zmienna  $h_2$  stabilizuje się na wartości 9,9225, a  $h_1$  na 9,9218.



Rys. 1.1: Punkt pracy

Rys. 1.2: Przebieg zmiennej  $h_2$  dla skoków sterowania z punktu pracy



Rys. 1.3: Przebieg zmiennej  $h_1$  dla skoków sterowania z punktu pracy

Na podstawie przebiegów zmiennych  $h_2$  i  $h_1$  możemy stwierdzić że obiekt jest nieliniowy - obserwujemy że wyjście obiektu zmienia się nieproporcjonalnie do wejścia.

W pliku 'Zadanie1.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.

## 1.2. Linearyzacja modelu

Dokonaliśmy linearyzacji modelu wykorzystując rozwinięcie w szereg Taylora, otrzymaliśmy następujące równania:

$$h_1(t) = \frac{-\alpha_1 \sqrt{h_{1plin}} + F_d + F_1}{2C_1 h_{1plin}} + \frac{(\alpha_1 \sqrt{h_{1plin}} - 2F_d - 2F_1)}{4C_1 h_{1plin}^2} * (h_1(t) - h_{1plin})$$

$$h_2(t) = \frac{\alpha_1 \sqrt{h_{1plin}} - \alpha_2 \sqrt{h_{2plin}}}{3C_2 h_{2plin}^2} + \left( \frac{\alpha_1}{6C_2 \sqrt{h_{1plin}} h_{2plin}^2} \right) * (h_1(t) - h_{1plin}) +$$

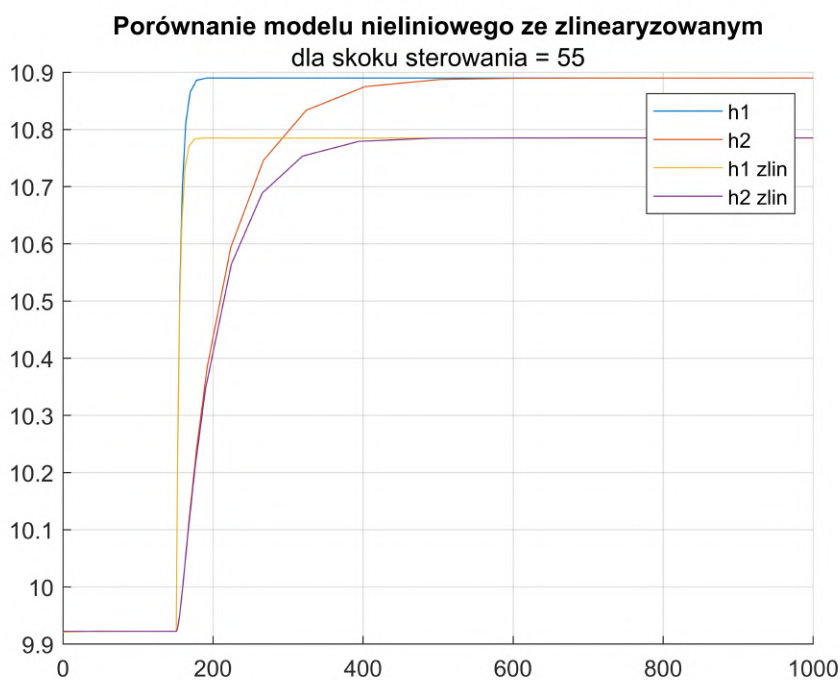
$$\left( \frac{-2\alpha_1 \sqrt{h_{1plin}}}{3C_2 h_{2plin}^3} + \frac{\alpha_2}{2C_2 h_{2plin}^{5/2}} \right) * (h_2(t) - h_{2plin});$$

Gdzie:

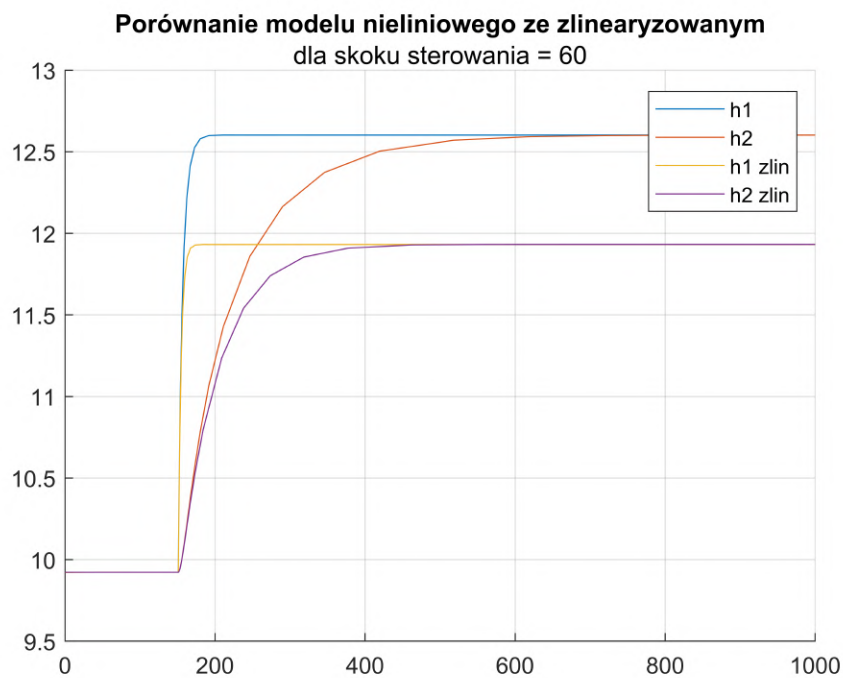
$h_{1plin}$  - punkt linearyzacji

$h_{2plin}$  - punkt linearyzacji

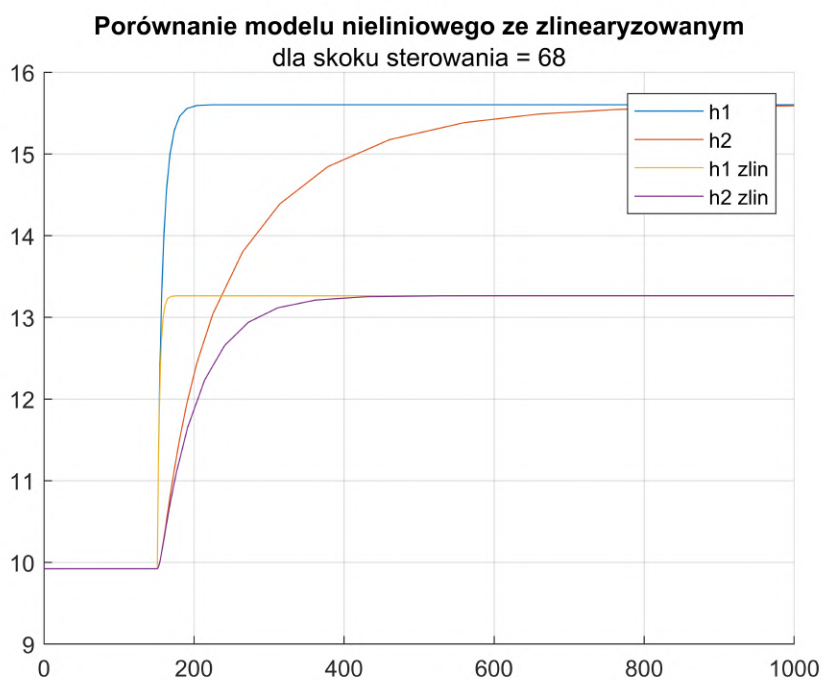
W celu porównania modeli linowych z nieliniowymi przeprowadziliśmy serię skoków wartości sterowania z punktu pracy:



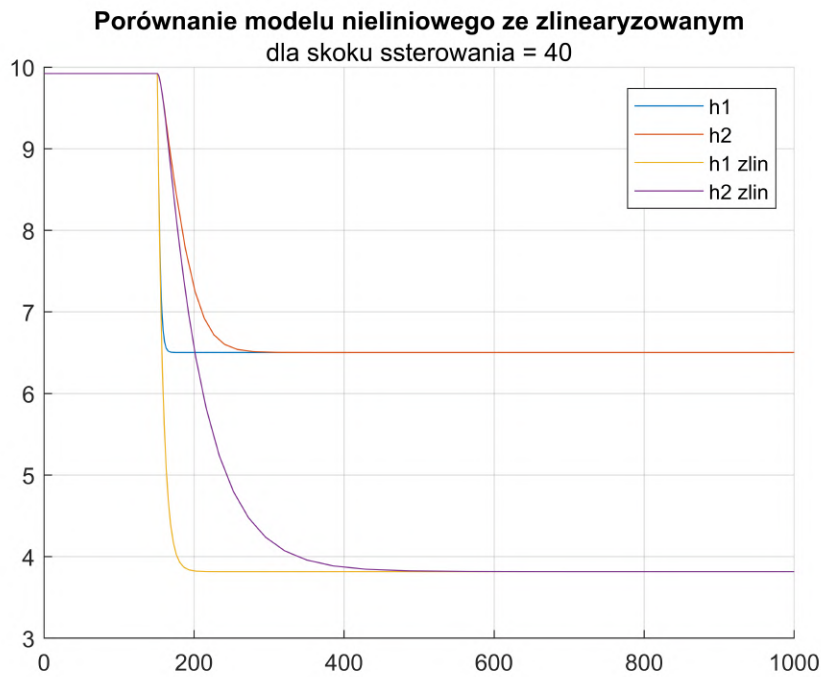
Rys. 1.4: Skok sterowania z punktu pracy na 55



Rys. 1.5: Skok sterowania z punktu pracy na 60



Rys. 1.6: Skok sterowania z punktu pracy na 68



Rys. 1.7: Skok sterowania z punktu pracy na 40

Wraz ze zwiększaniem sterowania oddalamy się od punktu pracy ( $h_1 = 9,9925$ ;  $h_2 = 9,9911$ ) dla którego obliczony został model zlinearyzowany, co obserwujemy jako sukcesywnie większe rozbieżności między modelem nieliniowym a zlinearyzowanym. Wraz z oddalaniem się od punktu linearyzacji (punktu pracy) tracimy dokładność modelu zlinearyzowanego.

W pliku 'Zadanie1.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.



### 1.3. Dyskretyzacja modeli

W celu implementacji regulatorów przekształciliśmy model ciągły na model dyskretny:

$$h_1(k) = \frac{F_{1in}(k-1-\tau) + F_d - \alpha_1 \sqrt{h_1(k-1)}}{2C_1 h_1(k-1)} * T + h_1(k-1) \quad (1.3)$$

$$h_2(k) = \frac{\alpha_1 \sqrt{h_1(k-1)} - \alpha_2 \sqrt{h_2(k-1)}}{3C_2 h_2(k-1)^2} * T + h_2(k-1) \quad (1.4)$$

Gdzie:  $T = 1$  - krok symulacji

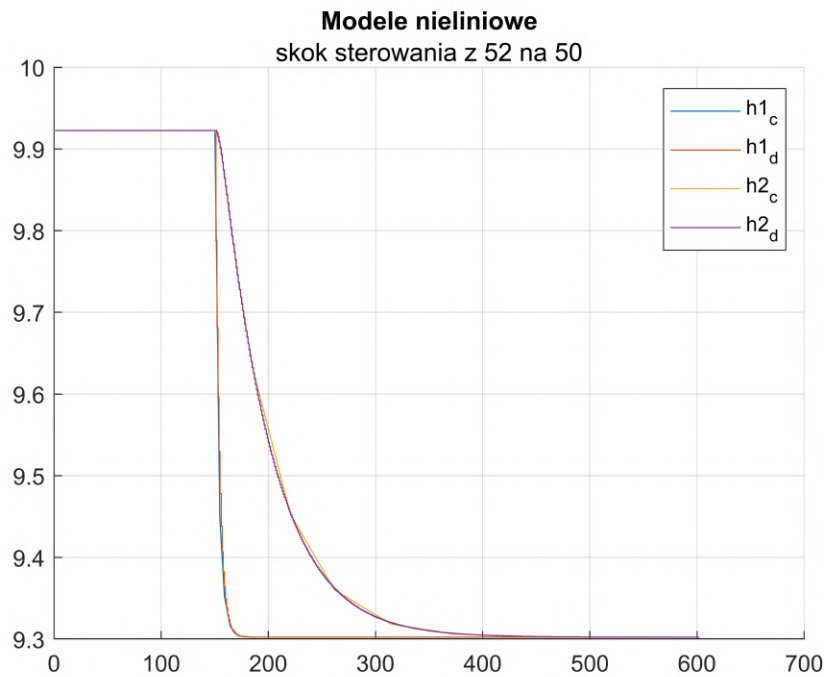
Model zlinearyzowany dyskretny:

$$h_1(k) = \frac{-\alpha_1 * \sqrt{h_{1plin}} + F_d + F_1}{2 * C_1 * h_{1plin}} + \frac{(\alpha_1 * \sqrt{h_{1plin}} - 2 * F_d - 2 * F_1)}{4 * C_1 * h_{1plin}^2} * (h_1(k-1) - h_{1plin}) * T + h_1(k-1) \quad (1.5)$$

