

**Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska**

Praca magisterska

**Zastosowanie modeli statyki typu Takagi-Sugeno z
następnikami hiperbolicznymi w algorytmach
regulacji predykcyjnej**

Autor:
inż. Wojciech Rogalski

Promotor:
dr hab. inż. Piotr Marusak

Warszawa, 2025

Spis treści

| | |
|--|----|
| 1. Streszczenie | 2 |
| 2. Wstęp | 3 |
| 3. Przegląd literatury | 4 |
| 3.1. Algorytmy regulacji predykcyjnej | 5 |
| 3.2. Dynamic Matrix Control | 7 |
| 3.2.1. Algorytm DMC w wersji analitycznej | 8 |
| 3.2.2. Algorytm DMC w wersji numerycznej | 9 |
| 3.2.3. Inne warianty DMC | 10 |
| 3.3. Regulacja rozmyta | 11 |
| 3.3.1. Modele Takagi-Sugeno | 13 |
| 3.3.2. Podejście Parallel Distributed Compensation | 14 |
| 3.4. Modele Hammersteina i Wienera | 15 |
| 4. Zbiorniki w kaskadzie | 16 |
| 4.1. Identyfikacja | 18 |
| 4.1.1. Charakterystyka statyczna | 18 |
| 4.1.2. Porównanie modelu liniowego i nieliniowego | 19 |
| 4.2. Model Hammersteina | 21 |
| 4.2.1. Nieliniowy blok statyczny | 21 |
| 4.2.2. Następniki liniowe | 21 |
| 4.2.3. Następniki nieliniowe | 23 |
| 4.2.4. Porównanie | 25 |
| 4.3. Model Wienera | 31 |
| 4.3.1. Następniki | 31 |
| 4.3.2. Porównanie | 33 |
| 4.4. Regulacja predykcyjna | 39 |
| 4.4.1. Regulator DMC | 39 |
| 4.4.2. Porównanie | 40 |
| 4.5. Podsumowanie | 46 |
| Bibliografia | 47 |
| Spis rysunków | 49 |
| Spis tabel | 50 |

1. Streszczenie

Tu będzie streszczenie pracy

2. Wstęp

Tu będzie wstęp do całej pracy

3. Przegląd literatury

Przedmiotem rozważań niniejszej pracy było wykorzystanie hiperbolicznych następników w modelach statyki typu Takagi - Sugeno w algorytmach regulacji predykcyjnej. Temat ten zatem porusza kilka istotnych zagadnień: regulacja predykcyjna, modelowanie rozmyte typu Takagi - Sugeno, modele Hammersteina i Wienera.

Początki regulacji predykcyjnej datuje się na lata 70 XX. wieku. Wtedy też rozwój techniki mikroprocesorowej umożliwił wdrażanie - bardziej kosztownych obliczeniowo niż bardzo dobrze znany regulator PID - algorytmów regulacji predykcyjnej zaliczonych wówczas do grupy tzw. zaawansowanych technik regulacji (*advanced control techniques*) [9]. Na początku popularnością cieszyły się głównie w branży chemicznej i petrochemicznej, ale jak pokazują czasy współczesne, zyskały uznanie również w takich branżach jak: robotyka, motoryzacja czy energetyka. Zawdzięczają to zaletom jakie oferują [5, 6, 16]:

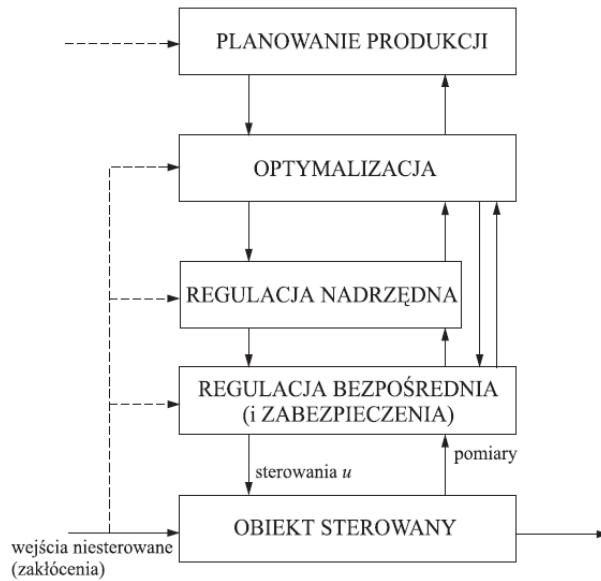
- możliwość uwzględnienia ograniczeń sygnałów zarówno wejściowych, jak i wyjściowych
- dobra jakość regulacji także w przypadku obiektów o trudnej dynamice - duże opóźnienia
- efektywne regulowanie procesu wielowymiarowego
- możliwość zaprojektowania nieliniowego algorytmu regulacji predykcyjnej
- możliwość uwzględnienia wpływu zakłóceń - jeśli dostępny jest ich mierzalny model
- główna zasada regulacji predykcyjnej jest stosunkowo prosta i uniwersalna

Omawiając kolejne zagadnienie nie sposób nie wspomnieć o regulacji nieliniowej, która pod koniec lat 70. ubiegłego stulecia wyznaczyła kierunek rozwoju przemysłowych układów regulacji i sterowania. Istotą regulacji nieliniowej bowiem jest projektowanie regulatora od razu dla obiektu modelowanego w szerokim zakresie zmian wielkości wejściowych i wyjściowych, a nie jak w przypadku regulatorów liniowych - tylko w otoczeniu wybranego punktu pracy. Takie podejście sprawia, że regulator jest odporny, zatem działa stabilnie przy zmianach parametrów obiektu równocześnie zapewniając odpowiednią jakość regulacji. Szybko okazało się, że podejście wykorzystujące zbiory rozmyte (*fuzzy sets*) oraz logikę rozmytą (*fuzzy logic*) mogą być solidnym i skutecznym narzędziem szczególnie tam, gdzie opis analityczny obiektu nie jest w pełni dostępny. Na szczególne wyróżnienie zasługują modele o strukturze Takagi-Sugeno, które bardzo dobrze nadają się do opisu silnie nieliniowych obiektów regulacji. Modelowanie rozmyte, zaraz obok modelowania z wykorzystaniem sieci neuronowych było podejściem najintensywniej rozwijanym i praktycznie stosowanym od lat 90. ubiegłego wieku [5, 16].

Ostatnim kluczowym zagadnieniem poruszonym w pracy są modele Hammersteina i Wienera. Opracowane w pierwszej połowie XX wieku lecz dynamicznie rozwijane i praktycznie stosowane dopiero w drugiej połowie, co było związane z postępowaniem w technologii systemów i modelowaniu nieliniowym. Ich istotę docenił autor w [2] wskazując, że owe modele są efektywnym podejściem do modelowania systemów, które mają liniową dynamikę, ale wprowadzają nieliniowości statyczne na wejściu (model Hammersteina) lub wyjściu (model Wienera).

3.1. Algorytmy regulacji predykcyjnej

Do lat 70. ubiegłego wieku warstwa regulacji bezpośredniej w warstwowej strukturze sterowania (Rys. 3.1) zapewniała przede wszystkim niezawodność [16]. Dominowały wówczas algorytmy PID, natomiast rozwój optymalizacji postawił nowe wymagania przed układami regulacji [5]. Okazało się bowiem, że skuteczna regulacja podczas optymalizacji punktu pracy przynosi duże korzyści ekonomiczne. To można było osiągnąć dzięki wprowadzeniu nowej grupy algorytmów - algorytmów regulacji predykcyjnej z przesuwym horyzontem, zwanym też zasadą sterowania repetycyjnego [5]. Pozwoliły po raz pierwszy rozwiązać problem uwzględnienia ograniczeń sygnałów sterujących.



Rys. 3.1: Warstwowa struktura sterowania [17].

W każdej chwili próbkowania, dysponując dynamicznym modelem obiektu, pomiarami sygnałów procesowych oraz znaną lub założoną trajektorią wyjść zadanych wyznaczana jest sekwencja przyszłych wartości sygnału sterującego przez optymalizację funkcji celu zdefiniowanej na horyzoncie predykcji [10].

Najczęściej spotykana postać funkcji kryterialnej wygląda następująco:

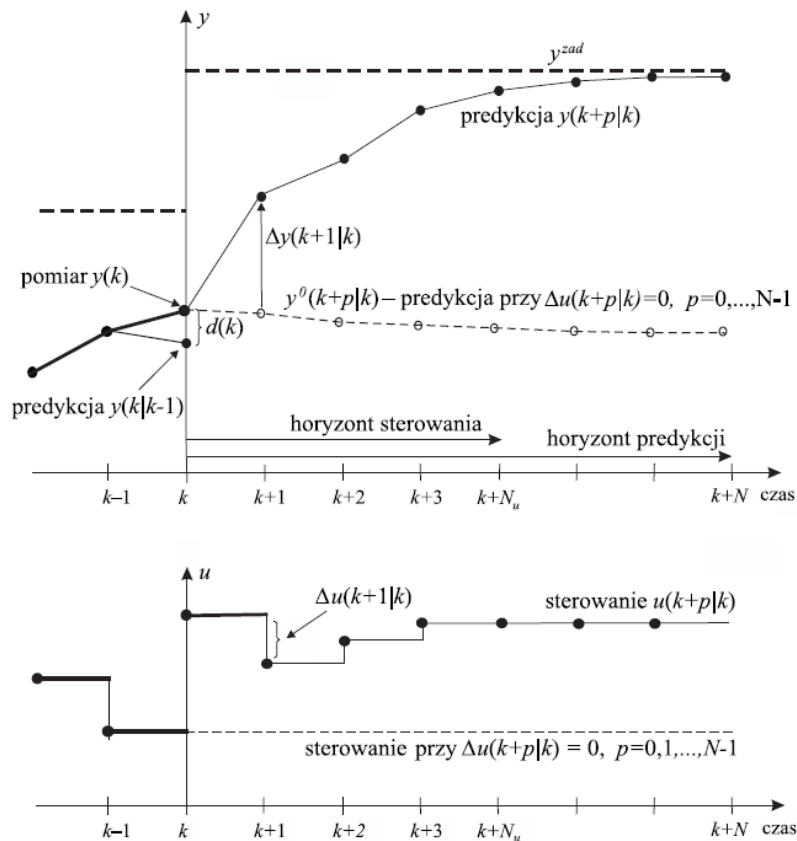
$$J(k) = \min_{\Delta U(k)} \left\{ \sum_{p=1}^N (y^{zad}(k+p|k) - \hat{y}(k+p|k))^2 + \lambda \sum_{p=0}^{N_u} (\Delta u(k+p|k))^2 \right\} \quad (3.1)$$

przy ograniczeniach:

$$\begin{aligned} u^{min} &\leq u(k+p|k) \leq u^{max} \\ -\Delta u^{max} &\leq \Delta u(k+p|k) \leq \Delta u^{max} \\ y^{min} &\leq \hat{y}(k+p|k) \leq y^{max} \end{aligned} \quad (3.2)$$

Wskaźnik jakości dostarcza informacji jakie wyniki powinien generować algorytm: uchyby regulacji muszą być jak najmniejsze, natomiast przyrosty sterowania nie powinny przyjmować dużych wartości, a ponadto korygowane są tzw. parametrem kary. Wyeliminowanie tego składnika ($\lambda = 0$) generuje przebiegi sygnału sterującego o dużych amplitudach i przyrostach, często niemożliwych do fizycznej realizacji.

Ograniczenia nałożone na sygnał sterujący związane są z uwarunkowaniami technologicznymi urządzeń wykonawczych, natomiast ograniczenia na sygnały wyjściowe służą spełnieniu norm technologicznych, ekonomicznych, a także zyskujących na coraz większym znaczeniu ekologicznych. Ogólna zasada regulacji predykcyjnej została zaprezentowana na Rys. 3.2



Rys. 3.2: Zasada działania regulacji predykcyjnej [18].

Przedstawione przebiegi zawierają kilka istotnych aspektów, które warto opatrzyć komentarzem:

- y^{zad} - trajektoria sygnału zadanego, przyjmowana jako wartość stała na horyzoncie predykcyjnym¹
- $\Delta y(k+p|k)$ - prognozowana trajektoria odpowiedzi wymuszonej zależna tylko od zmiennych decyzyjnych - przyszłych przyrostów sterowań
- $y^0(k+p|k)$ - przewidywana odpowiedź swobodna, czyli wartości odpowiadające sytuacji, w której na całym horyzoncie predykcyjnym N utrzymywana byłaby wartość sterowania z chwili poprzedniej $u(k-1)$
- $\Delta u(k+p|k)$ - kolejne przyrosty sterowania wyznaczane na horyzoncie sterowania N_u

Na tej podstawie oraz korzystając z zasady superpozycji można zapisać:

$$y(k+p|k) = y^0(k+p|k) + \Delta y(k+p|k) \quad p = 1 \dots N \quad (3.3)$$

¹ Najczęściej zakładana jest stała trajektoria zadana, natomiast w niektórych dziedzinach, np. robotyka, dopuszcza się uwzględnienie zmian trajektorii zadanej w celu uprzedniego reagowania na jej zmiany.

Zasada regulacji predykcyjnej jest dość uniwersalna co pozwoliło na wykształcenie kilku dominujących algorytmów, takich jak:

- DMC (*Dynamic Matrix Control*) - algorytm predykcyjny wykorzystujący model liniowy w postaci dyskretnych odpowiedzi skokowych
- GPC (*Generalized Predictive Control*) - algorytm predykcyjny wykorzystujący model liniowy w postaci dyskretnych równań różnicowych
- MPCs (*Model Predictive Control with State-space model*) - algorytm predykcyjny wykorzystujący w swoim opisie równania stanu
- MPHC (*Model Predictive Heuristic Control*) - algorytm predykcyjny wykorzystujący model liniowy w postaci odpowiedzi impulsowej

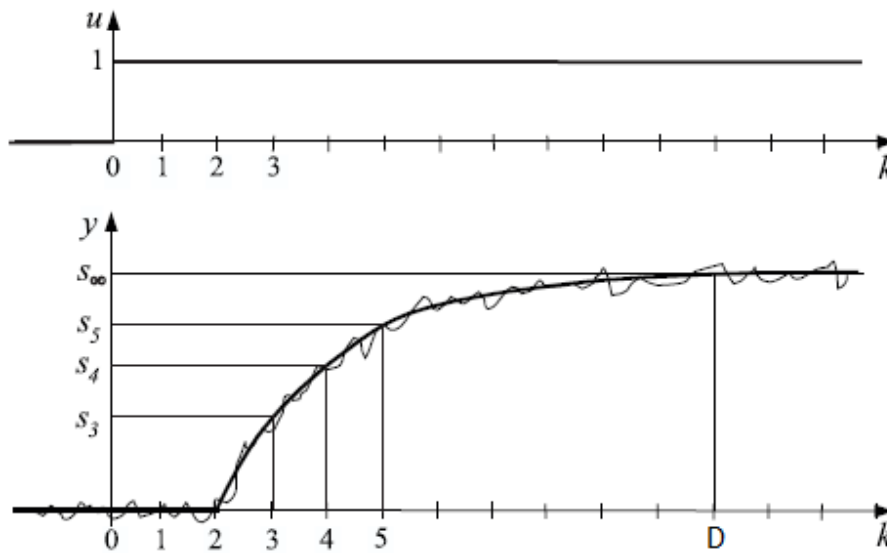
Ze względu na praktyczną naturę algorytmu DMC do dalszych rozważań postanowiono przyjąć właśnie ten algorytm.