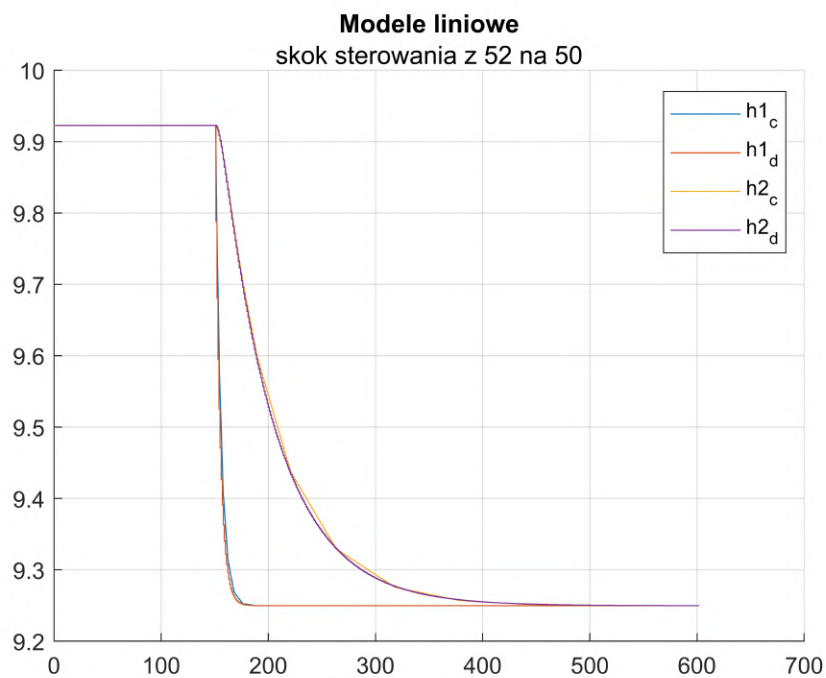
$$h_2 = \frac{\alpha_1 * \sqrt{h_{1plin}} - \alpha_2 * \sqrt{h_{2plin}}}{3 * C_2 * h_{2plin}^2} + \left( \frac{\alpha_1}{6 * C_2 * \sqrt{h_{1plin}} * h_{2plin}^2} \right) * (h(1) - h_{1plin}) +$$

$$\left( \frac{-2 * \alpha_1 * \sqrt{h_{1plin}}}{3 * C_2 * h_{2plin}^3} + \frac{\alpha_2}{2 * C_2 * h_{2plin}^{5/2}} \right) * (h_2(k-1) - h_{2plin}) * T + h_2(k-1) \quad (1.6)$$

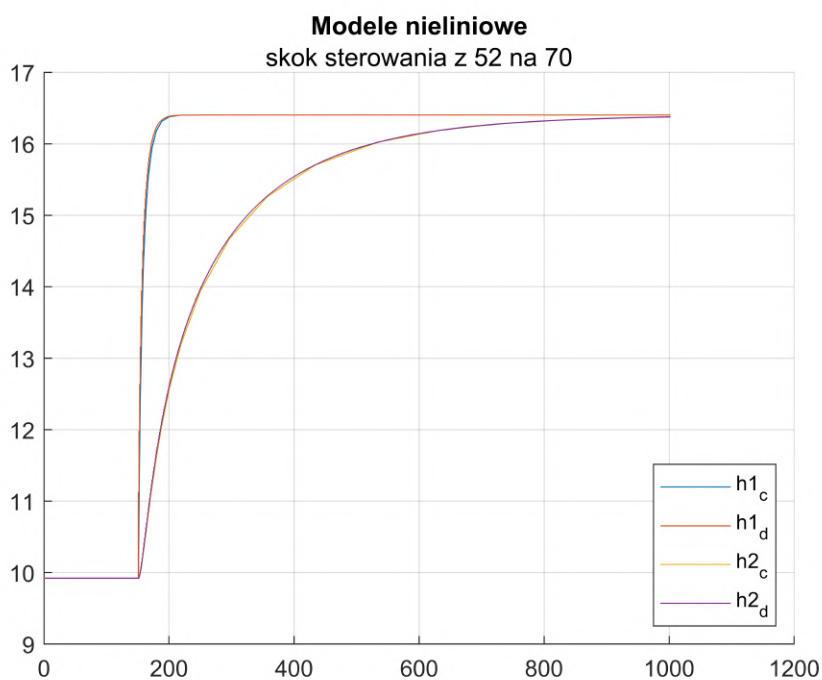
Porównanie modeli zdyskretyzowanych i ciągłych:



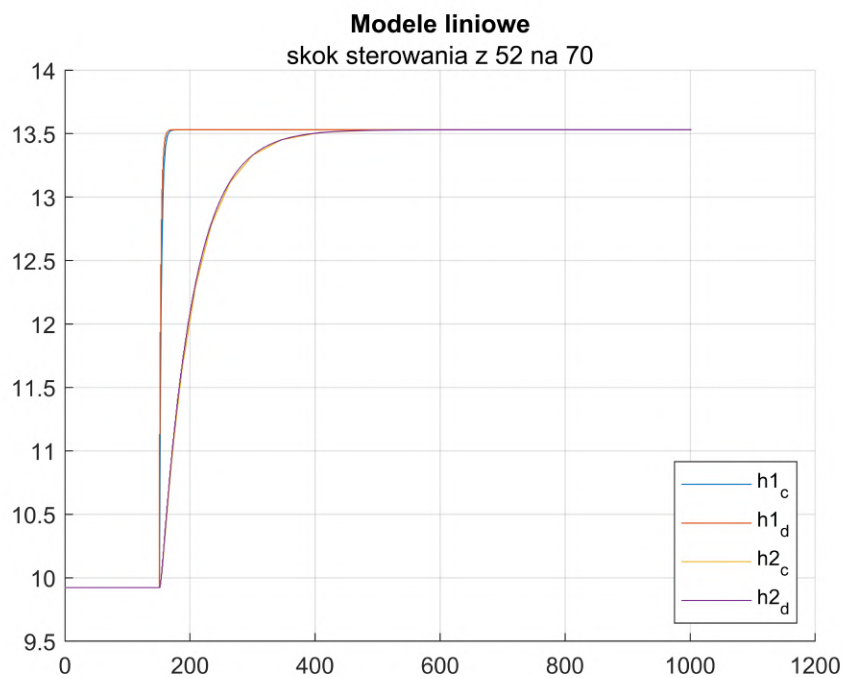
Rys. 1.8: Porównanie działania modelu nieliniowego ciągłego i dyskretnego.



Rys. 1.9: Porównanie działania modelu zlinearyzowanego ciągłego i dyskretnego.



Rys. 1.10: Porównanie działania modelu nieliniowego ciągłego i dyskretnego.



Rys. 1.11: Porównanie działania modelu zlinearyzowanego ciągłego i dyskretnego.

Jak możemy zauważyć modele dyskretnie dobrze odwzorowują modele ciągłe - przebiegi są identyczne.

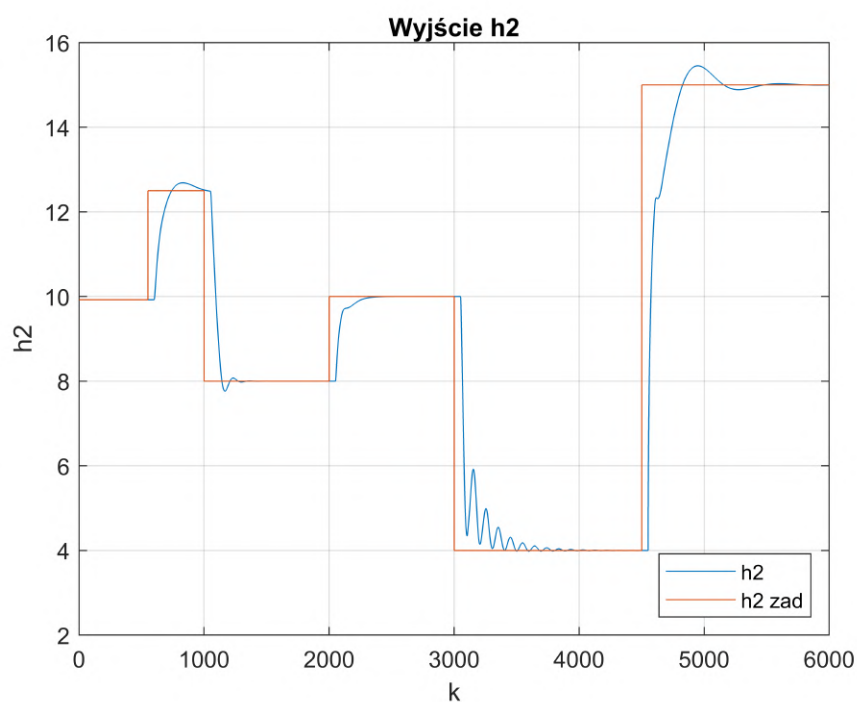
W pliku 'Zadanie1.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.

### 1.4. Konwencjonalny regulator DMC w wersji analitycznej

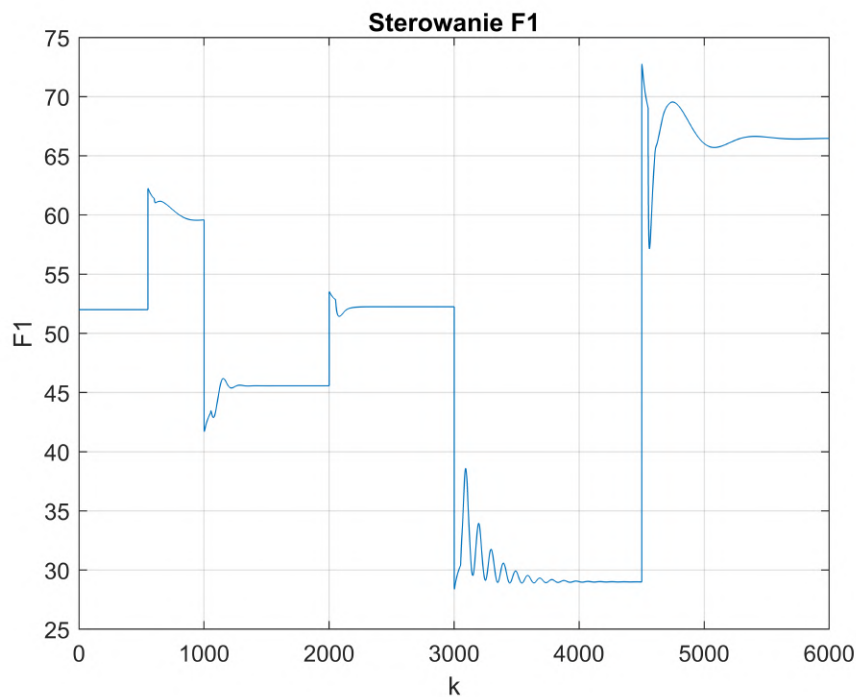
Do regulacji procesu wykorzystaliśmy zaimplementowany regulator DMC o parametrach:  $N = 300$ ,  $N_u = 2$ ,  $D = 500$ ,  $\lambda = 0.1$ .

Implementacja znajduje się w pliku 'DMC\_ana.m'. Pobiera on odpowiedzi skokowe za pomocą funkcji 'pobranie\_modelu.m'

Przebiegi sterowania i wyjścia dla kilku skoków wartości zadanej:



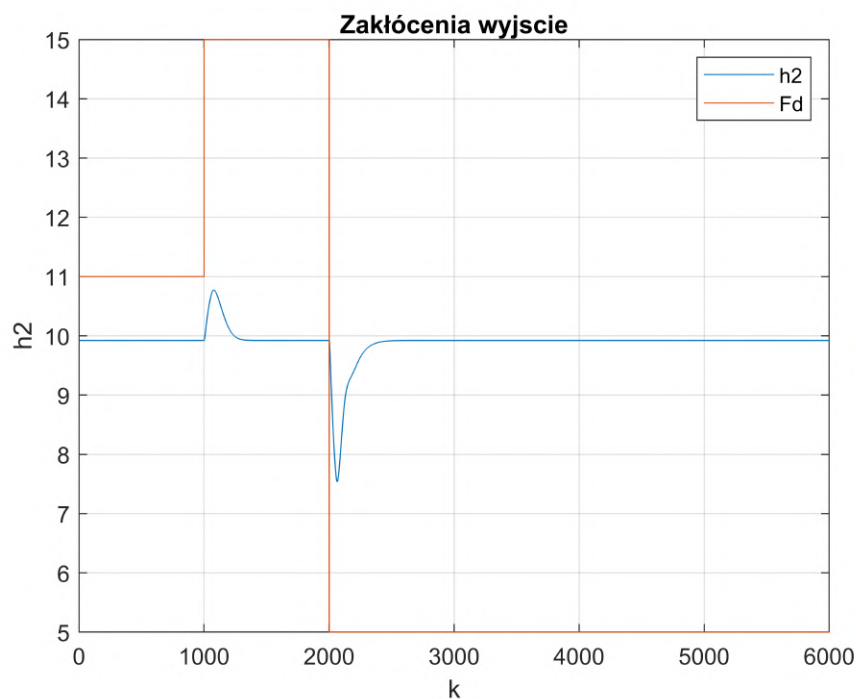
Rys. 1.12: Przebieg zmiennej wyjściowej h2 dla konwencjonalnego regulatora DMC.



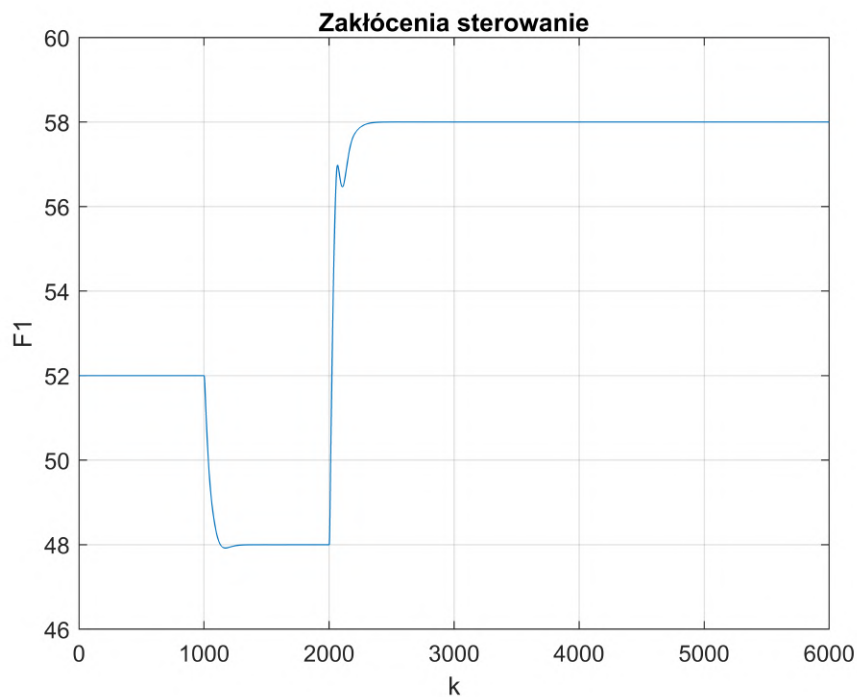
Rys. 1.13: Przebieg zmiennej sterującej  $F1$  dla konwencjonalnego regulatora DMC.

Regulator DMC w podstawowej wersji generalnie działa dobrze, jedynie w okolicy  $y_{zad} = 4$ , obserwujemy spore oscylacje wyjścia i sterowania. Błąd dla tego regulatora wynosi  $1,33 \cdot 10^4$ .

Wpływ zakłóceń na działanie regulatora DMC o parametrach:  
 $N = 300$ ,  $N_u = 2$ ,  $D = 500$ ,  $\lambda = 0.1$ .



Rys. 1.14: Przebieg zmiennej wyjściowej  $h2$  dla regulatora DMC przy skokowych zakłóceniach



Rys. 1.15: Przebieg zmiennej sterującej  $F1$  dla regulatora DMC przy skokowych zakłóceniach.

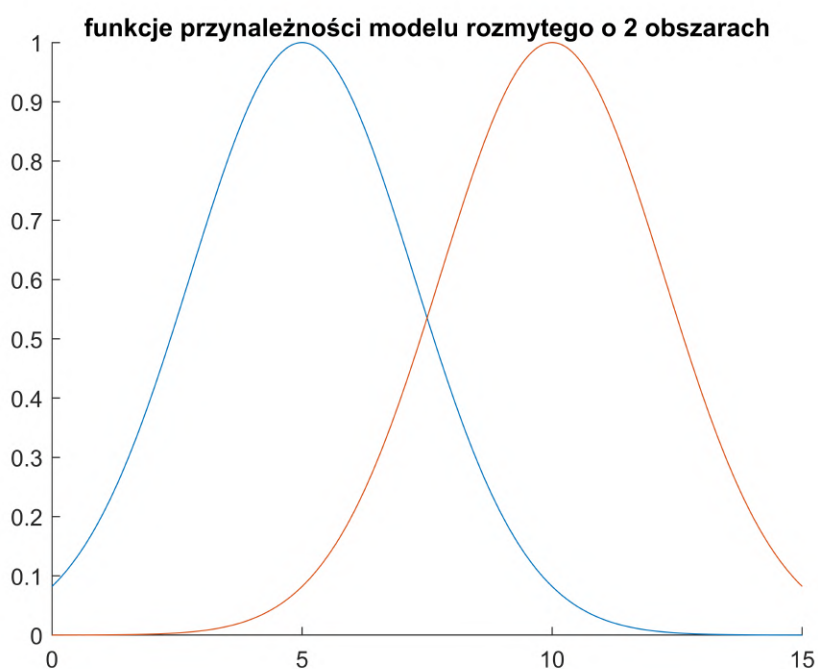
Regulator DMC jest odporny na zmianę wartości zakłóceń, obserwujemy że wraca on do wartości zadanej w skończonym czasie.

W pliku 'Zadanie1.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.

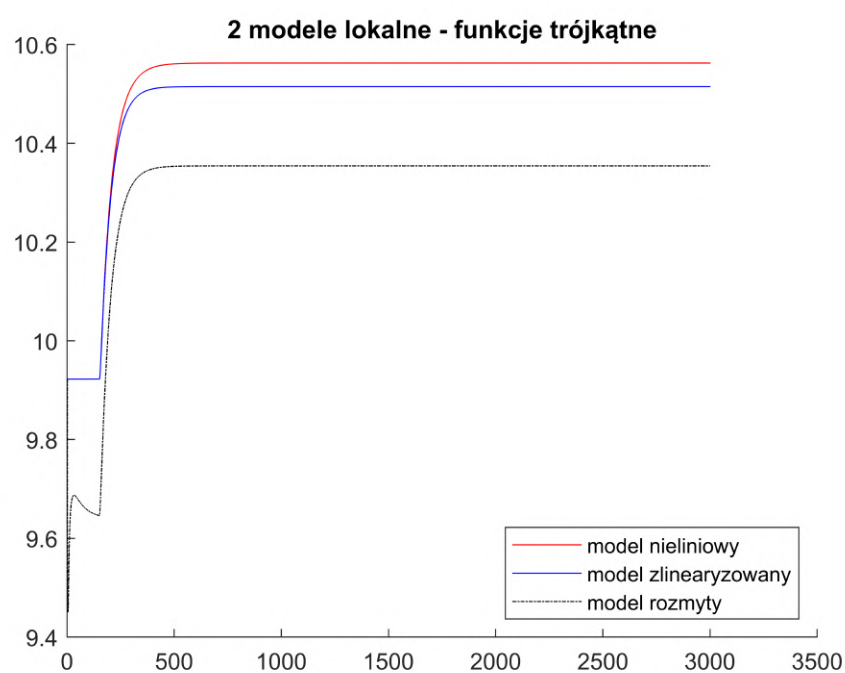
## 2. Zadanie 2

### 2.1. Opracowanie modeli rozmytych Takagi-Sugeno

Opracowaliśmy modele rozmyte dzieląc równomiernie przestrzeń zmiennej wyjściowej  $h_2 \in <0; 15>$  na kolejno 2,3,4,5 przedziałów. Linearyzacji modeli lokalnych dokonaliśmy w punkcie linearyzacji będącym w środku każdego z obszarów. Zmienna  $h_1$  przyjmuje takie same wartości w punktach linearyzacji jak  $h_2$  - uprzednio sprawdziliśmy eksperymentalnie czy osiągnięcie takich punktów przez model jest możliwe oraz czy zmienne  $h_1$  i  $h_2$  są sobie równe.

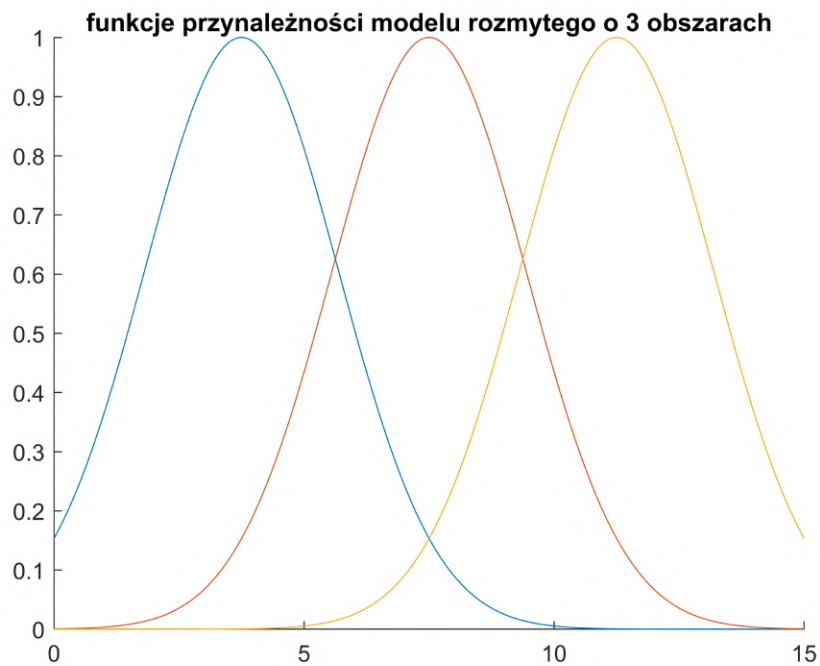


Rys. 2.1: Funkcje przynależności dla modelu o 2 obszarach.

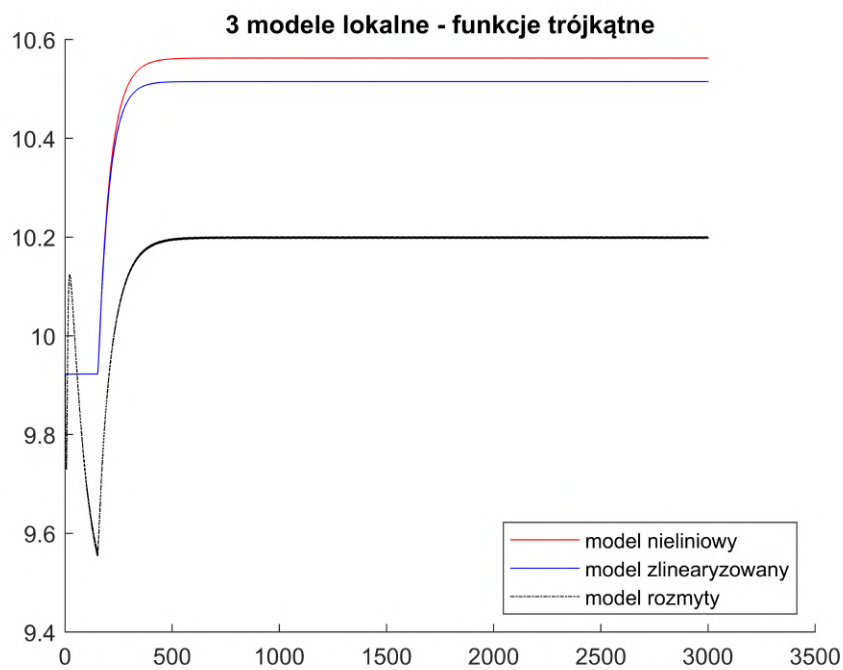


Rys. 2.2: Porównanie przebiegu zmiennej wyjściowej dla modelu rozmytego o 2 obszarach.

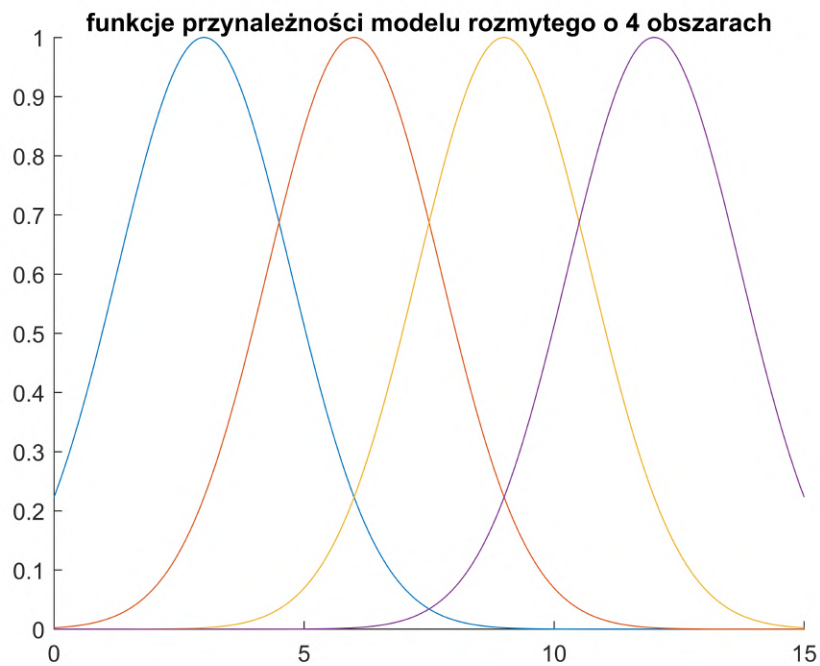




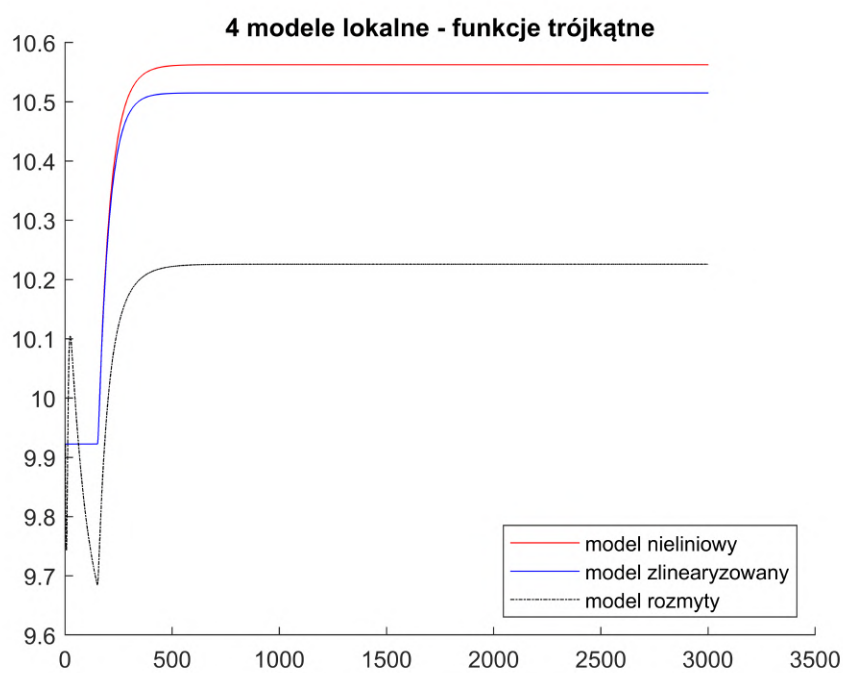
Rys. 2.3: Funkcje przynależności dla modelu o 3 obszarach.