3.2. Dynamic Matrix Control

Algorytm DMC pierwszy raz został zaimplementowany przez grupę Shell Oil w latach 70 XXw. Na tamten czas dało im to ogromną przewagę w branży petrochemicznej, a sam algorytm dzięki bezpośredniemu praktycznemu zastosowaniu stał się bardziej popularny. Algorytm ten zasadę swojego działania opiera na odpowiedzi skokowej, co wg autora [16] stanowi jeden z najskuteczniejszych form identyfikacji obiektu.

Na Rys. 3.3 przedstawiono przykład odpowiedzi skokowej na wymuszenie jednostkowe, zaprezentowane na horyzoncie dynamiki, czyli czas po którym wartość odpowiedzi skokowej można uznać za ustaloną, tj.

$$s_k = s_{k+1} = s_{k+2} = \dots = s_D = s_\infty \quad (3.4)$$



Rys. 3.3: Odpowiedź skokowa [19].

Korzystając z zasady superpozycji, dla każdej chwili dyskretniej $k + p$, można zatem zapisać:

$$y^{mod}(k + p|k) = y^{mod}(0) + \sum_{j=1}^{k+p} s(j) \Delta u(k + p - j) \quad (3.5)$$

Z kolei wartość przewidywanej trajektorii wyjść można zapisać jako suma wartości modelu i zakłócenia spowodowanego do wyjścia:

$$\hat{y}(k+p|k) = y^{mod}(k+p|k) + d(k+p|k) \quad (3.6)$$

gdzie zakłócenie $d(k)$ jest równe różnicy między pomiarem wyjścia w chwili k , a wartością obliczoną z modelu w chwili $k-1$:

$$d(k) = y(k) - y^{mod}(k|k-1) \quad (3.7)$$

natomiast ogólnie przejętą praktyką jest przyjmowanie modelu zakłóceń typu DMC, tj.

$$d(k+1|k) = d(k+2|k) = \dots = d(k+N|k) = d(k) \quad (3.8)$$

Takie założenia pozwalają na sformułowanie wyjściowej postaci trajektorii prognozowanej w algorytmie DMC:

$$\begin{cases} \hat{y}(k+p|k) = y^0(k+p|k) + \Delta y(k+p|k) \\ y^0(k+p|k) = y(k) + \sum_{j=1}^k (s(j+p) - s(j)) \Delta u(k-j) \\ \Delta y(k+p|k) = \sum_{j=1}^p s(j) \Delta u(k+p-j) \end{cases} \quad (3.9)$$

Zatem w ogólności można zapisać trajektoria prognozowana jest sumą odpowiedzi swobodnej i wymuszonej. Algorytm DMC występuje w wersji analitycznej bez oraz z rzutowaniem na ograniczenia oraz w wersji numerycznej, uwzględniającej te ograniczenia.

3.2.1. Algorytm DMC w wersji analitycznej

Do sformułowania prawa regulacji w wersji analitycznej, w postaci wektorowej niezbędne jest zdefiniowanie wektorów postaci:

$$Y^{zad} = \begin{bmatrix} y^{zad}(k+1|k) \\ \vdots \\ y^{zad}(k+N|k) \end{bmatrix}_{N \times 1}, \quad \hat{Y} = \begin{bmatrix} \hat{y}(k+1|k) \\ \vdots \\ \hat{y}(k+N|k) \end{bmatrix}_{N \times 1}, \quad \Delta U(k) = \begin{bmatrix} \Delta u(k|k) \\ \vdots \\ \Delta u(k+N_u-1|k) \end{bmatrix}_{N_u \times 1} \quad (3.10)$$

Oraz macierzy wagowych:

$$\Psi = \begin{bmatrix} \psi_1 & & \\ & \ddots & \\ & & \psi_N \end{bmatrix}_{N \times N}, \quad \Lambda = \begin{bmatrix} \lambda_0 & & \\ & \ddots & \\ & & \lambda_{N_u-1} \end{bmatrix}_{N_u \times N_u} \quad (3.11)$$

Tak zdefiniowane wektory pozwalają zapisać funkcję celu w postaci:

$$J(k) = \min_{\Delta U(k)} \{ \|Y^{zad} - \hat{Y}(k)\|_{\Psi}^2 + \|\Delta U(k)\|_{\Lambda}^2 \} \quad (3.12)$$

Odwołując się do równania 3.9 prognozowaną trajektorię wyjść w wersji macierzowo-wektorowej można zapisać w postaci:

$$\begin{cases} \hat{Y} = Y^0(k) + \Delta Y(k) \\ Y^0(k) = Y(k) + M^P \Delta U^P(k) \\ \Delta Y(k) = M \Delta U(k) \end{cases} \quad (3.13)$$

gdzie:

$$M^P = \begin{bmatrix} s_2 - s_1 & s_3 - s_2 & \cdots & s_D - s_{D-1} \\ s_3 - s_1 & s_4 - s_2 & \cdots & s_D - s_{D-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{N+1} - s_1 & s_{N+2} - s_2 & \cdots & s_{N+D-1} - s_{D-1} \end{bmatrix}_{Nx(D-1)} \quad (3.14)$$

$$M = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ s_2 & s_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_N & s_{N-1} & \cdots & s_{N-N_u+1} \end{bmatrix}_{NxN_u}, \quad \Delta U^P(k) = \begin{bmatrix} \Delta u(k-1) \\ \vdots \\ \Delta u(k-D+1) \end{bmatrix}_{(D-1) \times 1}$$

Wektor $\Delta U^P(k)$ opisuje wartości przyrostów sterowań z poprzednich chwil, natomiast macierz M nazywana jest macierzą dynamiczną. Uwzględniając postać funkcji kryterialnej (3.12) oraz przyjmując odpowiednie założenia $\Psi \geq 0$ oraz $\Lambda \geq 0$ wektor kolejnych przyrostów sterowań dany jest wzorem:

$$\Delta U(k) = K(Y^{zad} - Y^0) \quad (3.15)$$

gdzie

$$K = (M^T \Psi M + \Lambda)^{-1} M^T \Psi \quad (3.16)$$

Zważywszy na fakt, że w każdej iteracji algorytmu wykorzystywany jest tylko pierwsza z wyznaczonych wartości przyrostów sterowań, równanie 3.15 można uprościć i zapisać w postaci:

$$\Delta u(k) = \bar{K}_1(Y^{zad} - Y^0(k)) \quad (3.17)$$

gdzie \bar{K}_1 jest pierwszym wierszem macierzy K . Równanie 3.17 opisuje analityczną postać algorytmu DMC. Kolejnym istotnym aspektem jest rzutowanie wyznaczanych wartości na ograniczenia:

$$\begin{aligned} u^{min} &\leq u(k+1|k) \leq u^{max} \\ -\Delta u^{max} &\leq \Delta u(k|k) \leq \Delta u^{max} \\ y^{min} &\leq \hat{y}(k|k) \leq y^{max} \end{aligned} \quad (3.18)$$

Wersja analityczna DMC znajduje zastosowanie w praktyce, szczególnie w systemach liniowych o niewielkiej liczbie zmiennych sterowania i ograniczeń. Jednak w bardziej złożonych przypadkach, zwłaszcza z dużymi układami wielowymiarowymi lub w systemach nieliniowych, częściej stosuje się wersję numeryczną.

3.2.2. Algorytm DMC w wersji numerycznej

Kwadratowa funkcja kryterialna - dzięki zastosowaniu modelu liniowego do predykcji - umożliwia rozwiązanie zadania minimalizacji w sposób analityczny, ale również numeryczny. Optymalizacja numeryczna ma tę przewagę, że pozwala uwzględnić ograniczenia nałożone na sygnał sterujący [5], tzn. w każdej iteracji wyznaczany jest wektor przyszłych sterowań w wyniku rozwiązania następującego zadania optymalizacji w postaci standardowej:

$$\min\{J(x) = \frac{1}{2}x^T Hx + f^T x\} \quad (3.19)$$

przy ograniczeniach:

$$\begin{aligned} x_{min} &\leq x \leq x_{max} \\ Ax &\leq b \end{aligned} \quad (3.20)$$

gdzie:

$$\begin{aligned}
 x &= \Delta U(k), \quad x_{min} = -\Delta U_{max}, \quad x_{max} = \Delta U_{max} \\
 H &= 2(M^T \Psi M + \Lambda) \\
 f &= -2M^T \Psi (Y^{zad}(k) - Y^0(k)) \\
 A &= \begin{bmatrix} -J \\ J \\ -M \\ M \end{bmatrix}, \quad b = \begin{bmatrix} -U_{min} + U(k-1) \\ U_{max} - U(k-1) \\ -Y_{min} + Y^0(k) \\ Y_{max} - Y^0(k) \end{bmatrix} \\
 J &= \mathbb{I}_{N_u \times N_u}, \quad Y_{min} = [y_{min}, \dots, y_{min}]^T, \quad Y_{max} = [y_{max}, \dots, y_{max}]^T
 \end{aligned} \tag{3.21}$$

Działanie regulatora DMC z ograniczeniami zostało przedstawione w pracy [11], w której autorzy dokonali szeregu testów sprawdzających wpływ przeprowadzanej optymalizacji w każdym kroku na jakość regulacji.

3.2.3. Inne warianty DMC

Zastosowanie algorytmów predykcyjnych w latach 70. ubiegłego stulecia pokazało jak duże korzyści - nie tylko w aspekcie jakości regulacji, ale także ekonomicznych - może przynieść ich wdrożenie do obiektu sterowania. Wówczas ten fakt spowodował dynamiczny rozwój tych algorytmów, czego efektem są m.in. algorytmy wykorzystujące nieliniowe modele obiektów:

- Algorytm DMC z sukcesywną linearyzacją (DMC-SL)
- Algorytm DMC z nieliniową optymalizacją (DMC-NO)
- Algorytm DMC z nieliniową predykcją i linearyzacją (DMC-NPL)

DMC-SL

Algorytm podstawowy, korzystający z modelu liniowego często może okazać się niewystarczający, np. gdy obiekt wykazuje silną nieliniowość. Dużą poprawę, tj. minimalizację błędów linearyzacji, można osiągnąć implementując algorytm predykcyjny z sukcesywną linearyzacją. W każdej iteracji, dokonywana jest linearyzacja modelu nieliniowego w punkcie pracy, w którym aktualnie znajduje się obiekt. Następnie na podstawie modelu liniowego wyznaczana jest odpowiedź skokowa oraz formowana jest macierz dynamiczna [6, 16]. Okazuje się, że linearyzacja modelu nie jest wymagana w każdym kroku, co zaprezentował autor [8]. Jeśli zmiany między kolejnymi wartościami predykcji nie są duże, można korzystać z uprzednio wyznaczonego modelu liniowego.

DMC-NPL

Algorytm DMC z nieliniową predykcją i linearyzacją jest zmodyfikowaną wersją algorytmu wykorzystującą sukcesywną linearyzację. W tym przypadku bowiem odpowiedź swobodna jest wyznaczana na podstawie modelu nieliniowego, natomiast linearyzacja służy wyznaczeniu odpowiedzi skokowej i sformułowania macierzy dynamicznej - podobnie jak w DMC-SL. Marginalizacja wpływu modelu liniowego na proces przynosi poprawę szczególnie w przypadkach gdy regulowany obiekt wykazuje silną nieliniowość charakteryzuje się szybkim przechodzeniem do odległych punktów równowagi, np. po wystąpieniu silnych zakłóceń czy też przy uruchamianiu lub wyłączaniu procesów [6, 16].

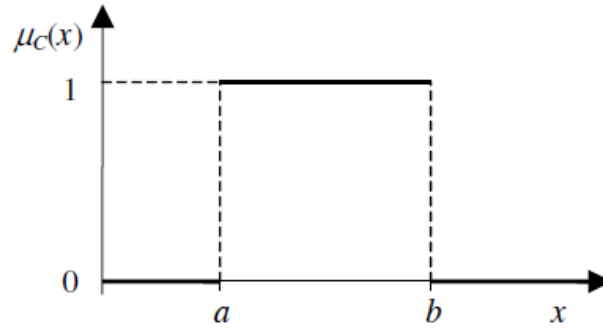
DMC-NO

Algorytm ten zaliczany jest do grupy algorytmów z nieliniową optymalizacją. Wykorzystuje pełny nieliniowy model procesu do predykcji. Prostota koncepcji i jakość regulacji na najwyższym poziomie nie są jednak w stanie zrekomensować istotnych wad tego podejścia, mianowicie złożoność obliczeniowa oraz fakt, że nie istnieją algorytmy rozwiązujące nieliniowy problem optymalizacji w możliwym do oszacowania czasie eliminują algorytm DMC-NO z praktycznego użycia [6].

3.3. Regulacja rozmyta

Tytułowy typ regulacji zrywa ze standardową procedurą podejmowania decyzji w sposób binarny, tzn. 1/0 [10]. Na Rys. 3.4 przedstawiono standardową funkcję przynależności do zbioru, którą można opisać wzorem:

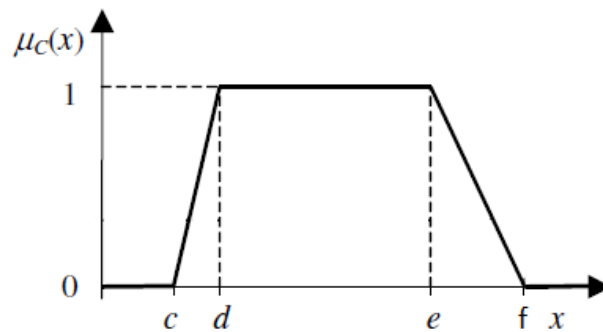
$$\mu_C = \begin{cases} 1, & x \in [a, b] \\ 0, & x \notin [a, b] \end{cases} \quad (3.22)$$



Rys. 3.4: Funkcja przynależności zbioru ostrego.

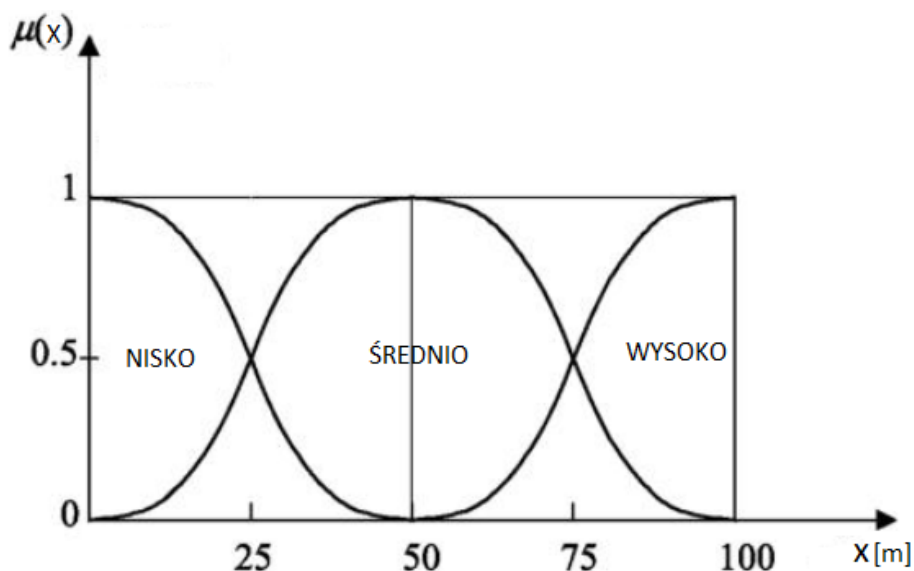
Natomiast istota wnioskowania rozmytego polega na zgromadzeniu wiedzy eksperckiej, operatora procesu, wspartej odpowiedzią skokową obiektu, które pozwolą na sformułowanie zbiorów rozmytych (Rys. 3.6) oraz bazy reguł definiującej regulator rozmyty [5, 6]. Funkcja przynależności zbioru rozmytego została zaprezentowana na Rys. 3.5 i w tym przypadku dana jest wzorem:

$$\mu_C = \begin{cases} 0, & x \leq c \vee x \geq f \\ \frac{x-c}{d-c}, & c \leq x \leq d \\ 1, & d \leq x \leq e \\ \frac{f-x}{f-e}, & e \leq x \leq f \end{cases} \quad (3.23)$$



Rys. 3.5: Funkcja przynależności zbioru rozmytego.

Zmienne z odcinków $[c, d]$ oraz $[e, f]$ przyjmują wartości z przedziału $[0, 1]$, w tym sensie granice zbioru przynależności są rozmyte. Natomiast wartość funkcji przynależności do zbioru określana jest stopniem przynależności. Należy w tym miejscu zwrócić uwagę na kształty funkcji przynależności. Funkcje trapezowe bądź trójkątne proste w swym zapisie, nie są różniczkowalne. Aspekt ten jest o tyle istotny, że zapewnia stabilność, zwiększa dokładność w systemach sterowania rozmytego, a także jest niezbędny podczas uczenia maszynowego, wykorzystującego metody gradientowe. Dlatego w wielu zastosowaniach przyjęło się korzystanie z innych kształtów funkcji przynależności, zapewniających różniczkowalność, takich jak: gaussowskie, dzwonowe, sigmoidalne.



Rys. 3.6: Przykładowe zbiory rozmyte.

Ważną kwestią w rozumieniu systemów rozmytych, jest pośrednik między zmienną numeryczną, a zmienną symboliczną - zmienna lingwistyczna. Na Rys. 3.6 przedstawiono rozmycie zmiennej "wysokość", która przyjmuje trzy wartości: "nisko", "średnio", "wysoko" i tak można zauważyć, że wartość numeryczna 75m ze stopniem przynależności 0,5 należy do zbiorów "średnio" oraz "wysoko" [15, 16, 20].

Po rozmyciu zmiennej lingwistycznej, określeniu stopni przynależności danych wartości do poszczególnych zbiorów pozostaje jeszcze zdefiniowanie bazy wiedzy, czyli tzw. zbioru reguł. Każda reguła składa się z części warunkowej, zwanej poprzednikiem oraz konsekwencji, określanej następnikiem. Ogólna struktura reguły w rozumieniu systemów rozmytych przyjmuje postać:

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } < \text{nastpnik} > \quad (3.24)$$

Poprzedniki reguł w najprostszym przypadku mogą zawierać pojedynczy warunek w innej sytuacji mogą składać się z kilku prostych warunków połączonych operacjami logicznymi (i, lub, nie).

- warunek prosty: JEŚLI x jest A TO $< \text{nastpnik} >$
- warunek złożony: JEŚLI x_1 jest A_1 lub x_2 jest A_2 i x_3 nie jest A_3 TO $< \text{nastpnik} >$

W przypadku następników reguł wyróżnić można ich trzy postaci:

1. Następnik ostry

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y = y_a \quad (3.25)$$

2. Następnik rozmyty

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y \text{ jest } Y_a \quad (3.26)$$

3. Następnik funkcyjny

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y = f(x_1, x_2, \dots, x_k) \quad (3.27)$$

Szczególnie istotne oraz wykorzystywane w praktyce są dwie ostatnie wymienione metody. Następniki w postaci zbiorów rozmytych są z powodzeniem wykorzystywane w modelach Mamdaniego [6, 15], gdzie wykorzystując wiedzę eksperta można z dużą dokładnością sterować obiektem regulacji w sposób rozmyty. Często praktyką w tym podejściu jest budowanie tablicy decyzyjnej, która w stosunkowo prosty i przejrzysty sposób redukuje bazę reguł do tabeli [6].

W kontekście niniejszej pracy skupiono się przede wszystkim na następnikach funkcyjnych opisujących modele Takagi - Sugeno, którym poświęcono następny rozdział.

3.3.1. Modele Takagi-Sugeno

Projektowanie modeli Takagi-Sugeno składa się z następujących etapów:

1. Obliczenie poziomów aktywacji reguł
2. Wyznaczenie konkluzji - obliczenie wartości następników funkcyjnych poszczególnych reguł
3. Wyznaczenie konkluzji końcowej - zsumowanie wartości następników funkcyjnych - ważone i normowane - z uwzględnieniem sił odpalenia reguł, co opisuje wzór 3.28.

$$y = \frac{\sum_{j=1}^r w^j y^j}{\sum_{j=1}^r w^j} \quad (3.28)$$

gdzie:

- r - liczba reguł
- w^j - siły odpalenia poszczególnych reguł
- y^j - wartości odpowiednich następników funkcyjnych

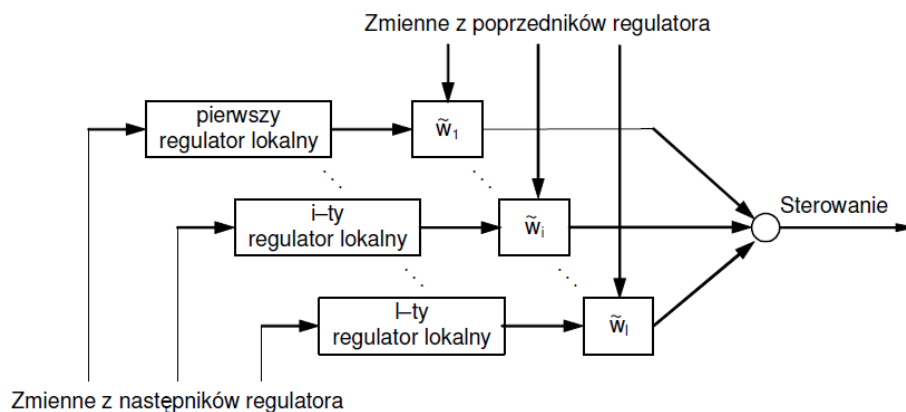
Ogólnie modele rozmyte służą do aproksymacji funkcji nieliniowych stąd w przypadku modeli Takagi-Sugeno następniki funkcyjne występują najczęściej w postaci funkcji wielomianowych pierwszego rzędu [5, 6]:

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y = a_0 + \sum_{j=1}^n a_j x_j \quad (3.29)$$

Natomiast w pracy skupiono się jakie korzyści może przynieść zastosowanie nieliniowych - hiperbolicznych - następników. Według autorów [14] pozwala to na zmniejszenie liczby zbiorów rozmytych, a tym samym reguł.

3.3.2. Podejście Parallel Distributed Combensation

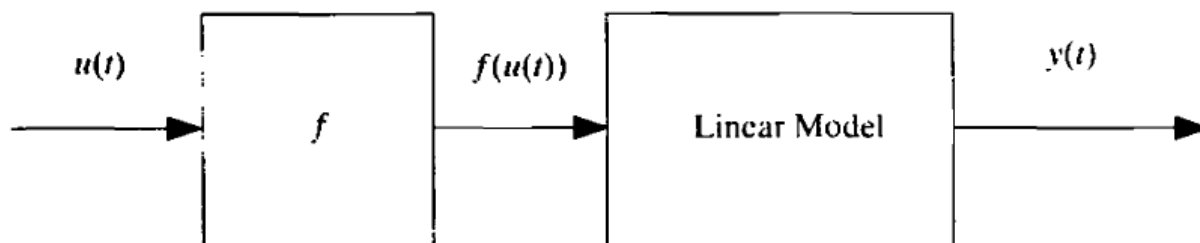
Podejście PDC (*Parallel Distributed Combensation*) zakłada dedukcję regulatora za pomocą modelu Takagi-Sugeno obiektu. Istota równoległej kompensacji rozproszonej polega na dobraniu dla każdego następnika rozmytego lokalnego regulatora liniowego. Zatem w wyniku takiego podejścia otrzymuje się tyle regulatorów z ilu modeli lokalnych, opisujących obiekt w danych obszarze, składa się system rozmyty. Na początku zakłada się taką samą postać poprzedników jak w modelu obiektu, następnie w miarę potrzeby dostraja [16, 6]. Struktura regulatora otrzymanego w wyniku zastosowania podejścia PDC została przedstawiona na Rys. 3.7.



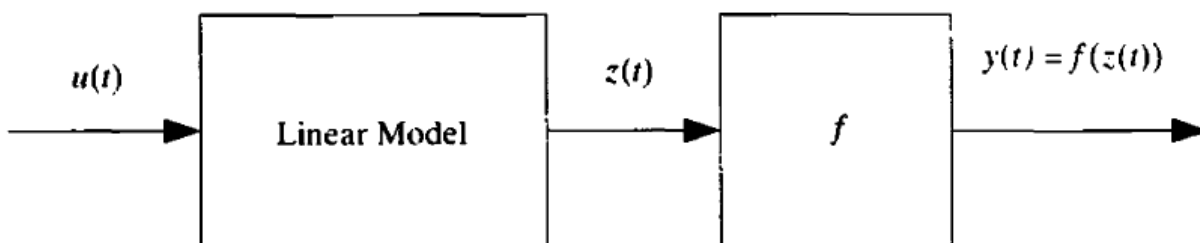
Rys. 3.7: Struktura regulatora rozmytego otrzymanego podejściem PDC [7].

3.4. Modele Hammersteina i Wienera

Istotą stosowania modeli Hammersteina i Wienera jest rozdzielenie liniowej dynamiki od zakłóceń wprowadzanych przez statyczne nieliniowości. Mogą one występować na wejściu (Rys. 3.8) bądź wyjściu (Rys. 3.9) [1].



Rys. 3.8: Model Hammersteina [3].



Rys. 3.9: Model Wienera [3].

Modele te doskonale nadają się do identyfikacji systemów nieliniowych. Często w tym celu wykorzystywane są funkcje wielomianowe, sieci neuronowe czy systemy rozmyte [4].

4. Zbiorniki w kaskadzie

W celu zapoznania się z dokładnym mechanizmem działania modeli Hammersteina i Wienera z realizacją nieliniowej statyki za pomocą modeli rozmytych typu Takagi - Sugeno wybrano obiekt zbiorników ustawionych w kaskadzie. Szczegółowy opis dynamiki i regulacji obiektów tego typu przedstawili autorzy w [Zbiorniki Bakun]. Dokładnie zilustrowali nieliniową dynamikę - w tym przypadku - trzech zbiorników w kaskadzie. w niniejszej pracy skupiono się na dwóch zbiornikach, jeden w kształcie walca o stałym przekroju poprzecznym, natomiast drugi w kształcie stożka o zmieniającej się powierzchni przekroju wraz z wysokością, co wprowadza dodatkową nieliniowość dynamiki. Obiekt zaprezentowano na Rys. 4.1, natomiast równania opisujące model fizyczny zapisano poniżej:

$$\begin{cases} \frac{dV_1}{dt} = F_1 + F_D - F_2(h_1) \\ \frac{dV_2}{dt} = F_2(h_1) - F_3(h_2) \\ F_2(h_1) = \alpha_1 \sqrt{h_1}, \quad F_3(h_2) = \alpha_2 \sqrt{h_2}, \quad V_1(h_1) = A_1 h_1, \quad V_2(h_2) = C_2 h_2^2, \quad F_1(t) = F_{1in}(t - \tau) \end{cases} \quad (4.1)$$

- Stałe:

$$A_1 = 540 \text{ cm}^2, \quad C_2 = 0,85, \quad \alpha_1 = 26, \quad \alpha_2 = 20 \quad (4.2)$$

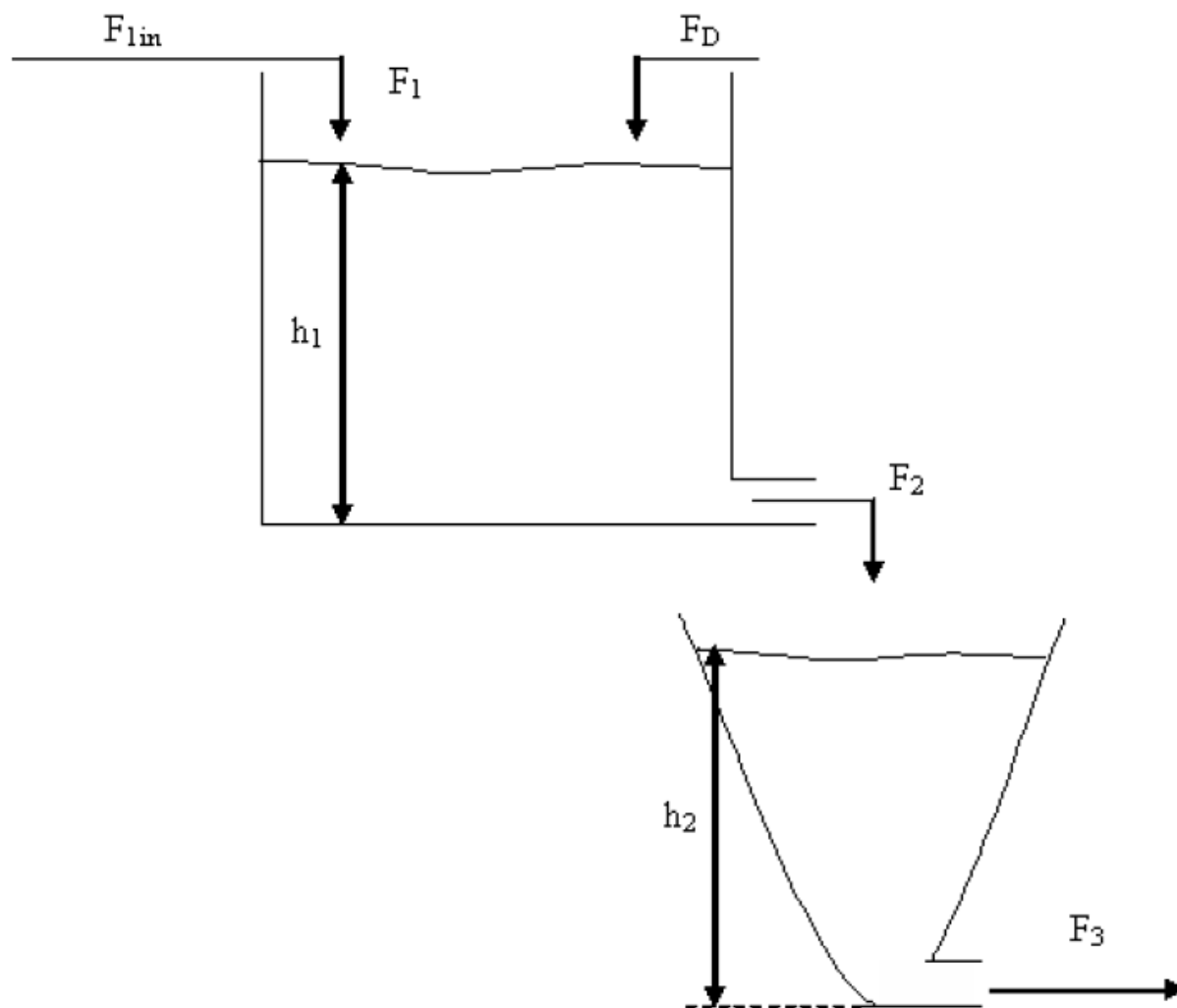
- Punkt pracy:

$$F_1 = 90 \frac{\text{cm}^3}{\text{s}}, \quad F_D = 30 \frac{\text{cm}^3}{\text{s}}, \quad \tau = 100 \text{ s}, \quad h_2 = 36 \text{ cm} \quad (4.3)$$

Wartością sterującą był dopływ F_{1in} natomiast zakłóceniem - F_D . Z kolei wyjściem - wartością regulowaną - wysokość cieczy w drugim zbiorniku h_2 . Pozostałe oznaczenia:

- A_1 - stałe pole przekroju poprzecznego pierwszego zbiornika
- C_2, α_1, α_2 - stałe opisujące przepływ cieczy między zbiornikami
- τ - stała czasowa, opisująca opóźnienie z jakim sterowanie F_1 działa na układ
- h_1 - wysokość cieczy w pierwszym zbiorniku

Użyte oznaczenia odpowiadają tym zastosowanym na rys. 4.1.