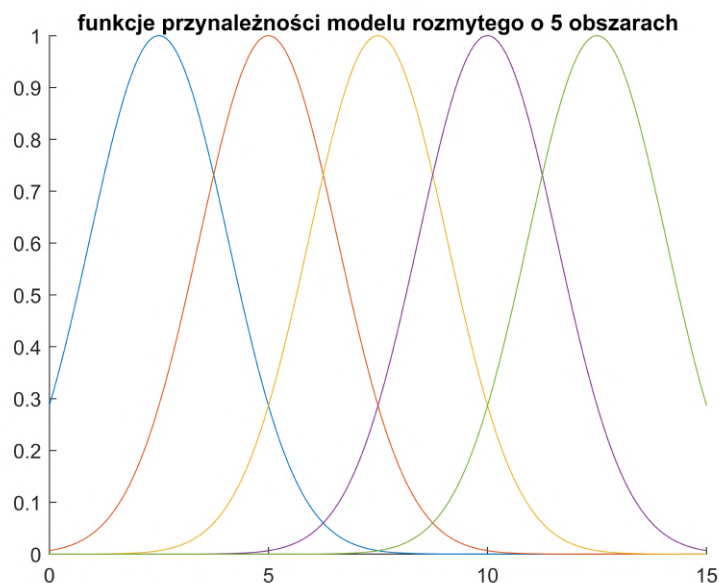
Rys. 2.4: Porównanie przebiegu zmiennej wyjściowej dla modelu rozmytego o 3 obszarach.



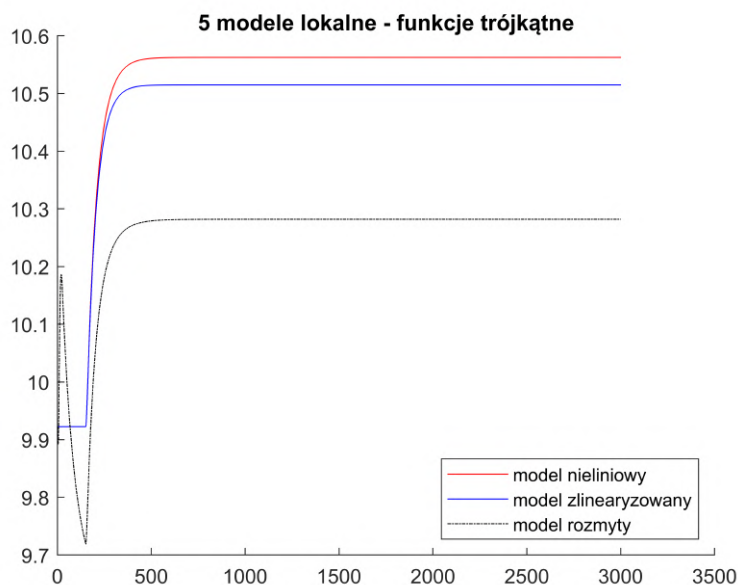
Rys. 2.5: Funkcje przynależności dla modelu o 4 obszarach.



Rys. 2.6: Porównanie przebiegu zmiennej wyjściowej dla modelu rozmytego o 4 obszarach.



Rys. 2.7: Funkcje przynależności dla modelu o 5 obszarach.



Rys. 2.8: Porównanie przebiegu zmiennej wyjściowej dla modelu rozmytego o 5 obszarach.

Nie możemy powiedzieć że modele rozmyte działają dobrze lub że działają źle - działają średnio. Obserwujemy pewien "uchyb" zależny od ilości użytych modeli lokalnych. Niestety nie jesteśmy w stanie zidentyfikować co jest przyczyną takiego działania - podejrzewamy błąd we wzorach którego nie znaleźliśmy.

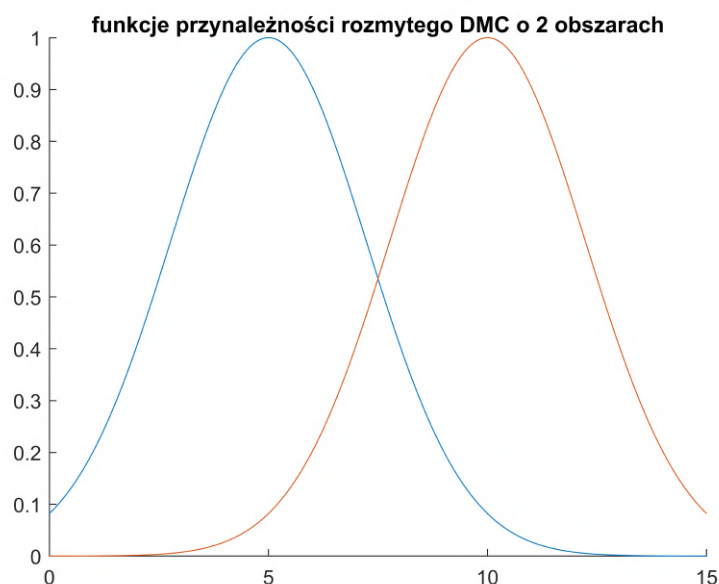
Dalsze badanie modelowania rozmytego - zmiana funkcji przynależności w naszej ocenie mija się z celem, gdy obserwujemy problemy w działaniu modelu przy równomiernie rozłożonych funkcjach przynależności.

W pliku 'modele\_rozmyte.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.

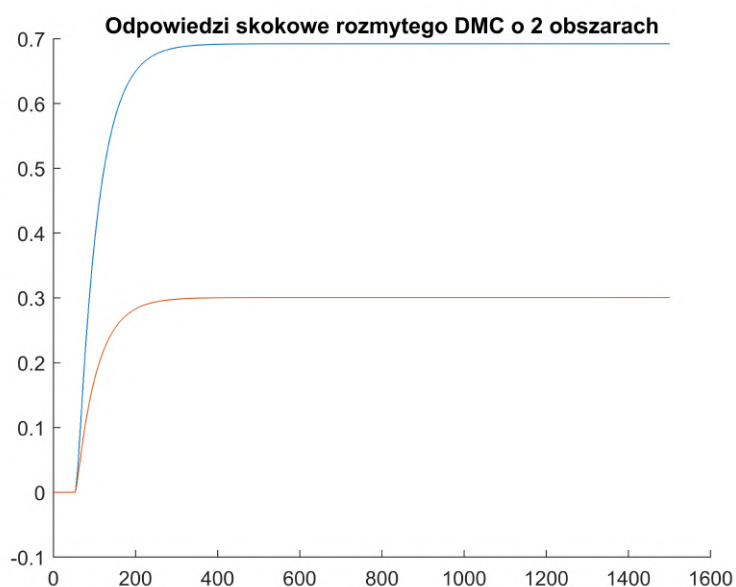
## 2.2. Implementacja rozmytego regulatora predykcyjnego

Zaimplementowaliśmy rozmyty regulator DMC zakładając początkowy kształt funkcji przynależności oparty na równomiernym podziale.

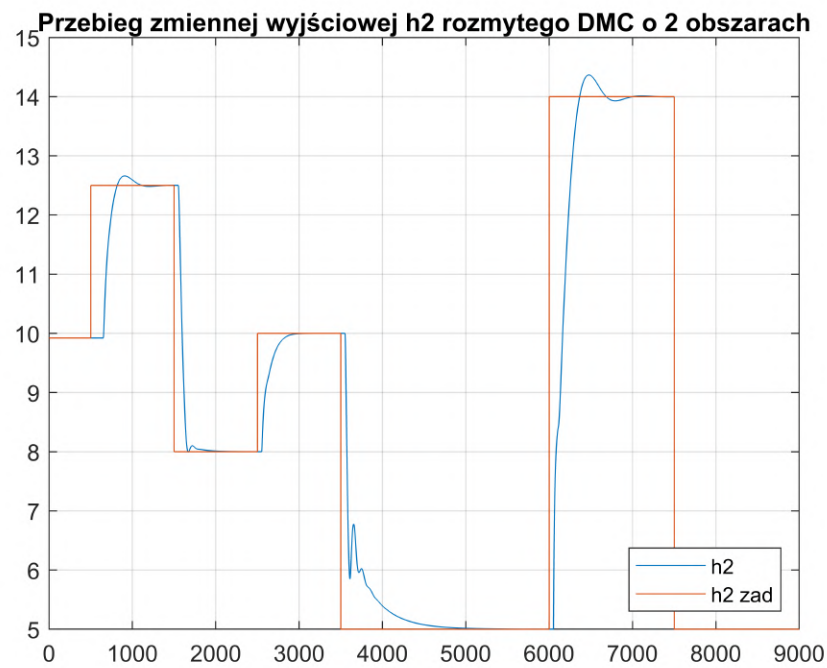
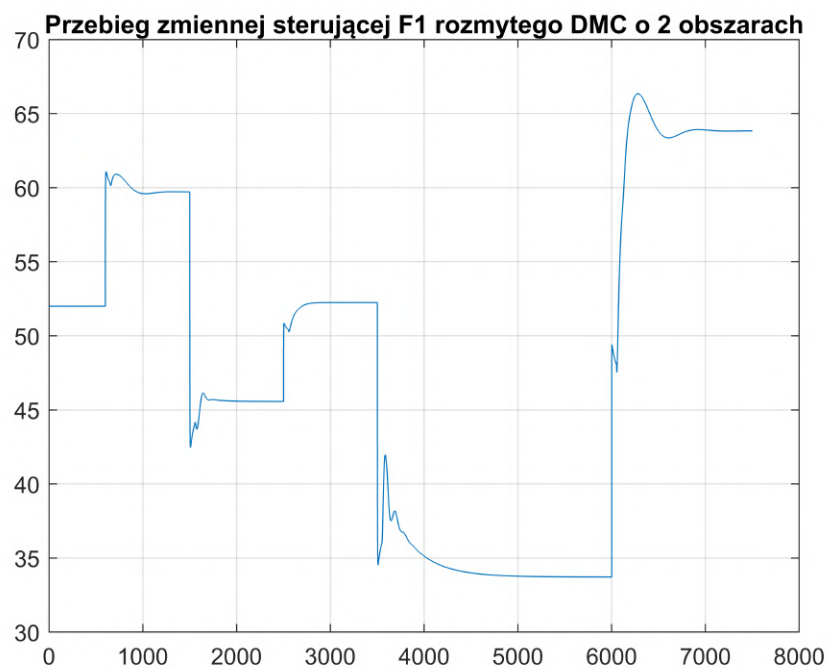
Przeprowadziliśmy analizę działania regulatora rozmytego dla kolejno 2, 3, 4, 5 obszarów. Dla 2 obszarów:



Rys. 2.9: Funkcje podziału dla DMC rozmytego o 2 obszarach.

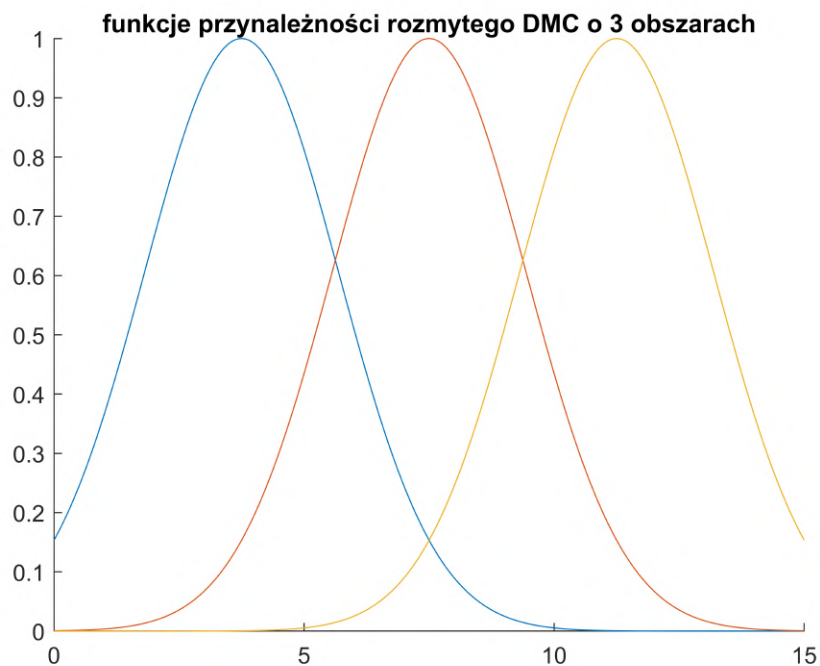


Rys. 2.10: odpowiedzi skokowe DMC rozmytego o 2 obszarach.