Rys. 4.1: Obiekt regulacji automatycznej.

Plan pracy podzielono na kilka kluczowych aspektów:

1. Dokonano identyfikacji modelu sprawdzono jego nieliniowość, poprzez wykreślenie charakterystyk statycznych.
2. Zlinearyzowano obiekt w punkcie pracy i wygenerowano wymuszenia skokowe, aby sprawdzić niedokładność modelu liniowego.
3. Dokonano identyfikacji modeli Hammersteina i Wienera. Nieliniową statykę zrealizowano zarówno stosując standardowe podejście - liniowe następniki modeli Takagi - Sugeno - jak i z wykorzystaniem następników hiperbolicznych.
4. Zaimplementowano regulator DMC w kilku wariantach, porównano działanie regulatorów i wybrano najlepszy.

4.1. Identyfikacja

W ramach identyfikacji obiektu wyznaczono charakterystyki statyczne oraz zbadano odpowiedź zbiorników na wymuszenia skokowe dopływów - F_1 oraz F_D . Ponadto dokonano porównania modelu liniowego i nieliniowego.

4.1.1. Charakterystyka statyczna

Charakterystyka statyczna obiektu to zależność między sygnałem wejściowym a odpowiadającym mu sygnałem wyjściowym w stanie ustalonym, czyli po zaniku procesów przejściowych. Opisuje ona zachowanie systemu w warunkach statycznych, bez uwzględniania dynamiki i opóźnień. W identyfikacji obiektu pozwala określić jego nieliniowości oraz przewidzieć, jak wyjście reaguje na różne wartości wejścia. W przedstawianej pracy poświęcono jej bardzo dużo uwagi, ze względu na istotną rolę, jaką odgrywa we wspomnianych modelach Hammersteina i Wienera. Dzięki niej można uprościć identyfikację parametrów, analizując najpierw charakterystykę statyczną, a dopiero potem właściwości dynamiczne systemu. Korzystając z modelu fizycznego, z równania 4.1 wyznaczono:

$$\frac{dV_1}{dt} = 0 \quad \wedge \quad \frac{dV_2}{dt} = 0 \quad (4.4)$$

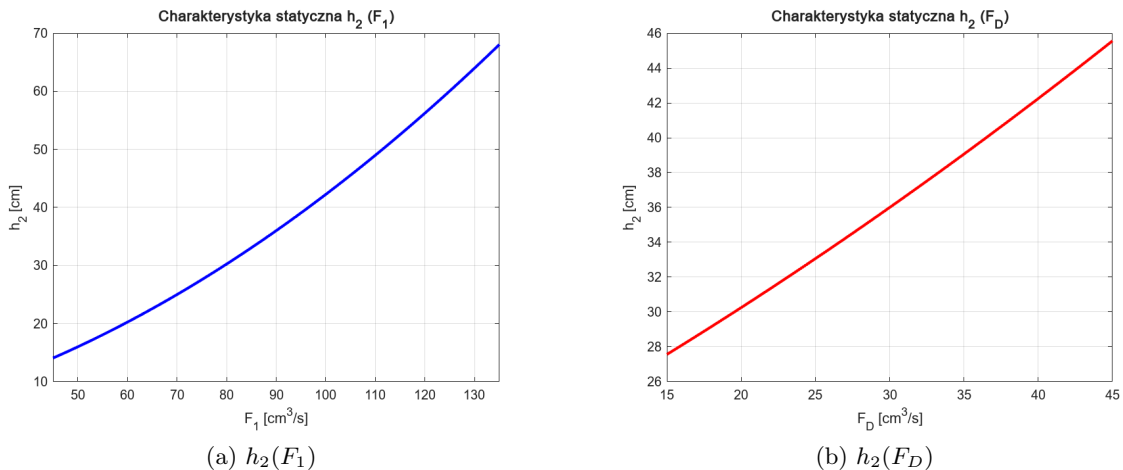
wobec tego:

$$\begin{cases} 0 &= F_1 + F_D - \alpha_1 \sqrt{h_1} \\ 0 &= \alpha_1 \sqrt{h_1} - \alpha_2 \sqrt{h_2} \end{cases} \quad (4.5)$$

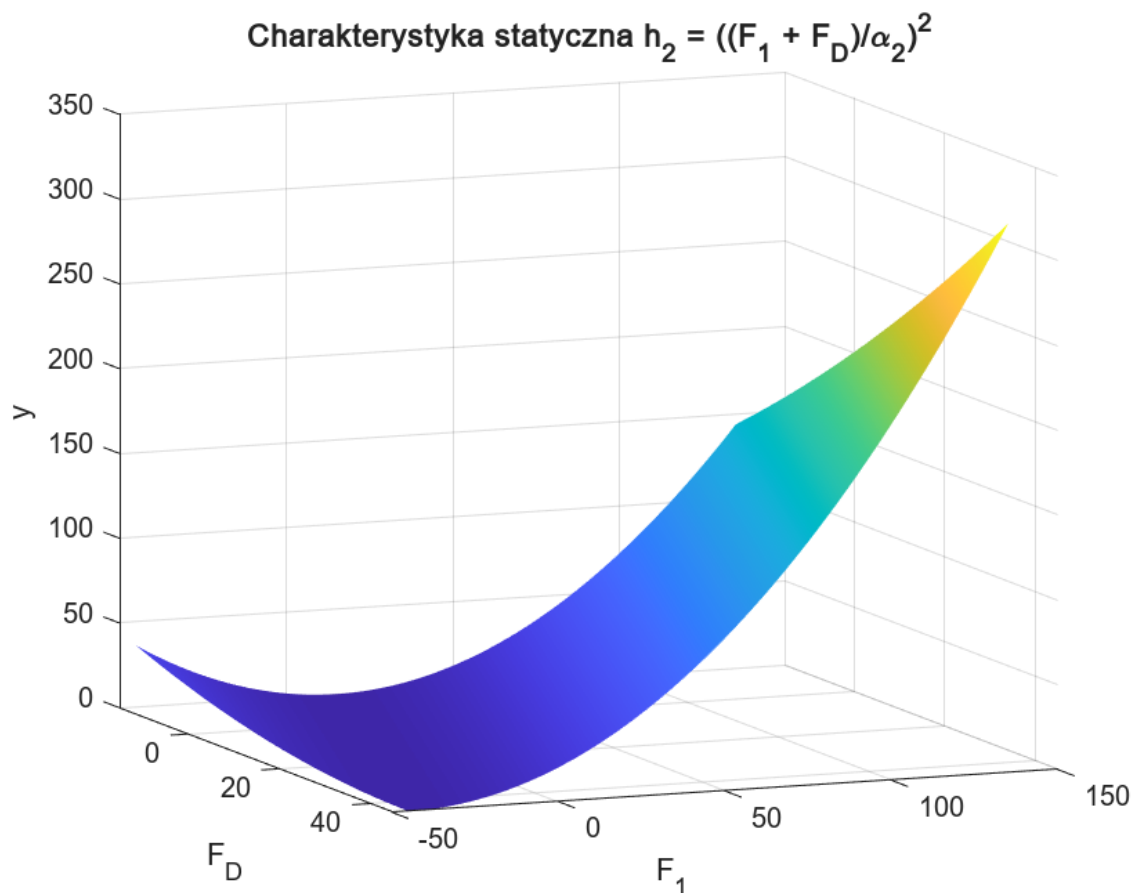
Po przekształceniach otrzymano wzór opisujący charakterystykę statyczną:

$$h_2 = \left(\frac{F_1 + F_D}{\alpha_2} \right)^2 \quad (4.6)$$

Wykres odpowiadający wyprowadzonemu wzorowi, w zależności od danej zmiennej - sterowanie lub zakłócenie - prezentuje się następująco:



Rys. 4.2: Charakterystyka statyczna.



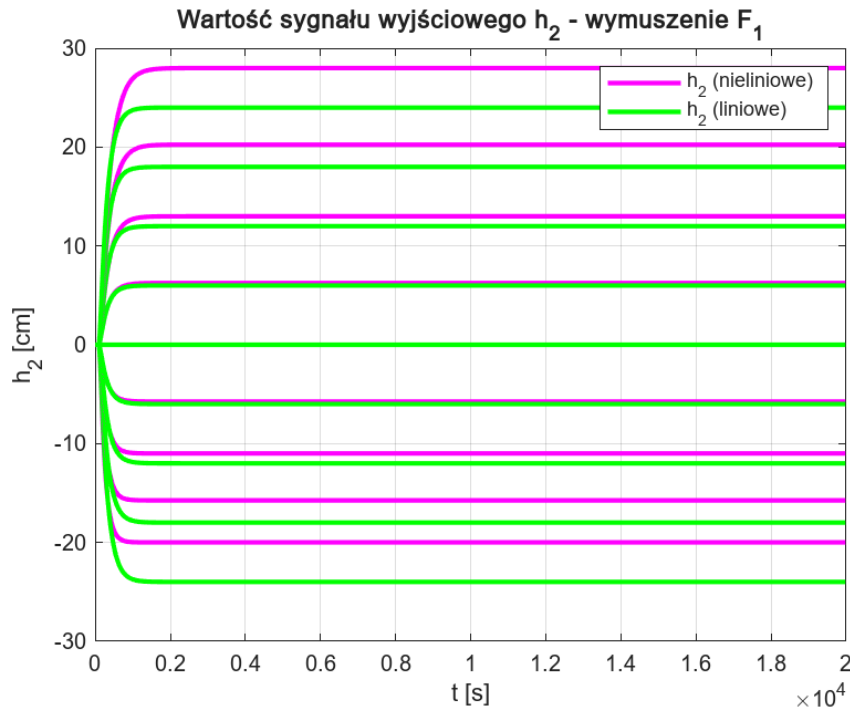
Rys. 4.3: Trójwymiarowa charakterystyka statyczna.

Otrzymano spodziewane przebiegi, mianowicie w obu przypadkach wykresy przedstawiają parabole. Podczas wyznaczania charakterystyk założono przedziały zmienności zmiennej sterującej oraz zakłócenia:

- $F_1 \in [45, 135]$, co odpowiada przyrostowi $\Delta F_1 \in [-45, 45]$
- $F_D \in [15, 45]$, co odpowiada przyrostowi $\Delta F_D \in [-15, 15]$

4.1.2. Porównanie modelu liniowego i nieliniowego

Wykresy charakterystyk statycznych uzupełniono serią wymuszeń skokowych, które prezentowały się następująco:



Rys. 4.4: Porównanie odpowiedzi obiektu nieliniowego i zlinearyzowanego w punkcie pracy na wymuszenia skokowe sygnału F_1 .

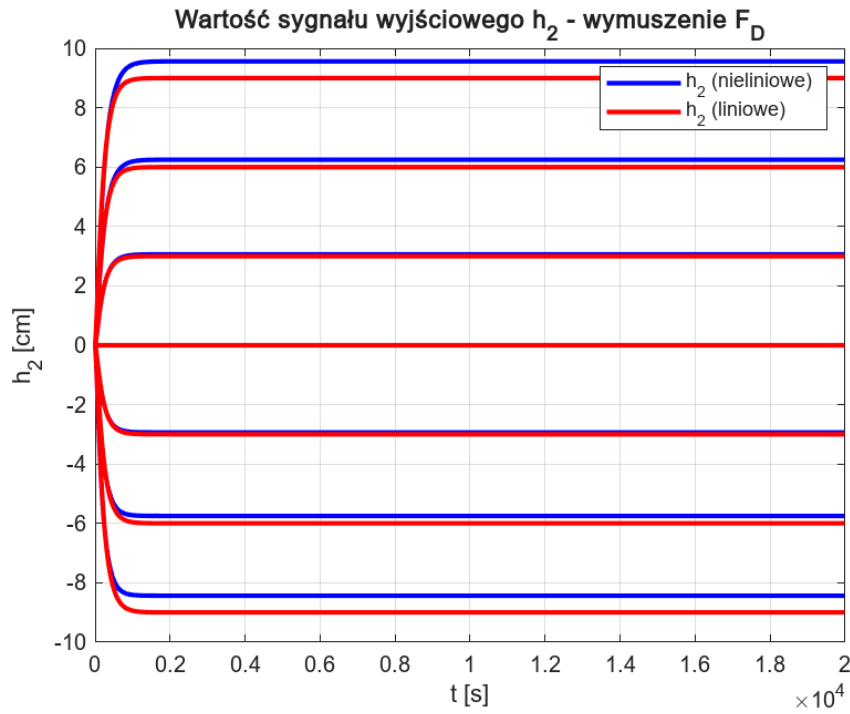
Otrzymano spodziewane przebiegi, mianowicie, blisko punktu linearyzacji model liniowy odzwierciedla rzeczywiste zachowanie układu, natomiast wraz z oddalaniem się od punktu pracy ta dokładność maleje. Utratę tę dokładności przedstawiono w Tab. 4.1, obliczając dla każdej pary odpowiedzi liniowej i nieliniowej, błąd średniokwadratowy, tj.

$$E = \frac{1}{kk} \sum_{k=1}^{kk} (y(k) - y_L(k))^2 \quad (4.7)$$

| Wymuszenie | Błąd |
|---------------|--------|
| $F_{10} - 40$ | 15,427 |
| $F_{10} - 30$ | 4,875 |
| $F_{10} - 20$ | 0,962 |
| $F_{10} - 10$ | 0,060 |
| F_{10} | 0,000 |
| $F_{10} + 10$ | 0,060 |
| $F_{10} + 20$ | 0,956 |
| $F_{10} + 30$ | 4,830 |
| $F_{10} + 40$ | 15,235 |

Tab. 4.1: Wartość błędu średnio kwadratowego w zależności od wielkości wymuszenia F_1 .

Charakter odpowiedzi obiektu na wymuszenia F_D wyglądał analogicznie do, wcześniej zaprezentowanych, wymuszeń F_1 .



Rys. 4.5: Porównanie odpowiedzi obiektu nieliniowego i zlinearyzowanego w punkcie pracy na wymuszenia skokowe sygnału F_D .

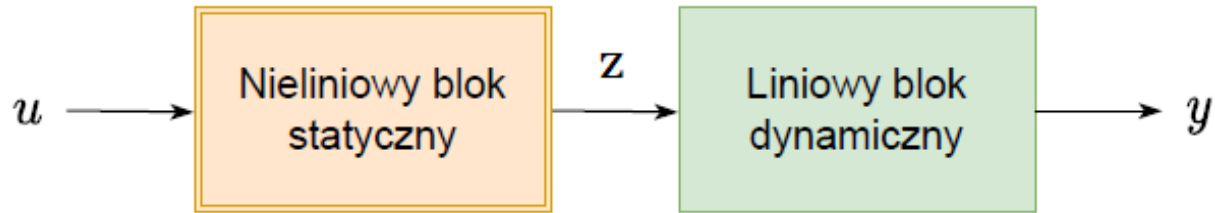
| Wymuszenie | Błąd |
|---------------|-------|
| $F_{10} - 15$ | 0,306 |
| $F_{10} - 10$ | 0,060 |
| $F_{10} - 5$ | 0,004 |
| F_{10} | 0,000 |
| $F_{10} + 5$ | 0,004 |
| $F_{10} + 10$ | 0,060 |
| $F_{10} + 15$ | 0,304 |

Tab. 4.2: Wartość błędu średnio kwadratowego w zależności od wielkości wymuszenia F_D .

Z racji na założony mniejszy przedział zmienności sygnału zakłócającego, różnice między modelem liniowym, a nieliniowym nie były tak duże jak w przypadku sygnału sterującego.

4.2. Model Hammersteina

Po przeanalizowaniu charakterystyki statycznej oraz porównaniu modelu nieliniowego z liniowym przystąpiono do identyfikacji modelu Hammersteina. Istota polega na umieszczeniu nieliniowego bloku statycznego przed liniowym blokiem dynamicznym, co pozwala na oddzielenie nieliniowości od dynamiki systemu. Jego główną zaletą jest prostsza identyfikacja parametrów, ponieważ najpierw określa się charakterystykę statyczną, a dopiero potem analizuje dynamikę. Takie podejście lepiej odwzorowuje systemy, w których nieliniowości wynikają z właściwości aktuatorów lub czujników, a część dynamiczna pozostaje liniowa. Graficzne ujęcie opisanego modelu zilustrowano na rys. 4.5.



Rys. 4.6: Reprezentacja graficzna modelu Hammersteina.

Cała procedura w przypadku analizowanego obiektu wygląda następująco, sygnał sterujący jest wejściem nieliniowego bloku statycznego, którego wyjściem jest przekonwertowany sygnał $z = f(u)$. Następnie sygnał z trafia do liniowego bloku dynamicznego. Dzięki temu rozdzielaniu nieliniowości i dynamiki, możliwe jest zastosowanie klasycznych metod projektowania regulatorów dla części dynamicznej, co upraszcza proces sterowania.

4.2.1. Nieliniowy blok statyczny

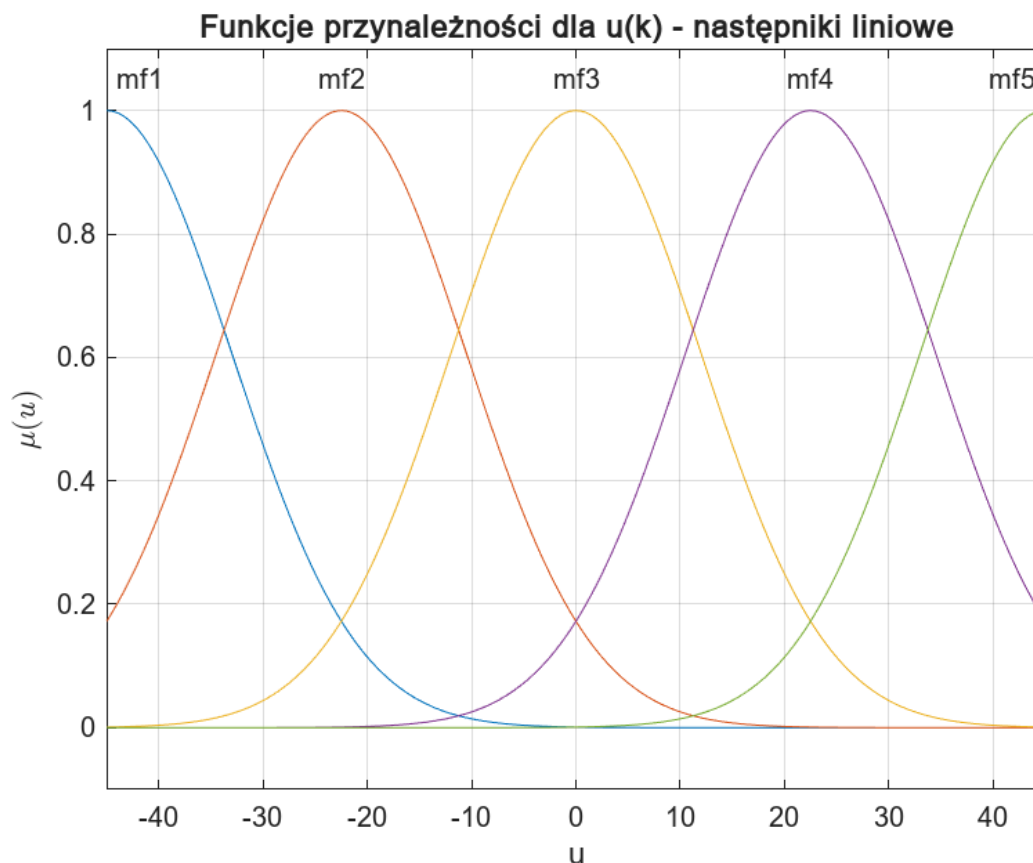
Nieliniowość w charakterystyce statycznej została wprowadzona za pomocą logiki rozmytej (ang. *fuzzy logic*), a konkretnie za pomocą modeli rozmytych Takagi-Sugeno. Zastosowano dwa podejścia, jedno standardowe z następnikami liniowymi, natomiast drugie z następnikami hiperbolicznymi.

4.2.2. Następniki liniowe

W standardowej wersji modeli Takagi-Sugeno następniki przyjmują liniową postać, dlatego to właśnie od nich postanowiono zacząć. Rozmyto zmienną wejściową oraz wybrano odpowiednią liczbę zbiorów rozmytych. Zastosowano następniki liniowe postaci:

$$\begin{aligned}
 \text{Reguła 1: Jeśli } u(k) \text{ jest } U_1, \text{ to : } y^1(k) &= a_1 u(k) + b_1 \\
 \text{Reguła 2: Jeśli } u(k) \text{ jest } U_2, \text{ to : } y^2(k) &= a_2 u(k) + b_2 \\
 &\vdots \\
 \text{Reguła 5: Jeśli } u(k) \text{ jest } U_5, \text{ to : } y^5(k) &= a_5 u(k) + b_5
 \end{aligned} \tag{4.8}$$

Natomiast wyjście systemu rozmytego obliczano zgodnie ze wzorem 3.28.



Rys. 4.7: Zbiory rozmyte - następni liniowe.

Zarówno do budowy modelu, jak i wyznaczenia parametrów następników wykorzystano narzędzia oferowane przez MATLAB w ramach *Fuzzy Logic Toolbox*. Korzystając z funkcji `sugfis()` zbudowano nieliniowy model rozmyty typu Takagi - Sugeno. Zdecydowano się na pięć zbiorów rozmytych o gaussowskim kształcie, co zapewnia różniczkowalność (SZAU). Następnie, dzięki wykorzystaniu `addInput()`, `addOutput()`, `addMF()` udało się zbudować bazę reguł - `addRule()`, co bezpośrednio przełożyło się na wyznaczenie współczynników pierwszej iteracji. Konieczne było późniejsze ręczne dostrajanie modelu, które przy względnie dużej liczbie zbiorów nie przysporzyło dużo problemów. Ostatecznie zdefiniowano następujące wartości parametrów następników:

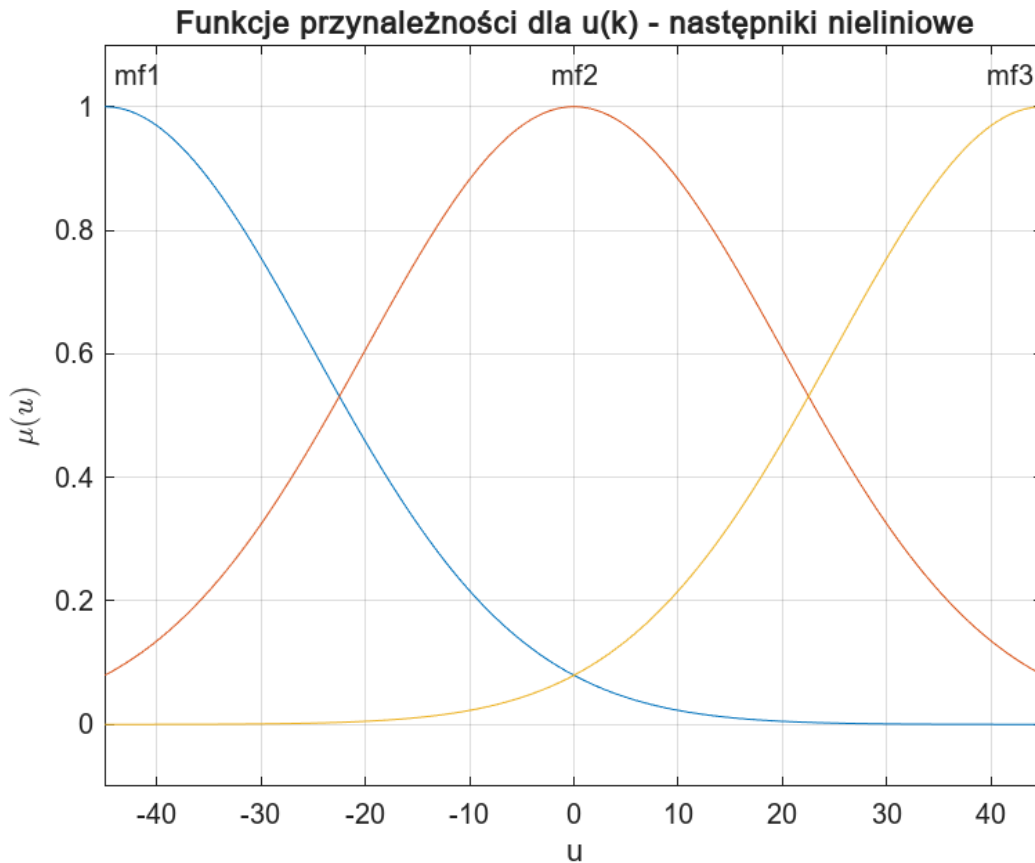
| Nr reguły | Współczynnik a_r | Współczynnik b_r |
|-----------|--------------------|--------------------|
| 1 | 0,7895 | 0,0001 |
| 2 | 0,8982 | 0,0002 |
| 3 | 1,0933 | 0,0001 |
| 4 | 1,0489 | 0 |
| 5 | 1,2034 | 0,0001 |

4.2.3. Następniki nieliniowe

Wprowadzając następniki w postaci hiperbolicznej spodziewano się zachowania dokładności przy jednoczesnym zmniejszeniu liczby zbiorów rozmytych [Robust observer-based controller design for Takagi–Sugeno systems with nonlinear consequent parts]. Sformułowano następującą bazę reguł:

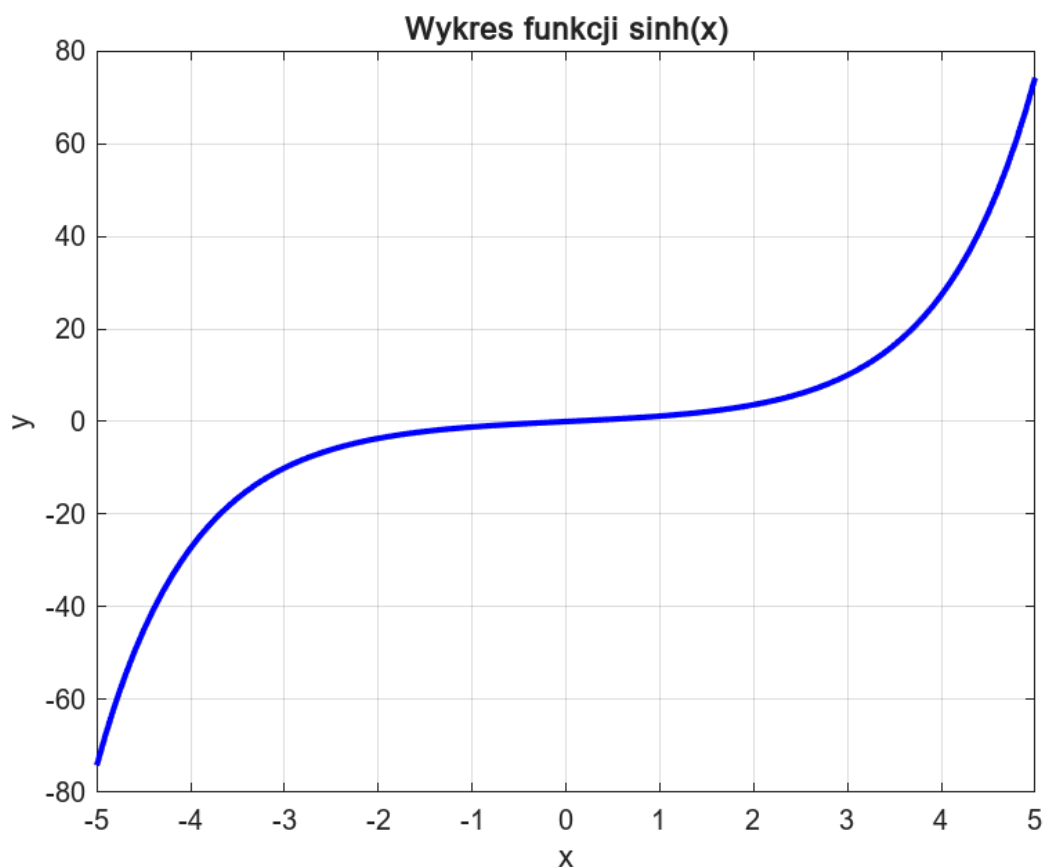
$$\begin{aligned}
 \text{Reguła 1: Jeśli } u(k) \text{ jest } U_1, \text{ to : } y^1(k) &= a_1 \sinh\left(\frac{u(k)}{b_1}\right) \\
 \text{Reguła 2: Jeśli } u(k) \text{ jest } U_2, \text{ to : } y^2(k) &= a_2 \sinh\left(\frac{u(k)}{b_2}\right) \\
 \text{Reguła 3: Jeśli } u(k) \text{ jest } U_3, \text{ to : } y^3(k) &= a_3 \sinh\left(\frac{u(k)}{b_3}\right)
 \end{aligned} \tag{4.9}$$

Zgodnie z oczekiwaniami udało się wprowadzić mniejszą liczbę zbiorów rozmytych.



Rys. 4.8: Zbiory rozmyte - następniki nieliniowe.

Procedura dostrajania parametrów następników różniła się w stosunku do odpowiedników liniowych. Wynikało to z tego, że wykorzystywane narzędzie do budowy modelu rozmytego domyślnie stroi parametry dla następników liniowych, stąd konieczność znacznej modyfikacji i ręcznego dostrajania. Następnie, po ręcznym dostrojeniu wykorzystano funkcję z pakietu *Optimization Toolbox*, mianowicie `fminsearch()`. Wybrano taką kolejność ze względu na fakt, że odwracając kolejność, tzn. stosując najpierw metodę Nelder-Meada, otrzymywane wyniki były gorsze niż te wybrane ręcznie. Było to spowodowane skłonnością do wpadania algorytmu w minima lokalne. Istotnym aspektem w tym przypadku okazała się normalizacja argumentu, bowiem kształt funkcji $\sinh()$ zmienia się bardzo gwałtownie dla rosnących wartości zmiennej (rys. 4.8).

Rys. 4.9: Wykres funkcji $\sinh()$.