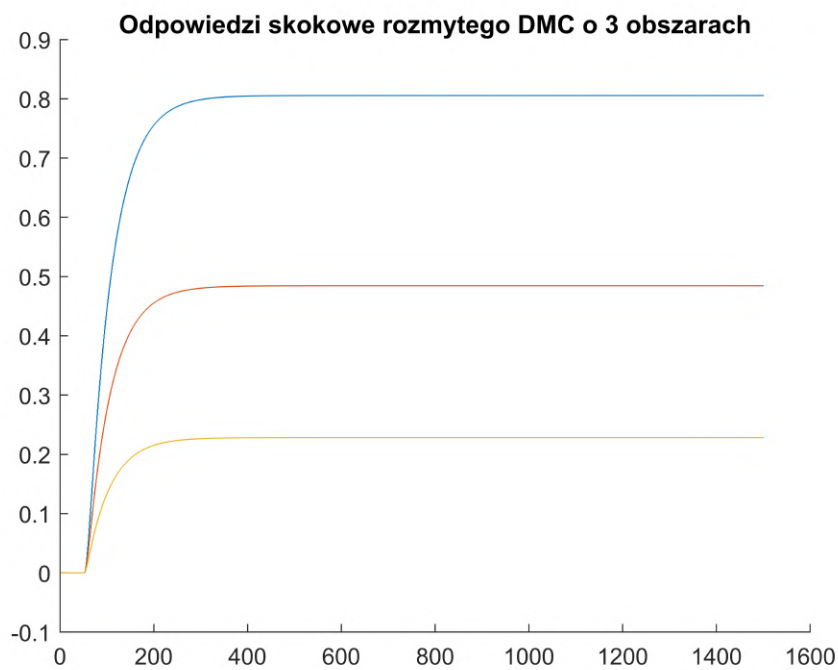
Rys. 2.11: Przebieg zmiennej wyjściowej  $h_2$  DMC rozmytego o 2 obszarach.

Rys. 2.12: Przebieg sterowania dla DMC rozmytego o 2 obszarach.

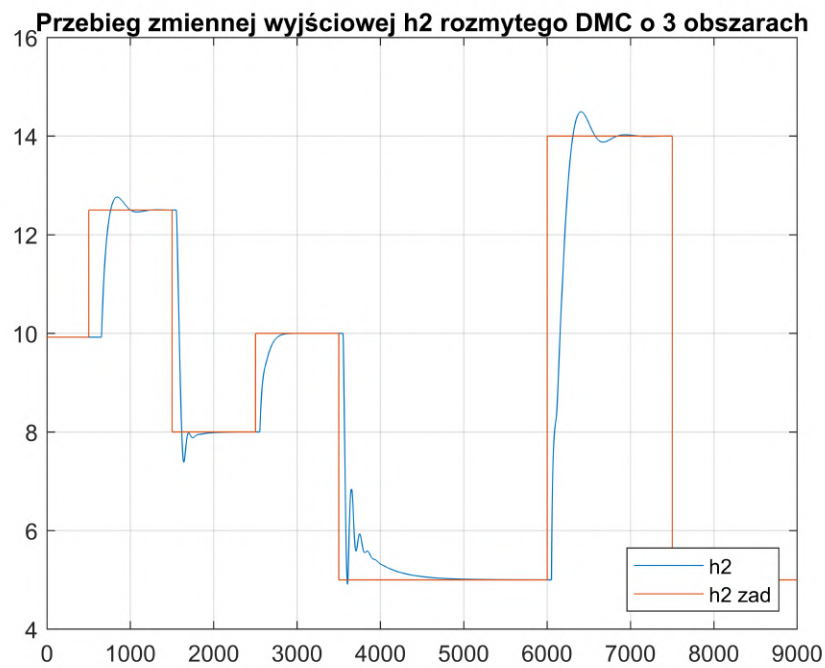
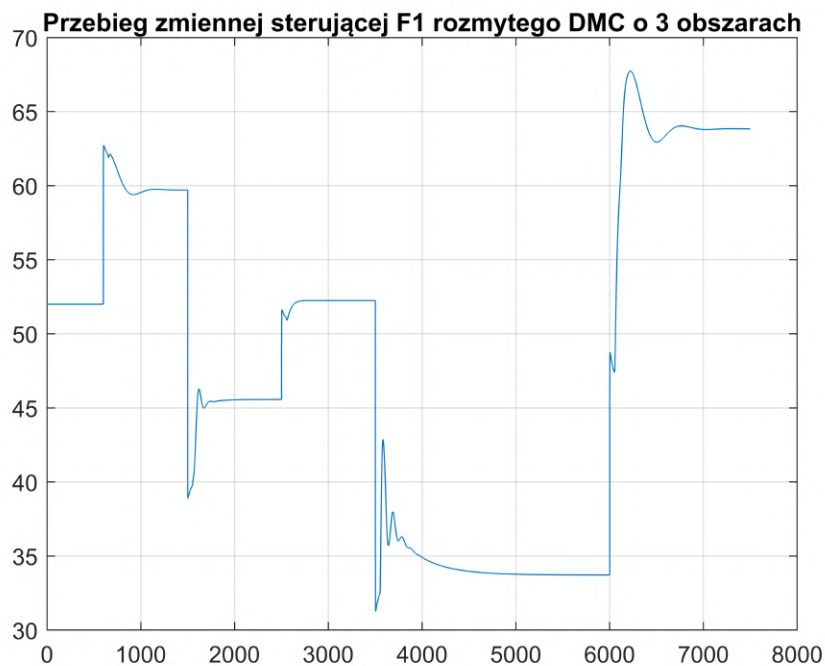
Dla 3 obszarów:



Rys. 2.13: Funkcje podziału dla DMC rozmytego o 3 obszarach.

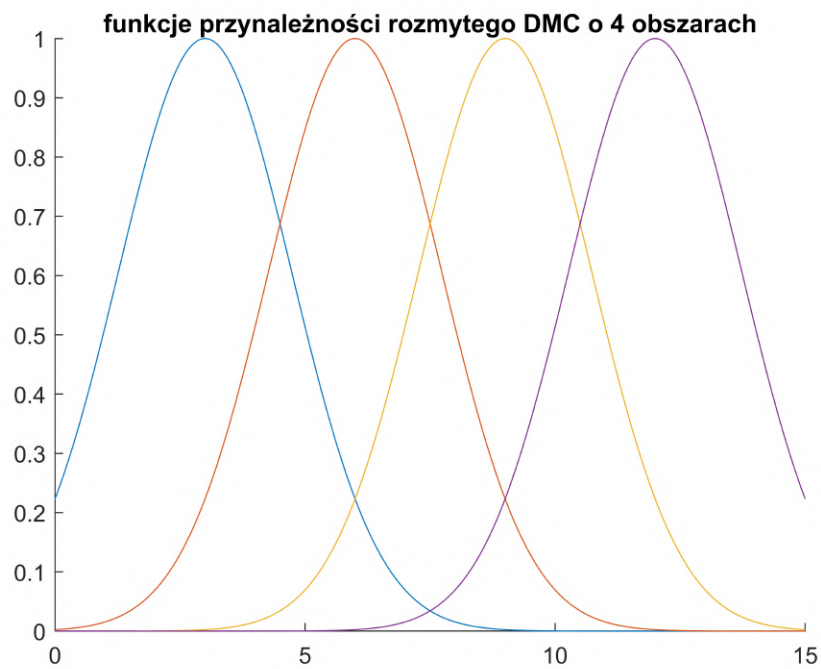


Rys. 2.14: odpowiedzi skokowe DMC rozmytego o 3 obszarach.

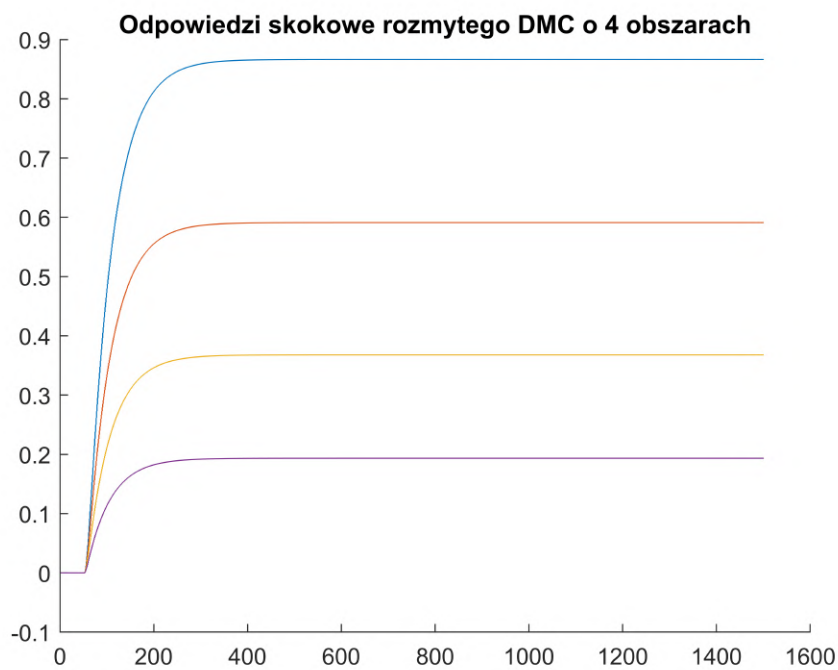
Rys. 2.15: Przebieg zmiennej wyjściowej  $h_2$  DMC rozmytego o 3 obszarach.

Rys. 2.16: Przebieg sterowania dla DMC rozmytego o 3 obszarach.

Dla 4 obszarów:

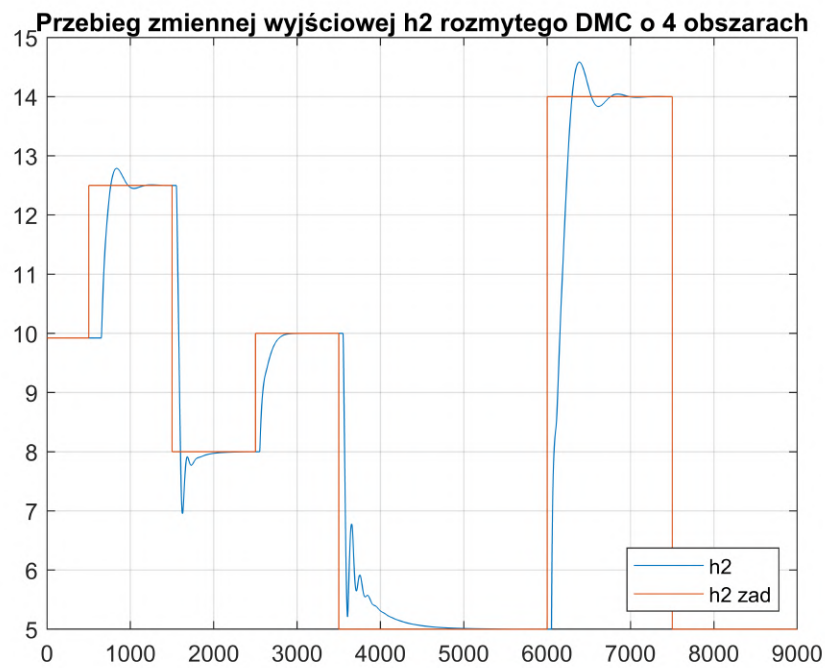
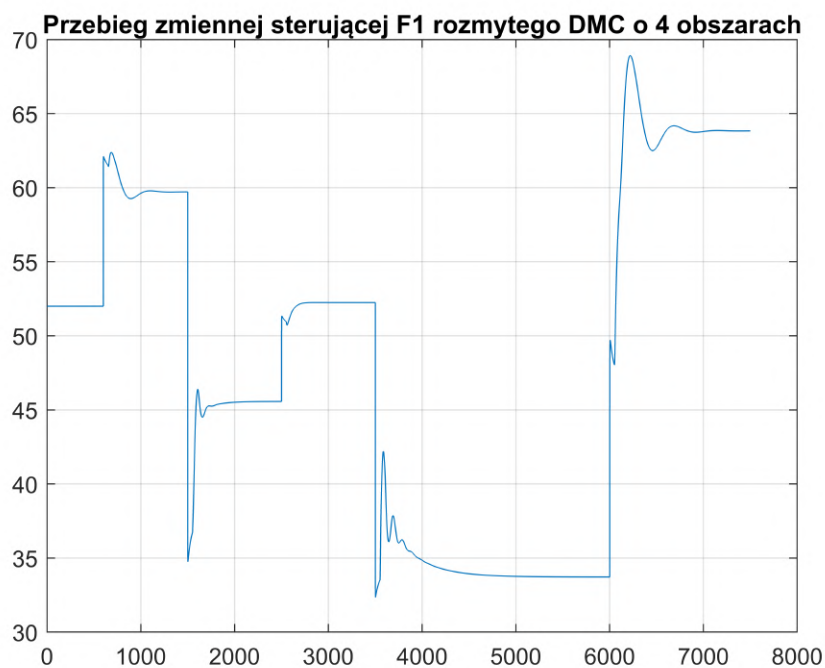


Rys. 2.17: Funkcje podziału dla DMC rozmytego o 4 obszarach.



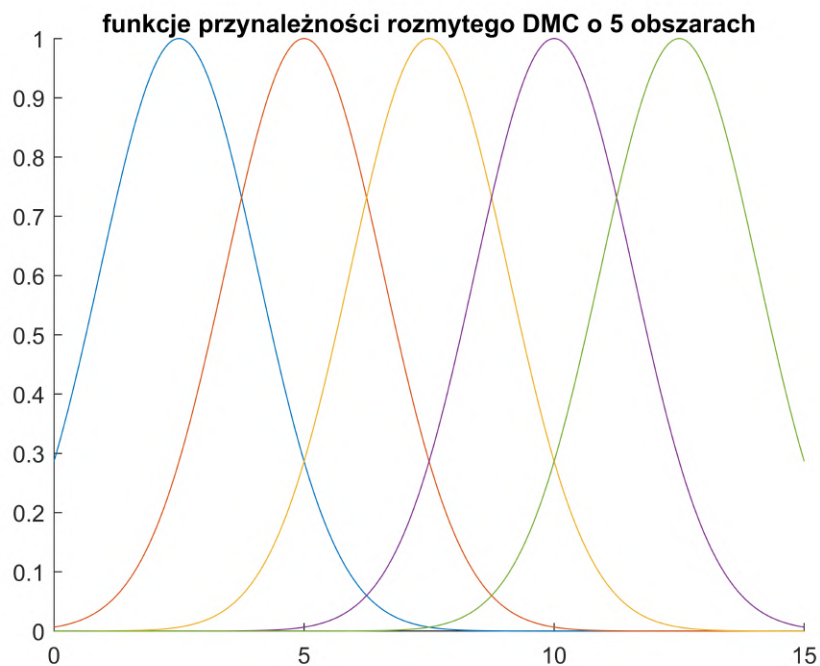
Rys. 2.18: odpowiedzi skokowe DMC rozmytego o 4 obszarach.



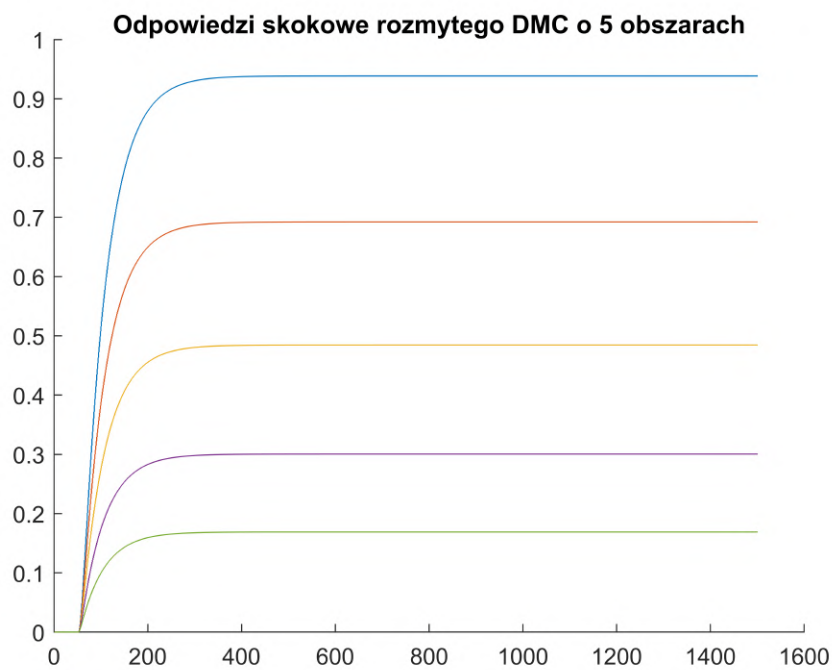
Rys. 2.19: Przebieg zmiennej wyjściowej  $h_2$  DMC rozmytego o 4 obszarach.

Rys. 2.20: Przebieg sterowania dla DMC rozmytego o 4 obszarach.

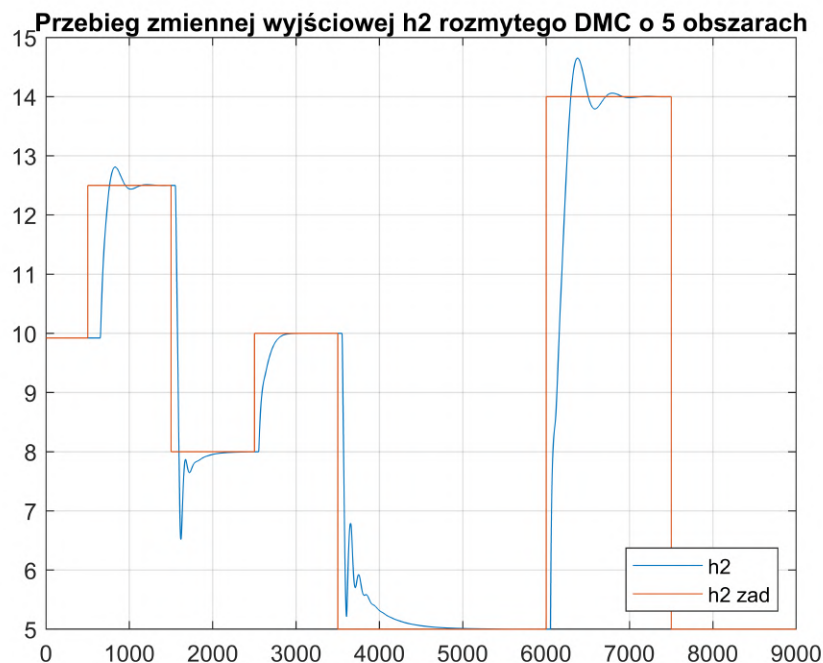
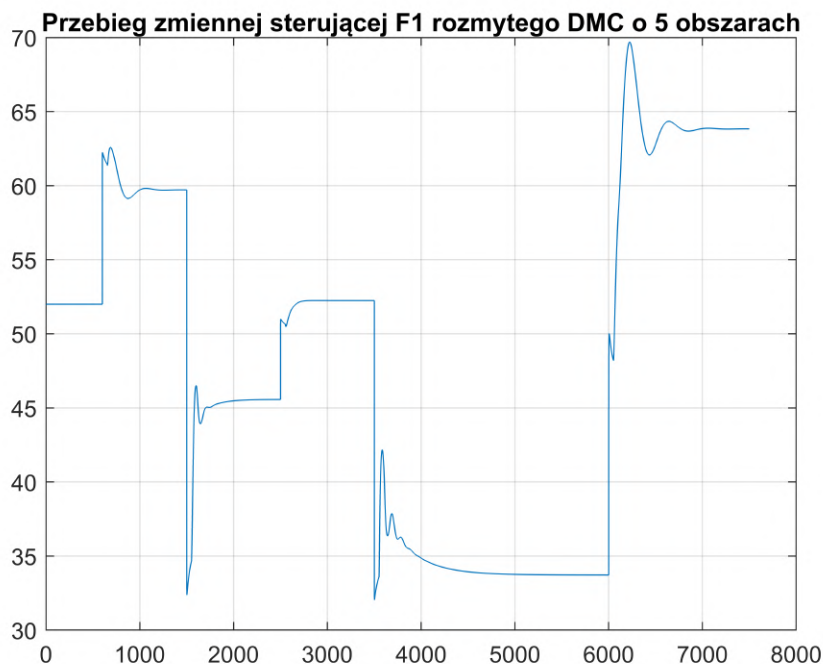
Dla 5 obszarów:



Rys. 2.21: Funkcje podziału dla DMC rozmytego o 5 obszarach.



Rys. 2.22: odpowiedzi skokowe DMC rozmytego o 5 obszarach.

Rys. 2.23: Przebieg zmiennej wyjściowej  $h_2$  DMC rozmytego o 5 obszarach.

Rys. 2.24: Przebieg sterowania dla DMC rozmytego o 5 obszarach.

Najlepszą jakość regulacji otrzymujemy dla regulatora rozmytego DMC o 3 obszarach lokalnych. Ma on mniejszy błąd regulacji od tego z 2 i 4 obszarami rozmytymi, wynoszący  $1.2622e+04$ , a także cechuje się najlepszą jakością regulacji.

### 2.3. Nastrojony regulator DMC analityczny rozmyty

Do regulacji procesu wykorzystaliśmy zaimplementowany regulator DMC analityczny rozmyty o parametrach:

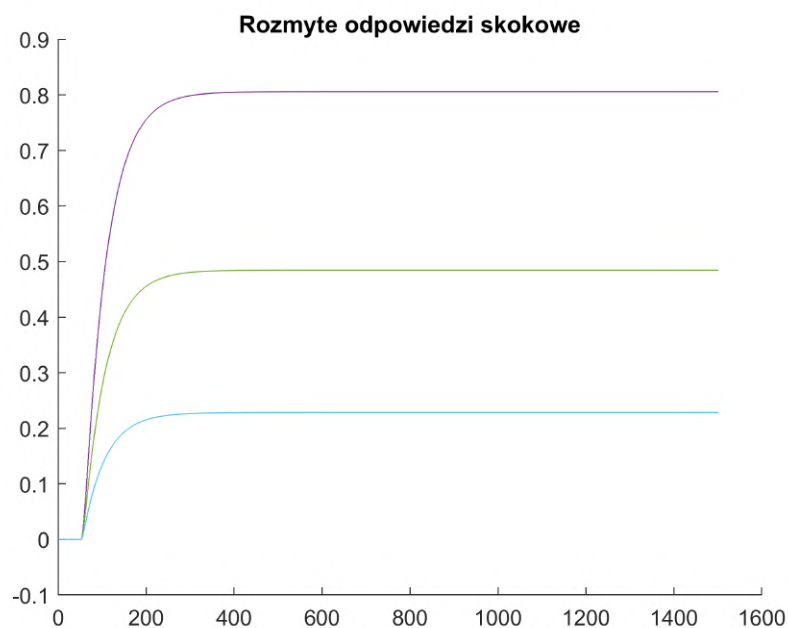
$N_1 = 300, Nu_1 = 1, \lambda_1 = 0.1,$

$N_2 = 300, Nu_2 = 1, \lambda_2 = 1,$

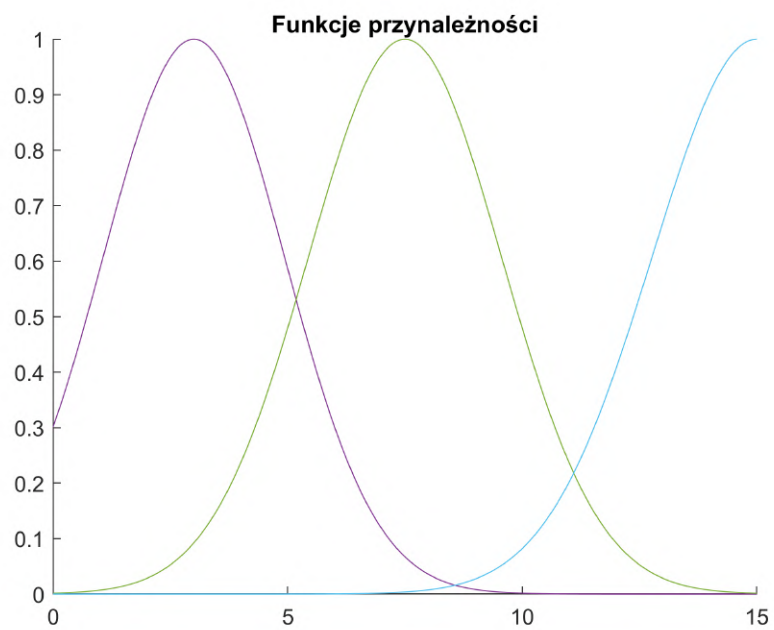
$N_3 = 500, Nu_3 = 1, \lambda_3 = 0.5,$

$D = 600$ , 3 obszary przynależności, funkcje Gaussowskie.