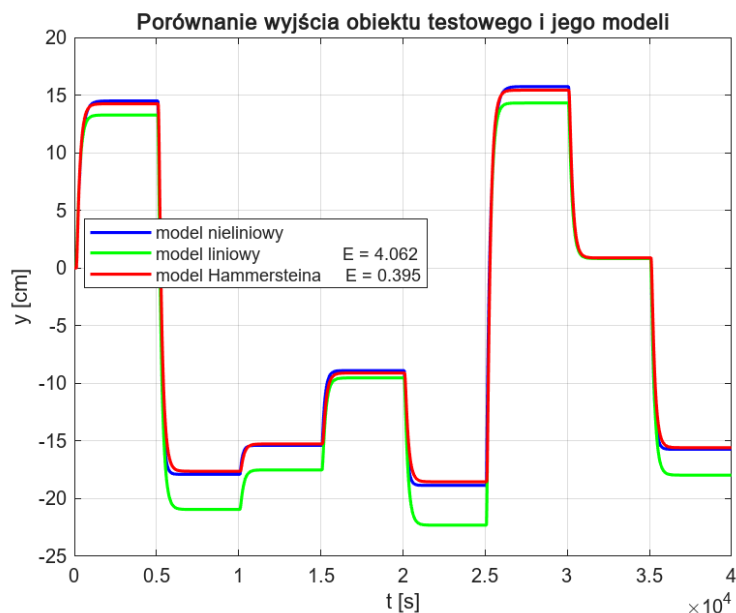
Ostateczne wartości współczynników następników reguł zebrano w tab. 4.3.

| Nr reguły | Współczynnik a_r | Współczynnik b_r |
|-----------|--------------------|--------------------|
| 1. | 46,4941 | 62,8862 |
| 2. | 36,2347 | 36,2592 |
| 3. | 110,4841 | 98,7690 |

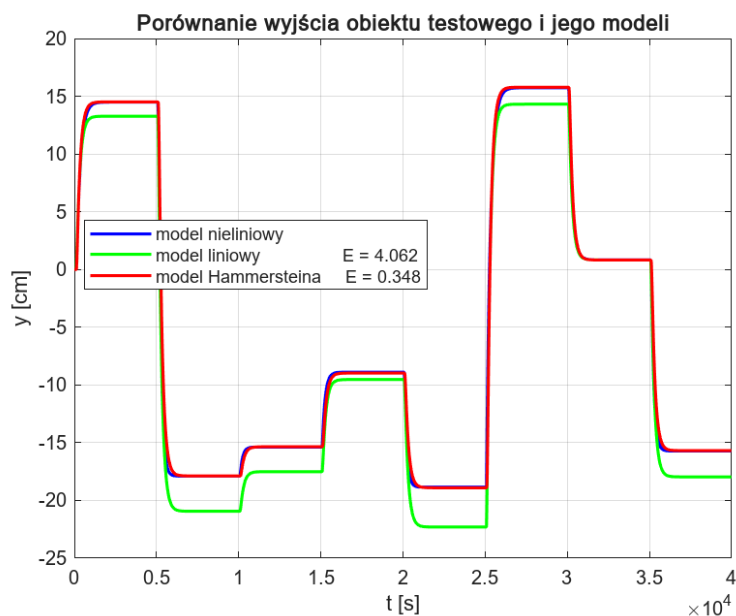
Tab. 4.3: Współczynniki hiperbolicznych następników reguł.

4.2.4. Porównanie

Dostroiwszy oba modele przyszedł czas na ich porównanie. Wygenerowano pięć sekwencji losowo zmieniającego się sygnału sterującego, następnie wynik został porównany do rezultatu uzyskanego dla modelu nieliniowego - wzorcowego - obliczonego za pomocą zmodyfikowanej metody Eulera. Wskaźnikiem porównawczym był błąd średnio kwadratowy. Aby zaznaczyć jakie korzyści wnosi nieliniowa część modelu Hammersteina na dokładność modelowania, w każdej sekwencji obliczono także wskaźnik dla modelu liniowego.

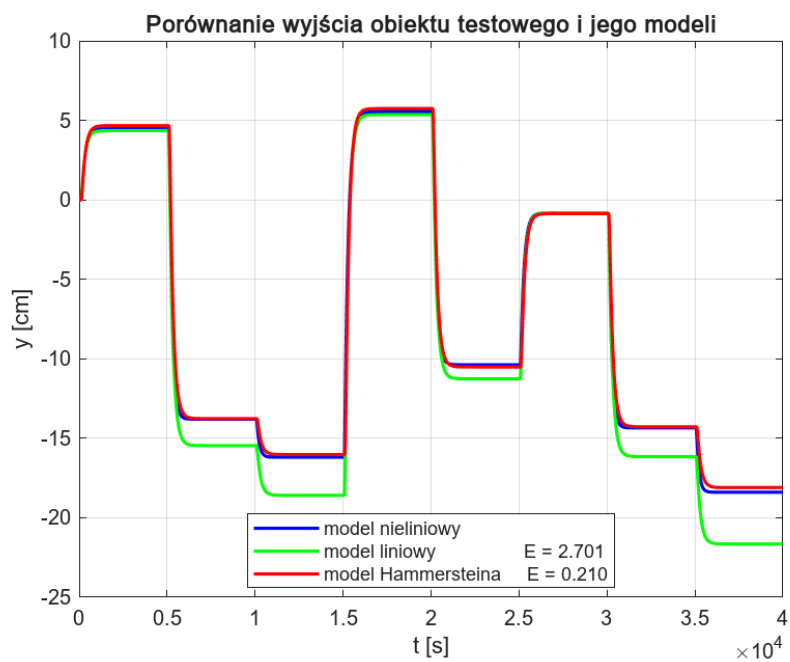


(a) Następniki liniowe

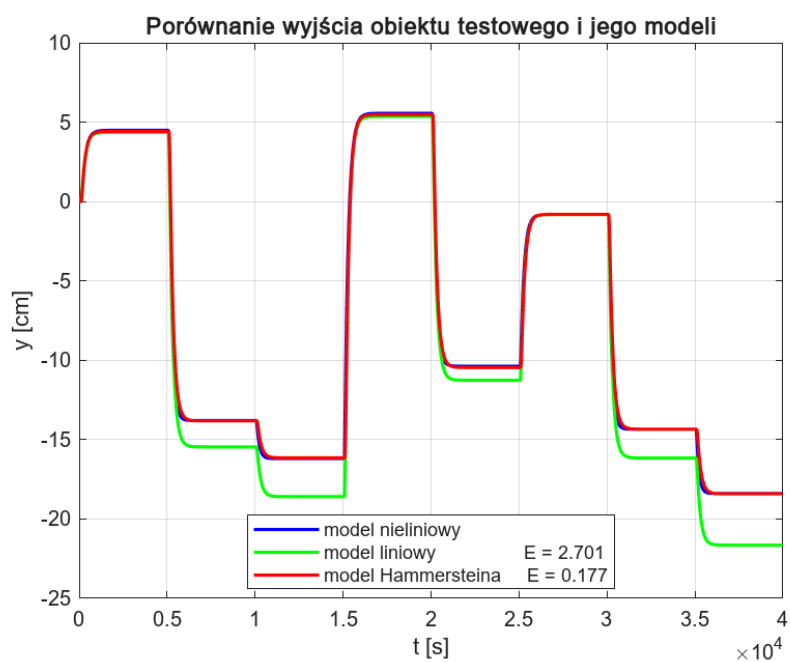


(b) Następniki nieliniowe

Rys. 4.10: Porównanie modelu Hammersteina z następnikami liniowymi i nieliniowymi - pierwsza sekwencja.

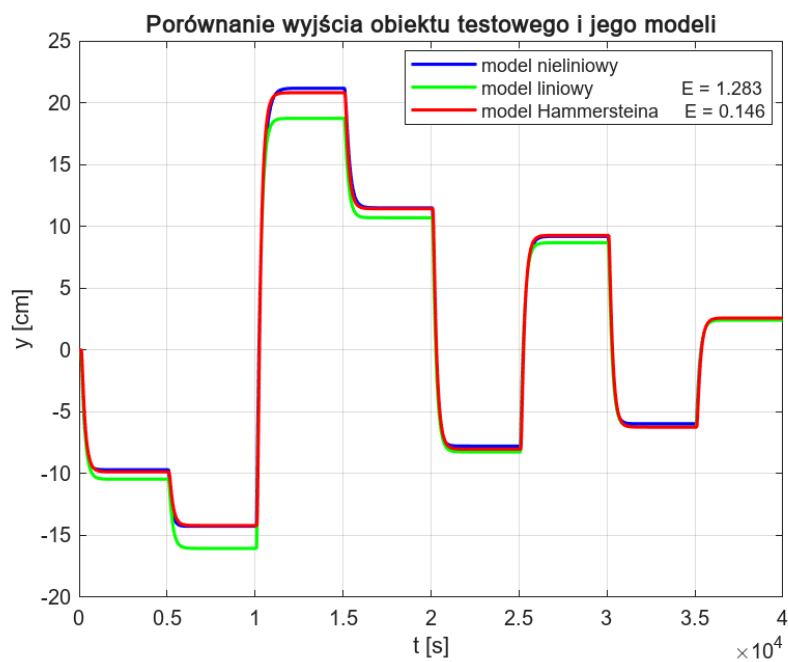


(a) Następniki liniowe

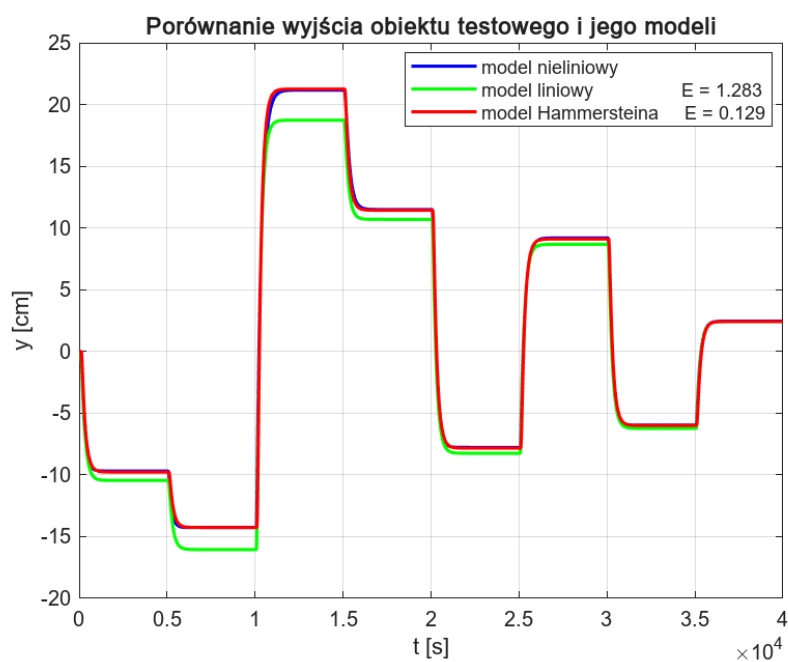


(b) Następniki nieliniowe

Rys. 4.11: Porównanie modelu Hammersteina z następnikami liniowymi i nieliniowymi - druga sekwencja.

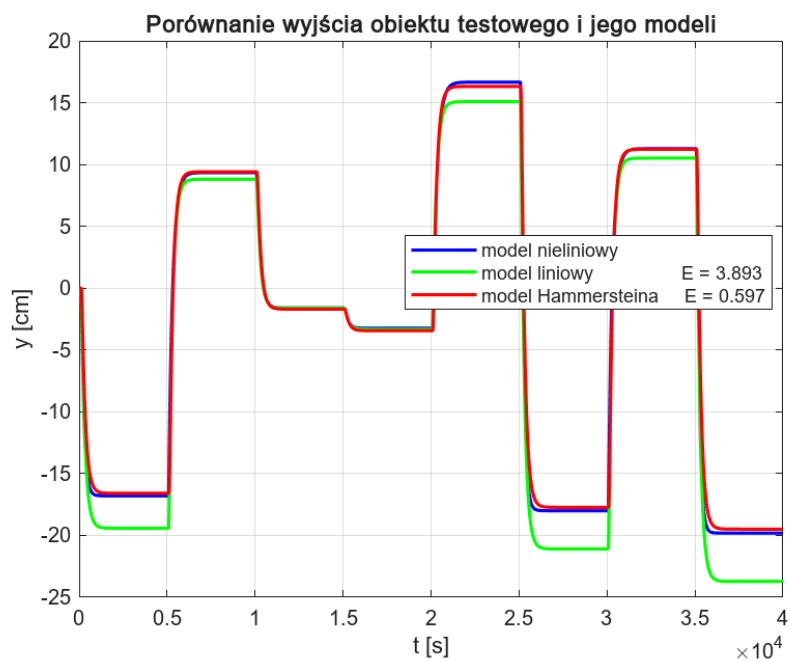


(a) Następniki liniowe

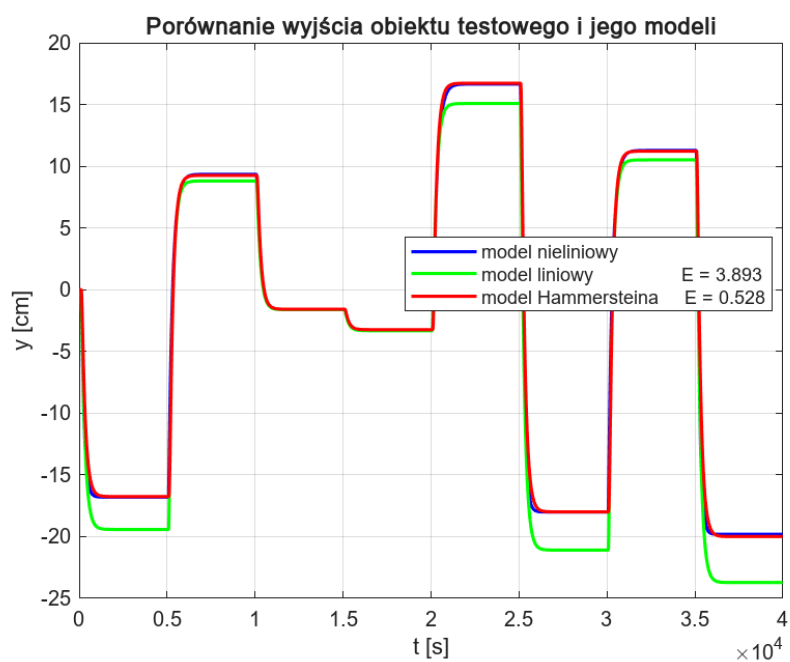


(b) Następniki nieliniowe

Rys. 4.12: Porównanie modelu Hammersteina z następnikami liniowymi i nieliniowymi - trzecia sekwencja.

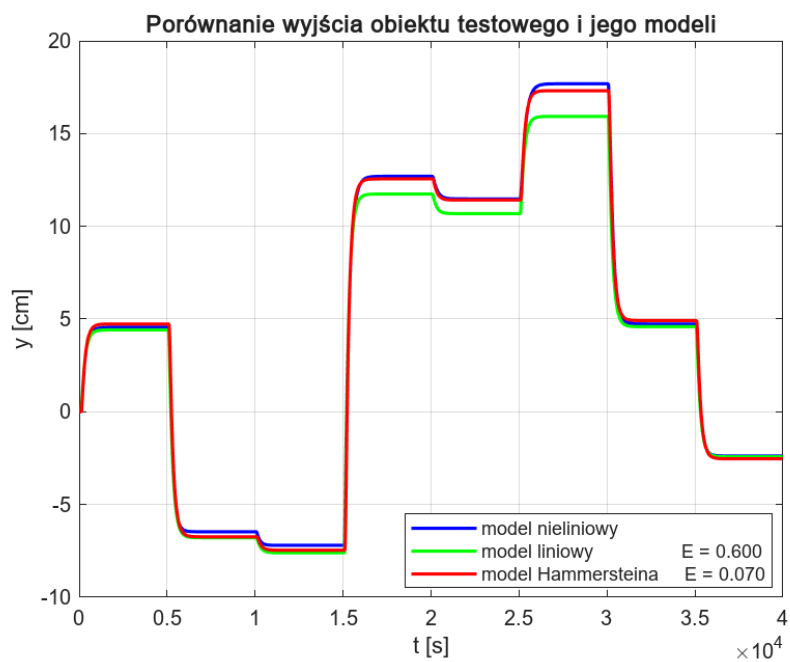


(a) Następniki liniowe

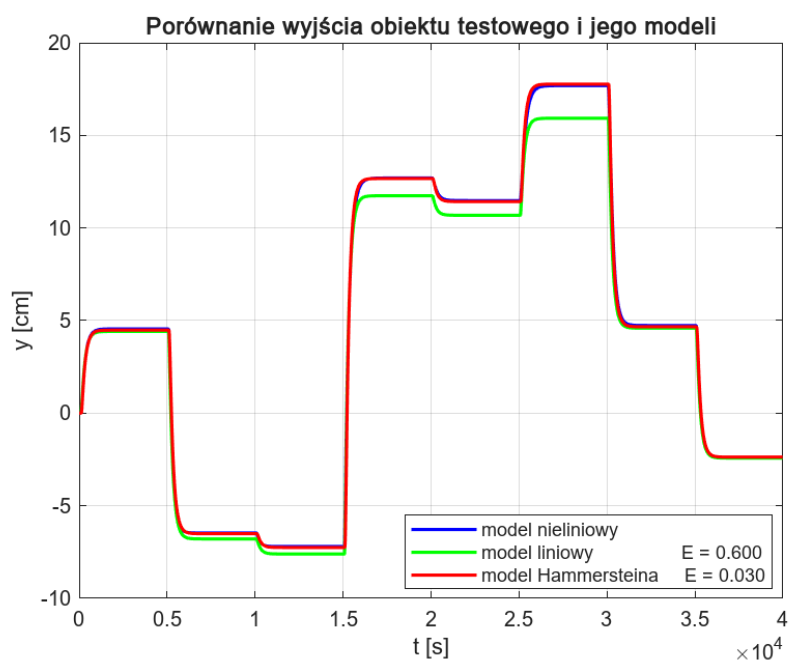


(b) Następniki nieliniowe

Rys. 4.13: Porównanie modelu Hammersteina z następnikami liniowymi i nieliniowymi - czwarta sekwencja.



(a) Następniki liniowe



(b) Następniki nieliniowe

Rys. 4.14: Porównanie modelu Hammersteina z następnikami liniowymi i nieliniowymi - piąta sekwencja.

Zaprezentowane wykresy na rys. 4.9 - 4.13 ilustrują przede wszystkim zysk zastosowania modelu Hammersteina do opisu obiektu. Otrzymane wyniki są nieporównywalnie lepsze w stosunku do modelu liniowego. Natomiast przyglądając się z bliska, można zauważyć również poprawę dzięki wprowadzeniu dodatkowej nieliniowości w definicji następników rozmytego modelu typu Takagi-Sugeno. W tab. 4.4 zebrano uzyskane wyniki.

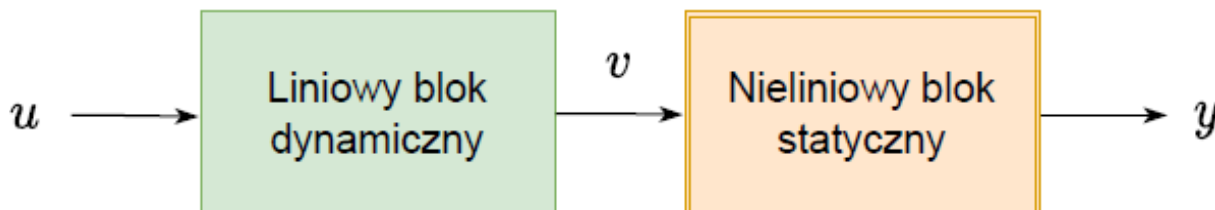
| Nr sekwencji | Model liniowy | Model Hammersteina | |
|--------------|---------------|--------------------|-----------------------|
| | | Następniki liniowe | Następniki nieliniowe |
| 1. | 4,062 | 0,395 | 0,348 |
| 2. | 2,701 | 0,210 | 0,177 |
| 3. | 1,283 | 0,146 | 0,129 |
| 4. | 3,893 | 0,597 | 0,528 |
| 5. | 0,600 | 0,070 | 0,030 |

Tab. 4.4: Porównanie modeli.

Należy pamiętać, że w przypadku następników hiperbolicznych zredukowano liczbę zbiorów rozmytych. Zatem udało się poprawić dokładność modelowania, przy jednoczesnym zmniejszeniu liczby reguł modelu TS.

4.3. Model Wienera

Model Wienera to struktura nieliniowego systemu, w której liniowy blok dynamiczny jest umieszczony przed nieliniowym blokiem statycznym. Oznacza to, że najpierw wejście przechodzi przez liniowy filtr, a następnie poddawane jest nieliniowej transformacji. Modele Wienera są szczególnie użyteczne w systemach, gdzie obserwowana nieliniowość wynika głównie z ograniczeń lub nasycenia elementów wykonawczych, takich jak silniki, zawory, czy inne elementy mechaniczne. Zatem istota tego modelu jest taka sama jak w przypadku modelu Hammersteina - oddzielenie nieliniowej statyki od liniowej dynamiki - z tym, że nieliniową statykę poprzedzono liniową dynamiką - odwrotnie niż jak to było w przypadku omówionego modelu, co graficznie zilustrowano na rys. 4.14.



Rys. 4.15: Reprezentacja graficzna modelu Wienera.

Sygnał wejściowy trafia na liniowy blok dynamiczny, którego wyjściem jest przekonwertowany sygnał $v = f(u)$, który następnie trafia na nieliniową statykę, której wyjściem jest sygnał y .

Warto podkreślić, że model Wienera jest trudniejszy w interpretacji niż model Hammersteina. Wynika to z tego, że sygnał wejściowy przechodzi najpierw przez system liniowy, a dopiero potem przez nieliniowy element, co utrudnia analizę wpływu nieliniowości. Dodatkowo, sygnał po filtracji liniowej może mieć bardziej złożone właściwości, co komplikuje ocenę jego zachowania w bloku nieliniowym, a także może przełożyć się na bardziej wymagającą estymację parametrów w modelu.

4.3.1. Następniki

Zarówno w przypadku następników liniowych, jak i hiperbolicznych wykorzystano dokładnie te same definicje, jakie zastosowano w przypadku modelu Hammersteina - opisane wzorami 4.8 oraz 4.9. Przypomniano jedynie ogólną strukturę poszczególnych reguł.

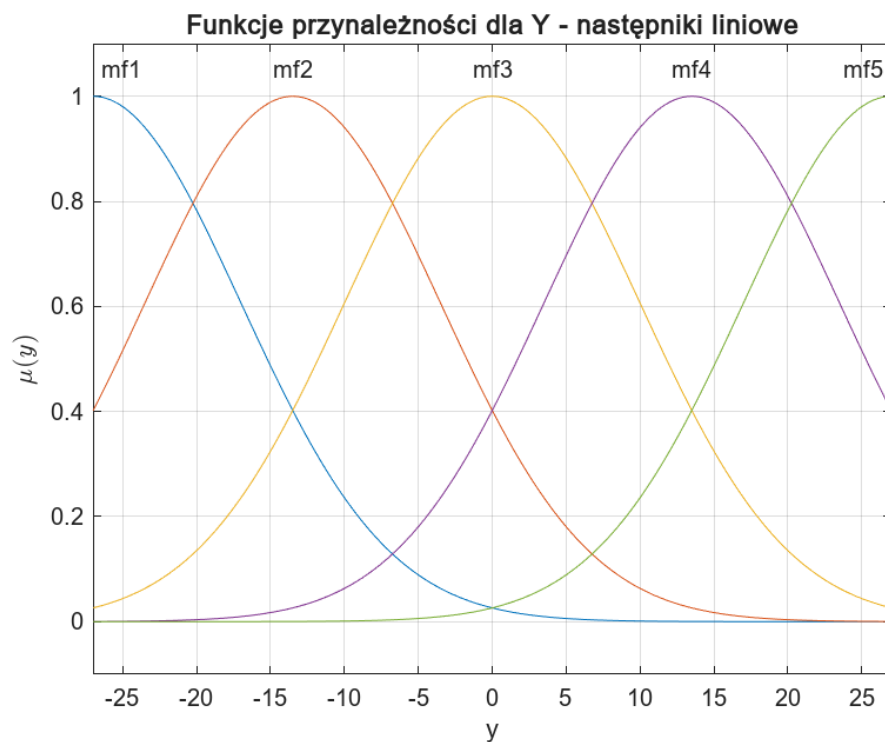
Następniki liniowe:

$$\text{Reguła n: Jeśli } u(k) \text{ jest } U_n, \text{ to : } y^n(k) = a_n u(k) + b_n \quad (4.10)$$

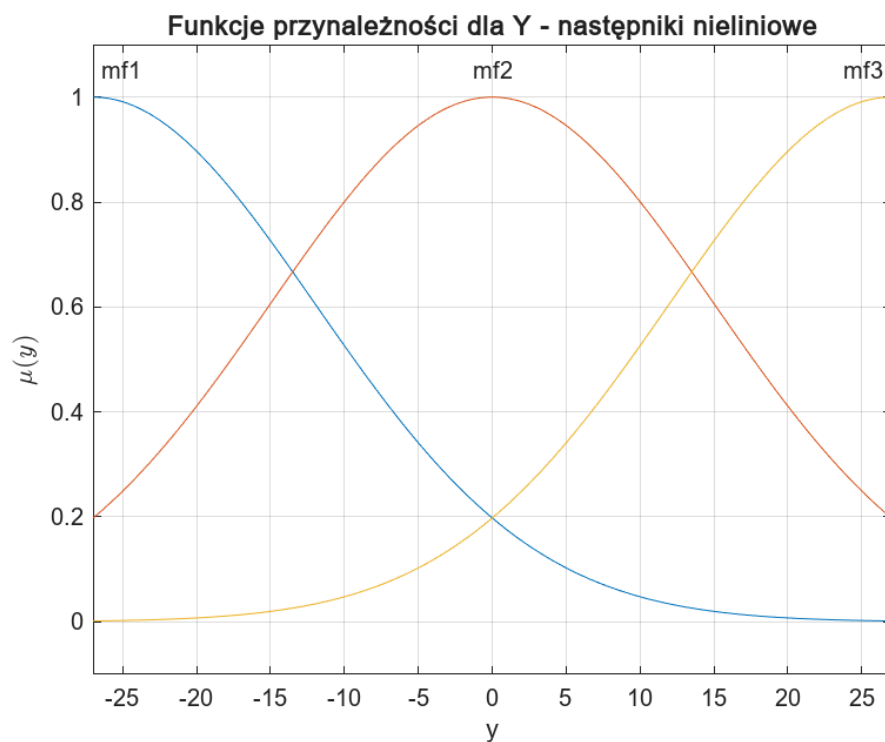
Następniki hiperboliczne:

$$\text{Reguła n: Jeśli } u(k) \text{ jest } U_n, \text{ to : } y^n(k) = a_n \sinh\left(\frac{u(k)}{b_n}\right) \quad (4.11)$$

Nie uległa zmianie również liczba i postać zbiorów rozmytych - zmienił się jedynie zakres rozmywania.



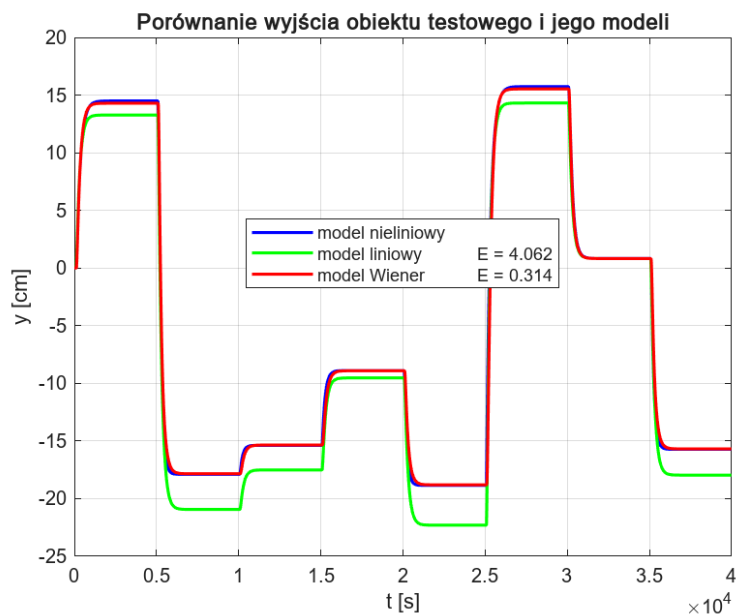
Rys. 4.16: Zbiory rozmyte - następniki liniowe.



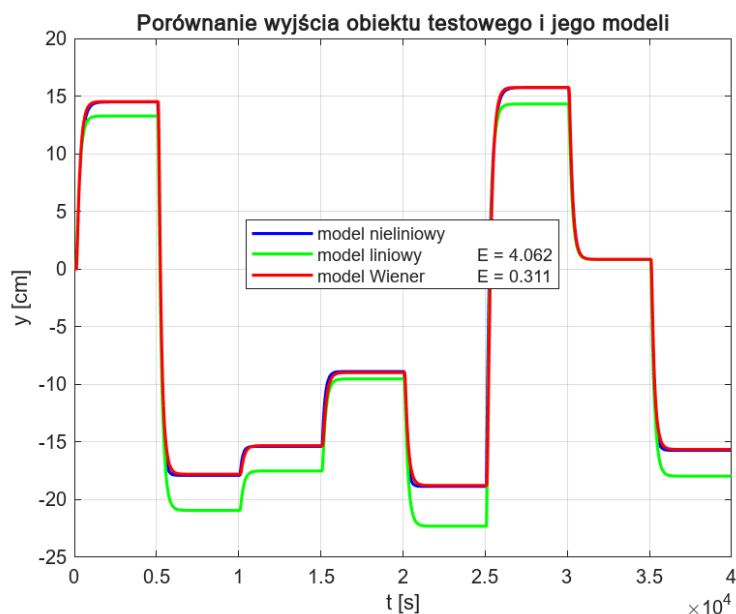
Rys. 4.17: Zbiory rozmyte - następniki nieliniowe.

4.3.2. Porównanie

Procedura porównawcza wyglądała identycznie jak w przypadku modelu Hammersteina, tj. dla tych samych wygenerowanych sekwencji sygnału sterującego wyznaczono odpowiedzi modelu nieliniowego, liniowego, modelu Wienera z następnikami liniowymi oraz hiperbolicznymi. Przebiegi przedstawiono na rys. 4.17 - 4.21, natomiast dane liczbowe zgromadzono w tab. 4.5.