Implementacja znajduje się w pliku 'DMC\_ana\_rozmyty.m'. Pobiera on funkcje przynależności za pomocą funkcji 'funkcje\_podziału.m' i odpowiedzi skokowe za pomocą 'pobranie\_modelu\_rozmyte'. Odpowiedzi skokowe i funkcje przynależności:

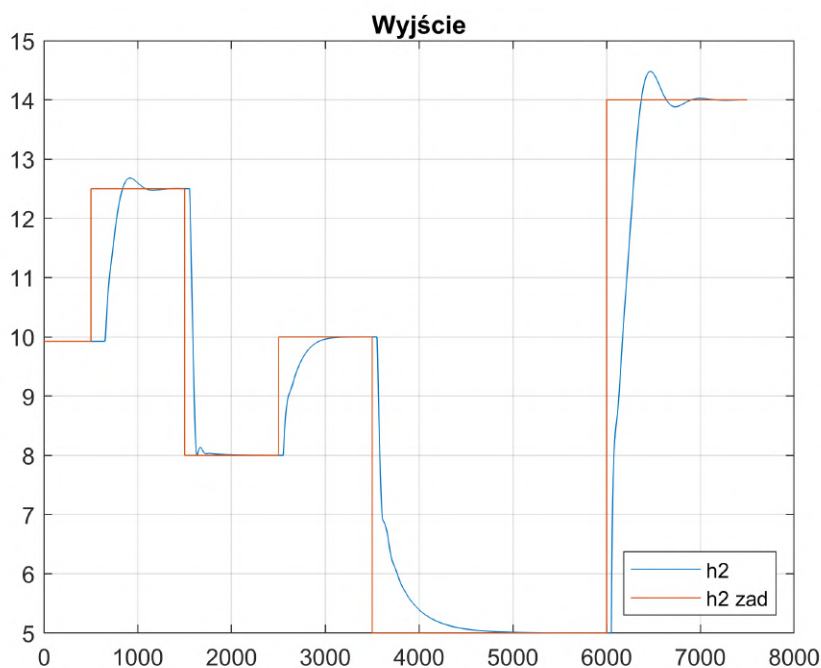


Rys. 2.25: Lokalne odpowiedzi skokowe dla regulatora DMC analitycznego rozmytego.

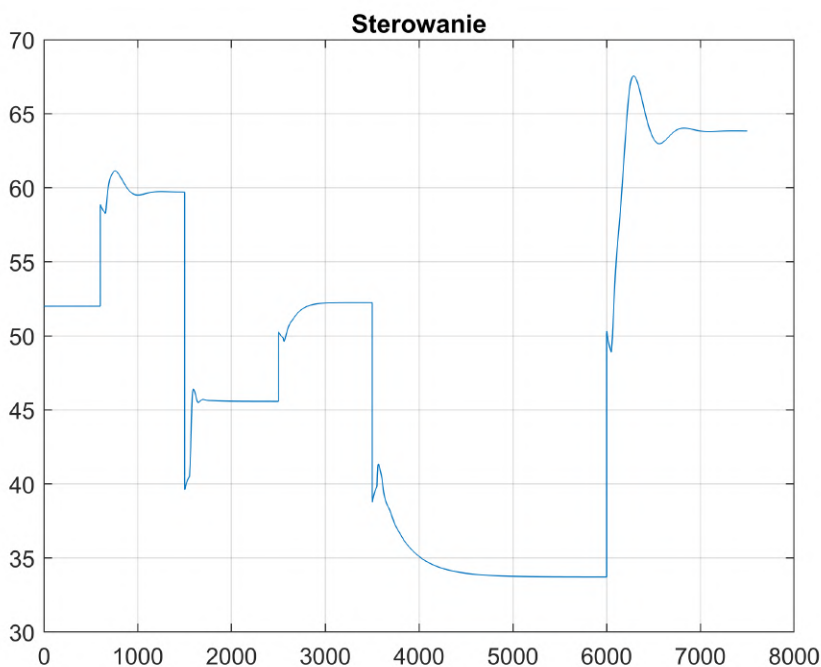


Rys. 2.26: Funkcje przynależności dla regulatora DMC analitycznego rozmytego.

Przebiegi sterowania i wyjścia dla kilku skoków wartości zadanej:



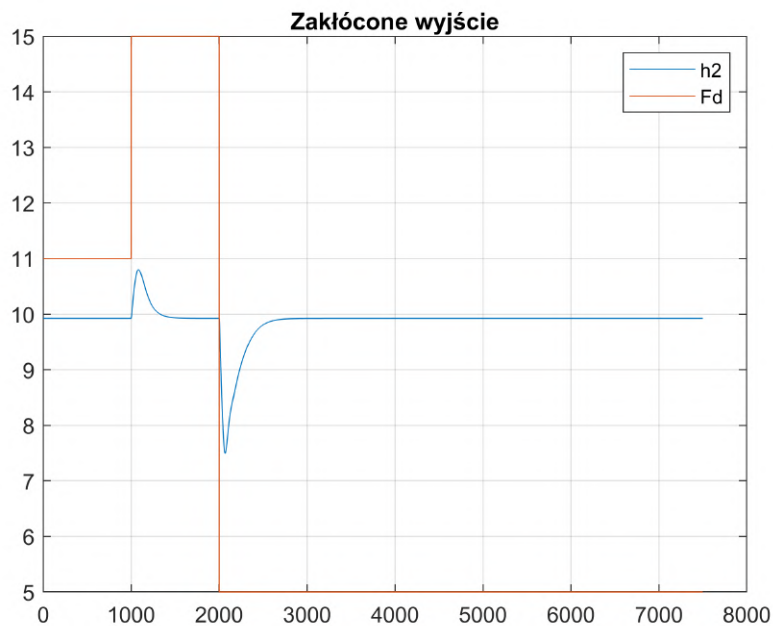
Rys. 2.27: Przebieg zmiennej wyjściowej  $h_2$  dla regulatora DMC analitycznego rozmytego.



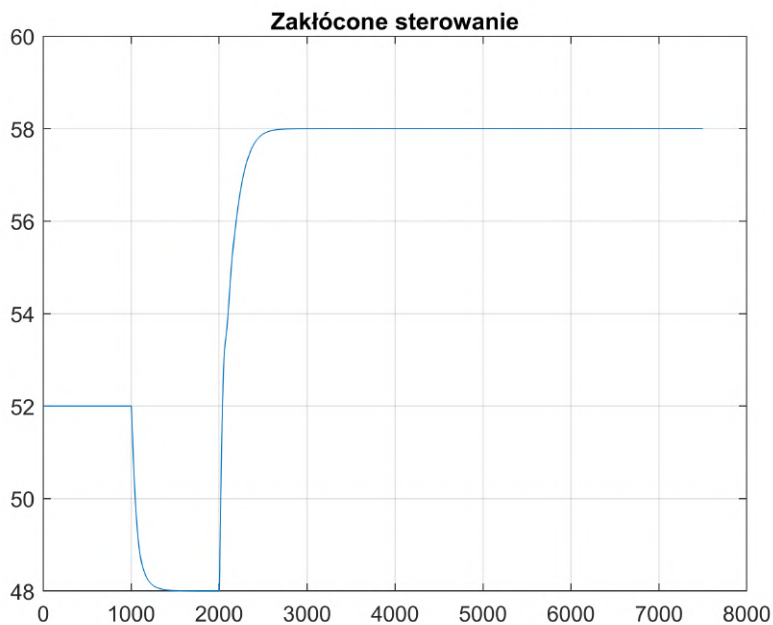
Rys. 2.28: Przebieg zmiennej sterującej  $F_1$  dla regulatora DMC analitycznego rozmytego.

DMC rozmyty ma lepsze przebiegi od zwykłej wersji analitycznej (rys. 2.7). Udało się pozbyć oscylacji w okolicy  $y_{zad} = 4$  i uchybu w okolicy  $y_{zad} = 8$ . Dokładniejsze strojenie obszarów powinno zaszkutkować lepszymi przebiegami.

Wpływ zakłóceń na działanie regulatora DMC analitycznego rozmytego:



Rys. 2.29: Przebieg zmiennej wyjściowej  $h_2$  dla regulatora DMC analitycznego rozmytego przy skokowych zakłóceniach.



Rys. 2.30: Przebieg zmiennej sterującej  $F_1$  dla regulatora DMC analitycznego rozmytego przy skokowych zakłóceniach.

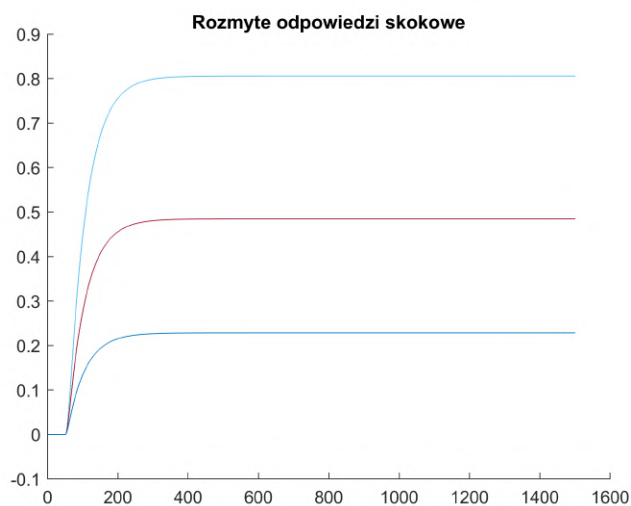
W pliku 'Zadanie2.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.

## 3. Zadanie 3

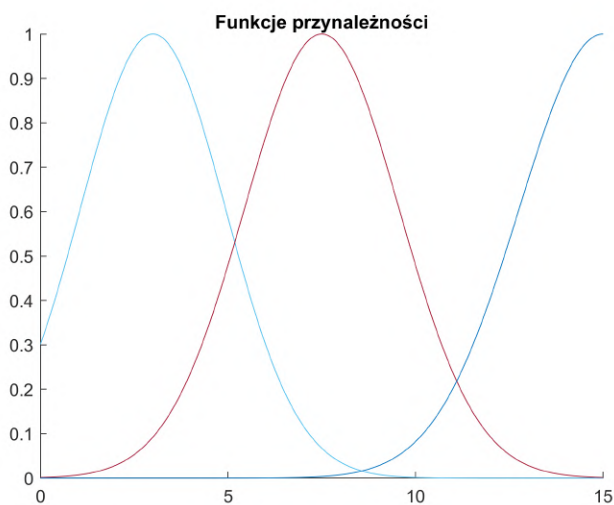
### 3.1. Regulator SL

Do regulacji procesu wykorzystaliśmy zaimplementowany regulator SL o parametrach:  $N = 500$ ,  $N_u = 1$ ,  $\lambda = 0.1$ ,  $D = 600$ , 3 obszary przynależności.

Implementacja znajduje się w pliku 'SL.m'. Pobiera on funkcje przynależności za pomocą funkcji 'funkcje\_podziału.m' i odpowiedzi skokowe za pomocą 'pobranie\_modelu\_rozmyte'.  
Odpowiedzi skokowe i funkcje przynależności:



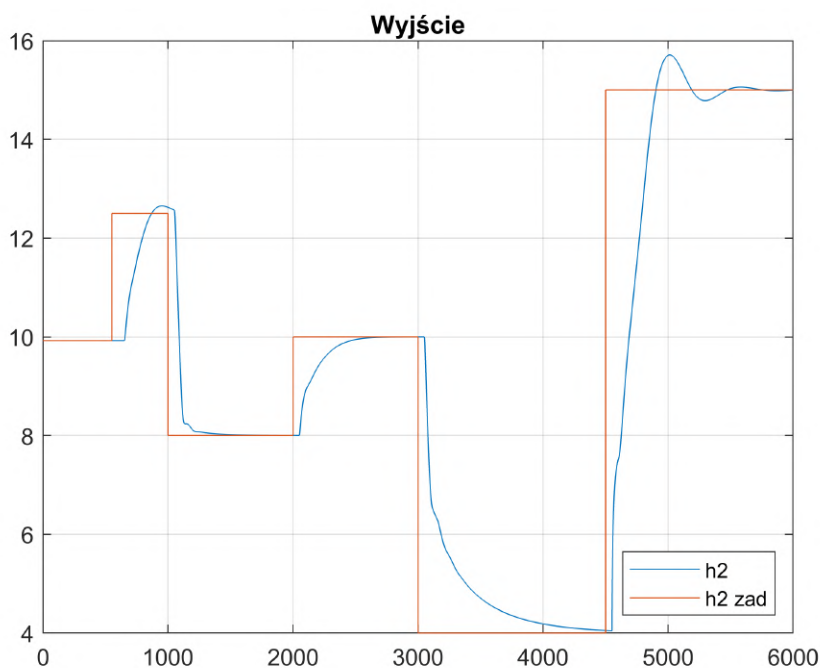
Rys. 3.1: Lokalne odpowiedzi skokowe dla regulatora SL.



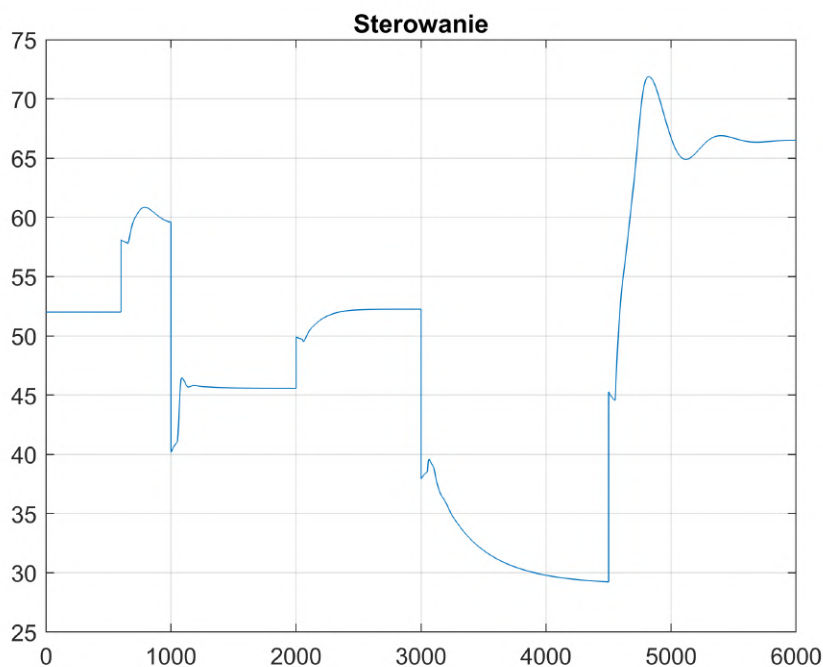
Rys. 3.2: Funkcje przynależności dla regulatora SL.



Przebiegi sterowania i wyjścia dla kilku skoków wartości zadanej:

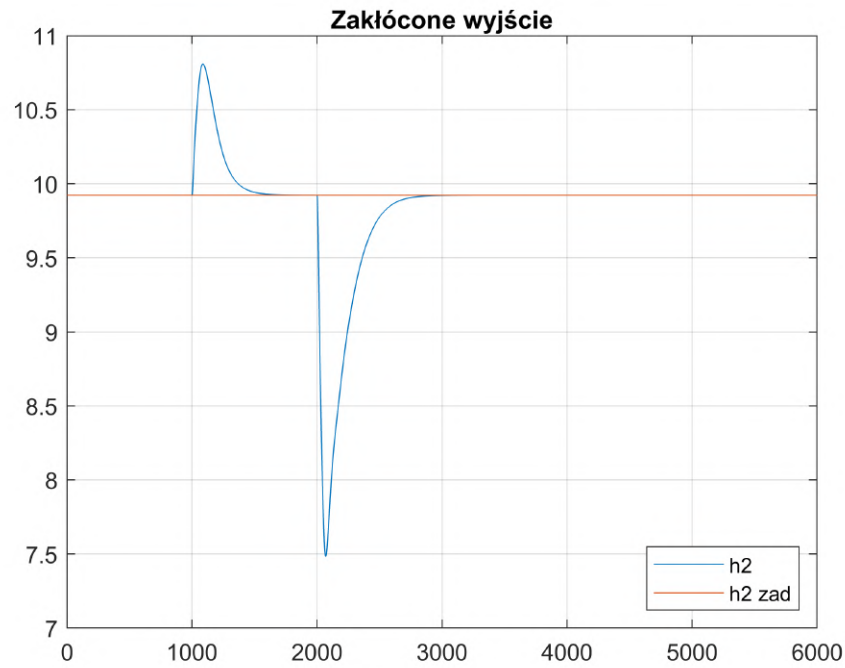


Rys. 3.3: Przebieg zmiennej wyjściowej  $h_2$  dla regulatora SL.

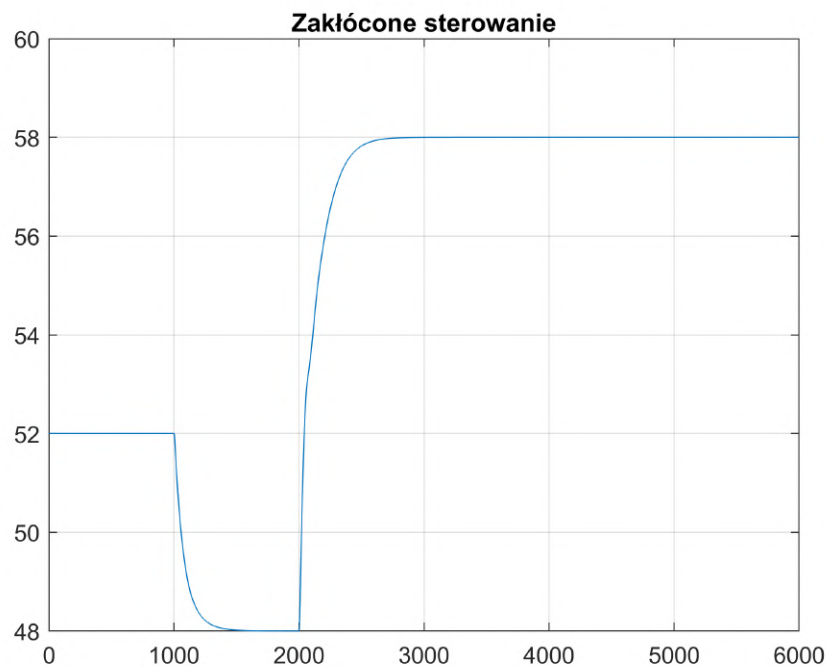


Rys. 3.4: Przebieg zmiennej sterującej  $F_1$  dla regulatora SL.

Regulator SL generalnie działa dobrze, w okolicy  $y_{\text{zad}} = 4$  brak oscylacji, ale dokładniejszy dobór funkcji przynależności mógłby poprawić szybkość dojścia do wartości zadanej. Wpływ zakłóceń na działanie regulatora SL:



Rys. 3.5: Przebieg zmiennej wyjściowej  $h_2$  dla regulatora SL przy skokowych zakłóceniach.

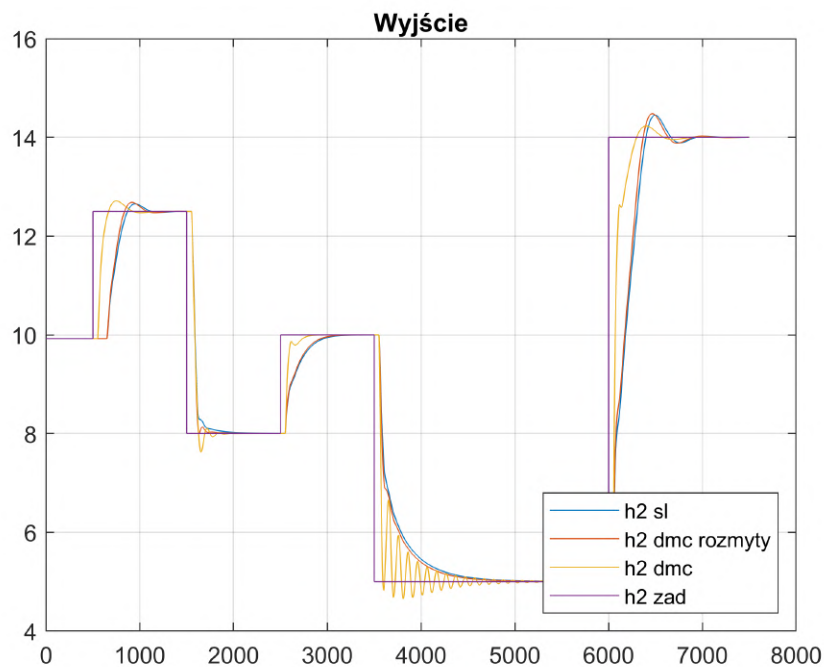


Rys. 3.6: Przebieg zmiennej sterującej  $F_1$  dla regulatora SL przy skokowych zakłóceniach.

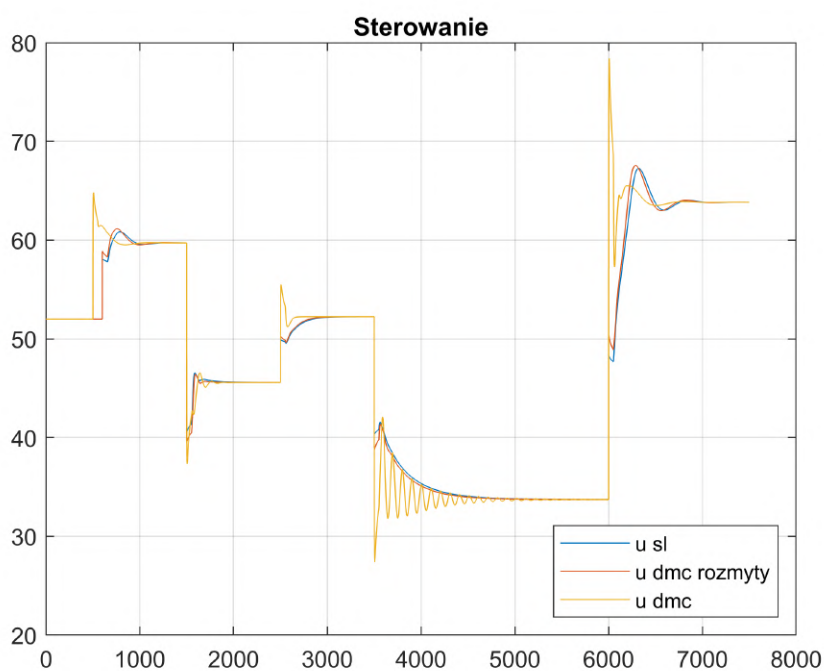
W pliku 'Zadanie3.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.

### 3.2. Porównanie działania

Przebiegi sterowania i wyjścia dla kilku skoków wartości zadanej:

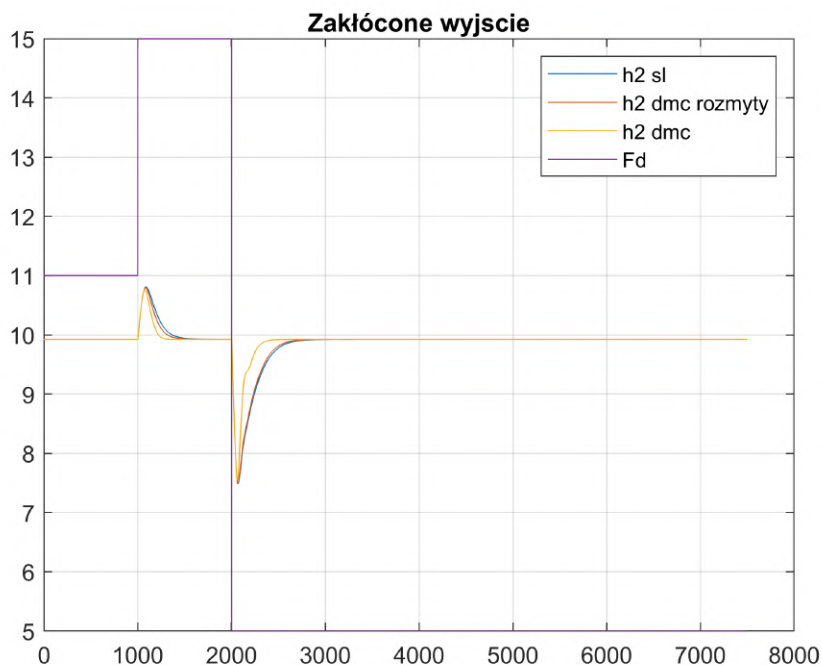


Rys. 3.7: Przebieg zmiennej wyjściowej  $h_2$  dla regulatorów.

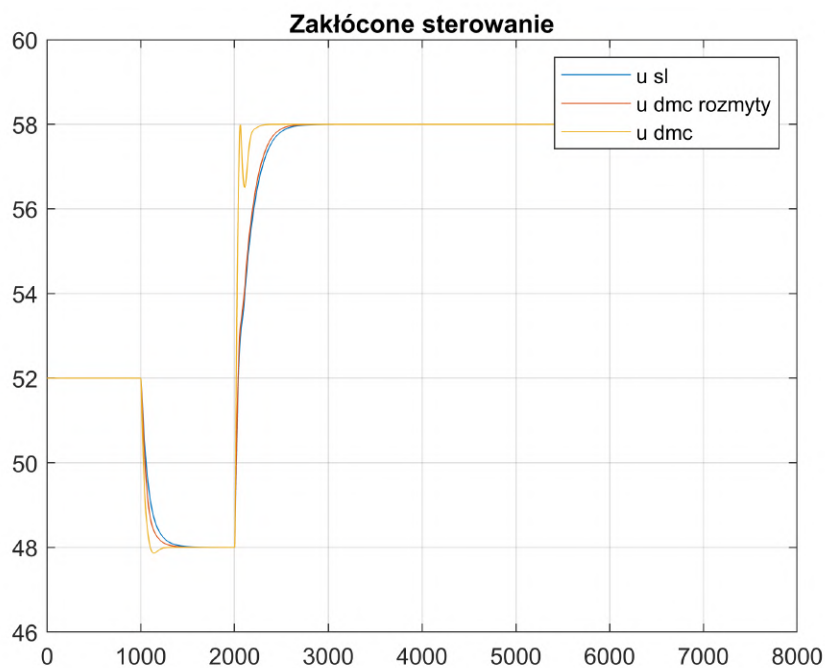


Rys. 3.8: Przebieg zmiennej sterującej  $F_1$  dla regulatorów.

Można zauważyć, że algorytmy rozmyte radzą sobie lepiej. Przebiegi wyjścia są łagodniejsze, nie mają oscylacji jak w klasycznej wersji dmc. Na przebiegach zmian wejścia pod wpływem zakłóceń również widać takie efekty. Wpływ zakłóceń na działanie regulatorów:



Rys. 3.9: Przebieg zmiennej wyjściowej  $h_2$  dla regulatorów przy skokowych zakłóceniach.



Rys. 3.10: Przebieg zmiennej sterującej  $F_1$  dla regulatorów przy skokowych zakłóceniach.

---

W pliku 'Zadanie3.m' znajduje się wywołanie odpowiednich funkcji w celu otrzymania powyższych przebiegów.