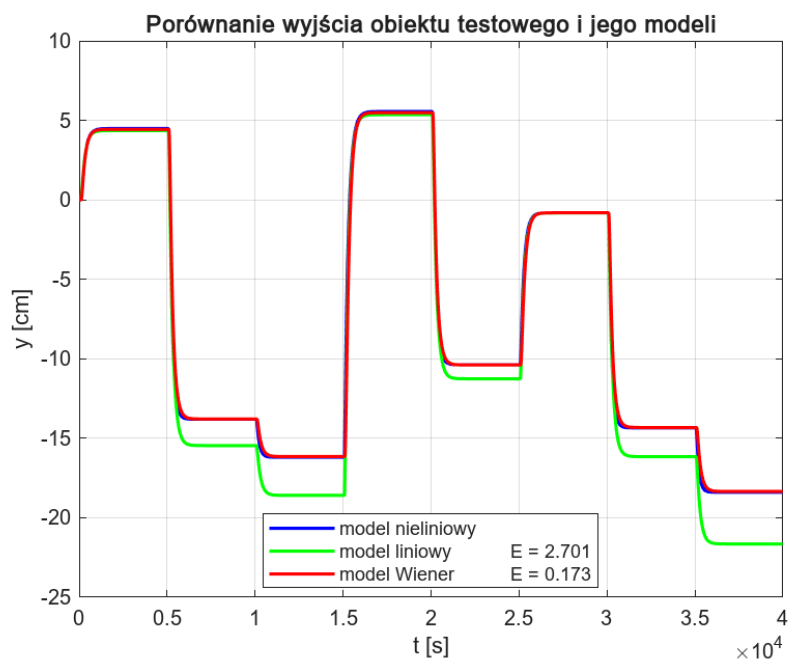


(a) Następniki liniowe

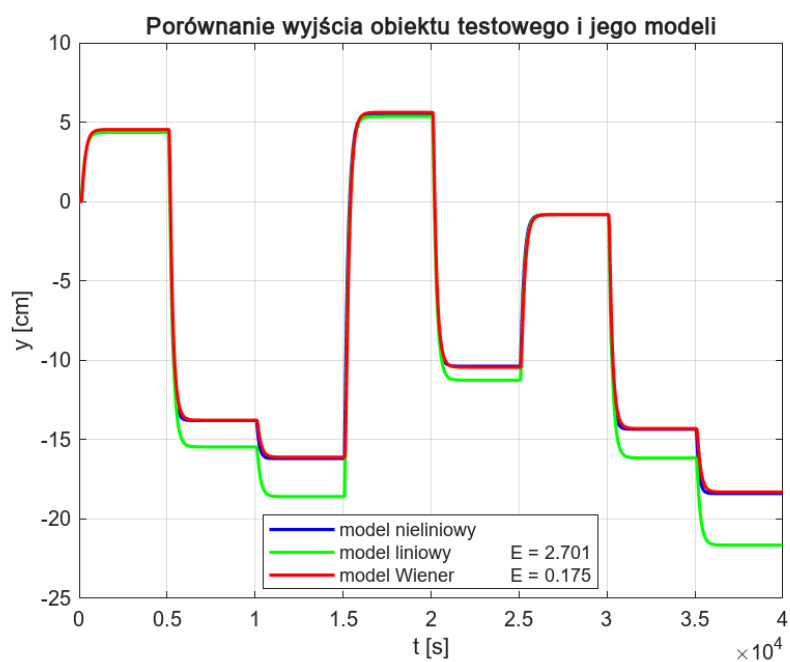


(b) Następniki nieliniowe

Rys. 4.18: Porównanie modelu Wienera z następnikami liniowymi i nieliniowymi - pierwsza sekwencja.

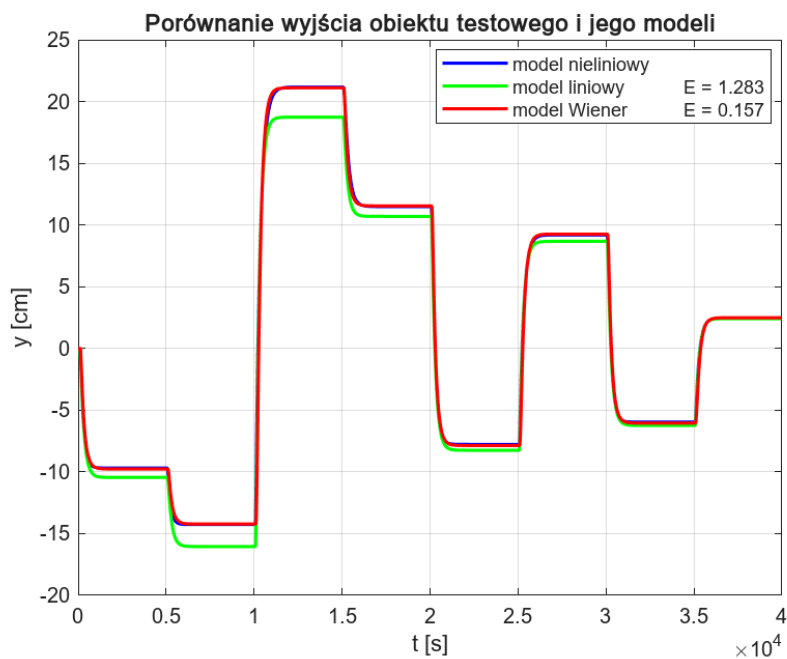
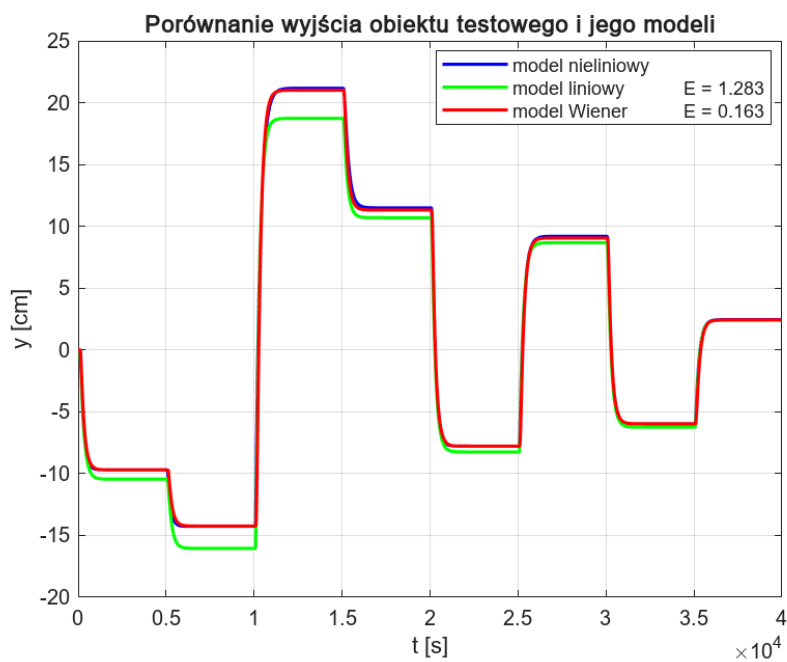


(a) Następniki liniowe

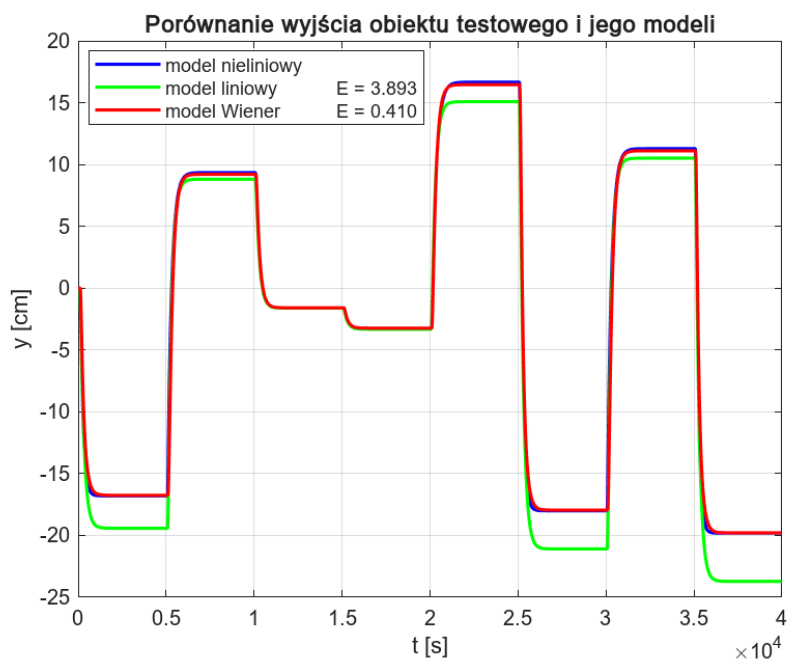


(b) Następniki nieliniowe

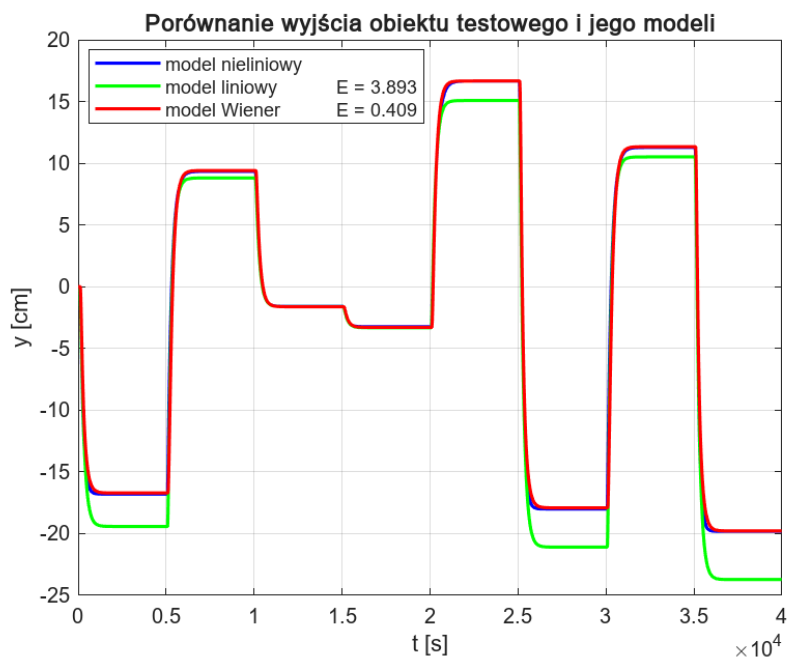
Rys. 4.19: Porównanie modelu Wienera z następnikami liniowymi i nieliniowymi - druga sekwencja.



Rys. 4.20: Porównanie modelu Wienera z następnikami liniowymi i nieliniowymi - trzecia sekwencja.

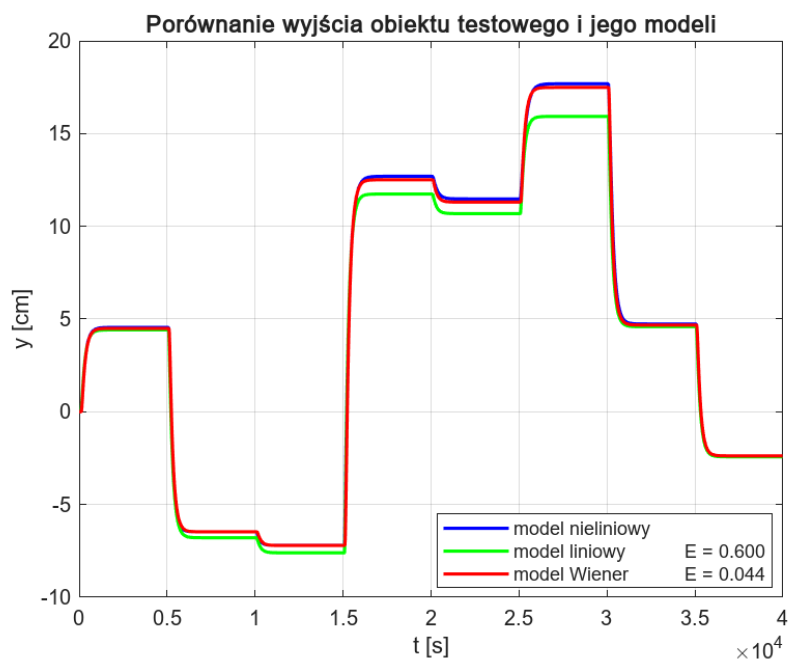


(a) Następniki liniowe

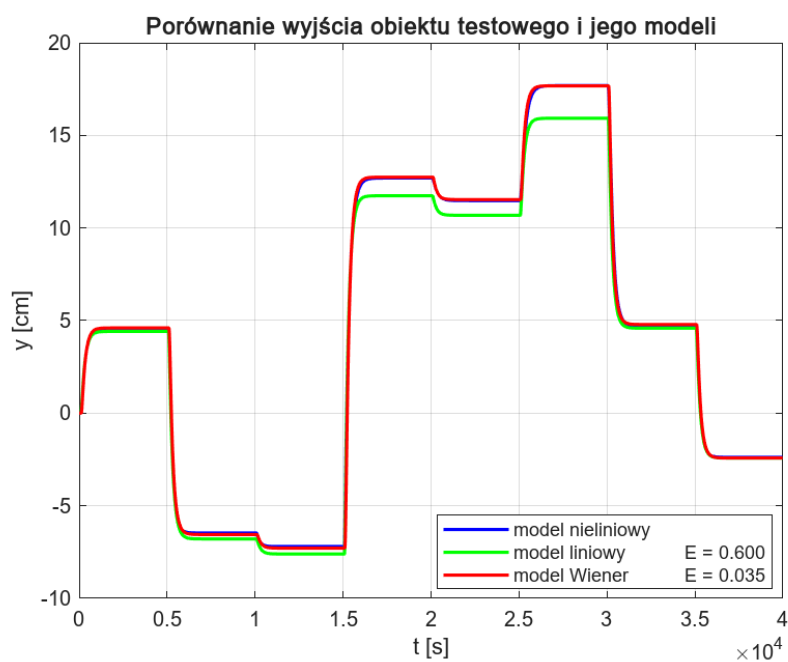


(b) Następniki nieliniowe

Rys. 4.21: Porównanie modelu Wienera z następnikami liniowymi i nieliniowymi - czwarta sekwencja.



(a) Następniki liniowe



(b) Następniki nieliniowe

Rys. 4.22: Porównanie modelu Wienera z następnikami liniowymi i nieliniowymi - piąta sekwencja.

Wyniki zebrane w tab. 4.5 dla modelu Wienera z następnikami liniowymi i nieliniowymi są niemalże identyczne, a ewentualne różnice są pomijalne. Porównując otrzymane rezultaty z tymi otrzymanymi w przypadku modelu Hammersteina okazuje się, że model Wienera jest nieco lepszy, co może wskazywać na większe nieliniowości na wyjściu procesu.

| Nr sekwencji | Model liniowy | Model Wienera | |
|--------------|---------------|--------------------|-----------------------|
| | | Następniki liniowe | Następniki nieliniowe |
| 1. | 4,062 | 0,314 | 0,311 |
| 2. | 2,701 | 0,173 | 0,175 |
| 3. | 1,283 | 0,163 | 0,157 |
| 4. | 3,893 | 0,410 | 0,409 |
| 5. | 0,600 | 0,044 | 0,035 |

Tab. 4.5: Porównanie modeli.

Udało się osiągnąć zamierzony efekt - mniejsza liczba reguł, bez utraty dokładności.

4.4. Regulacja predykcyjna

Współczesne systemy sterowania wymagają metod, które potrafią skutecznie przewidywać i reagować na zmieniające się warunki pracy obiektów. Regulacja predykcyjna stanowi jedno z najbardziej zaawansowanych podejść do sterowania, pozwalając na optymalne korygowanie trajektorii sterowania w czasie rzeczywistym. Dzięki wykorzystaniu modeli matematycznych układu oraz przewidywaniu jego przyszłych stanów, MPC, a w szczególności DMC, umożliwia uwzględnienie ograniczeń oraz minimalizację błędów sterowania.

W kontekście systemów nieliniowych szczególnie przydatne okazują się modele Hammersteina i Wienera, które rozdzielając nieliniową statykę od liniowej dynamiki układu ułatwiają implementację algorytmów predykcyjnych, ponieważ liniowa część modelu może być efektywnie analizowana i przewidywana przy użyciu klasycznych technik optymalizacyjnych, podczas gdy nieliniowa statyka może być traktowana jako dodatkowa warstwa przekształcająca sygnały sterowania lub wyjścia. Dzięki temu możliwe jest skuteczne sterowanie szeroką gamą obiektów przemysłowych, od procesów chemicznych po nowoczesne systemy robotyczne i autonomiczne. W niniejszym rozdziale zostaną omówione kluczowe zasady działania regulacji predykcyjnej oraz sposoby jej zastosowania w połączeniu z modelami Hammersteina i Wienera.

4.4.1. Regulator DMC

Tu coś o DMC, może jakiś schemacik blokowy

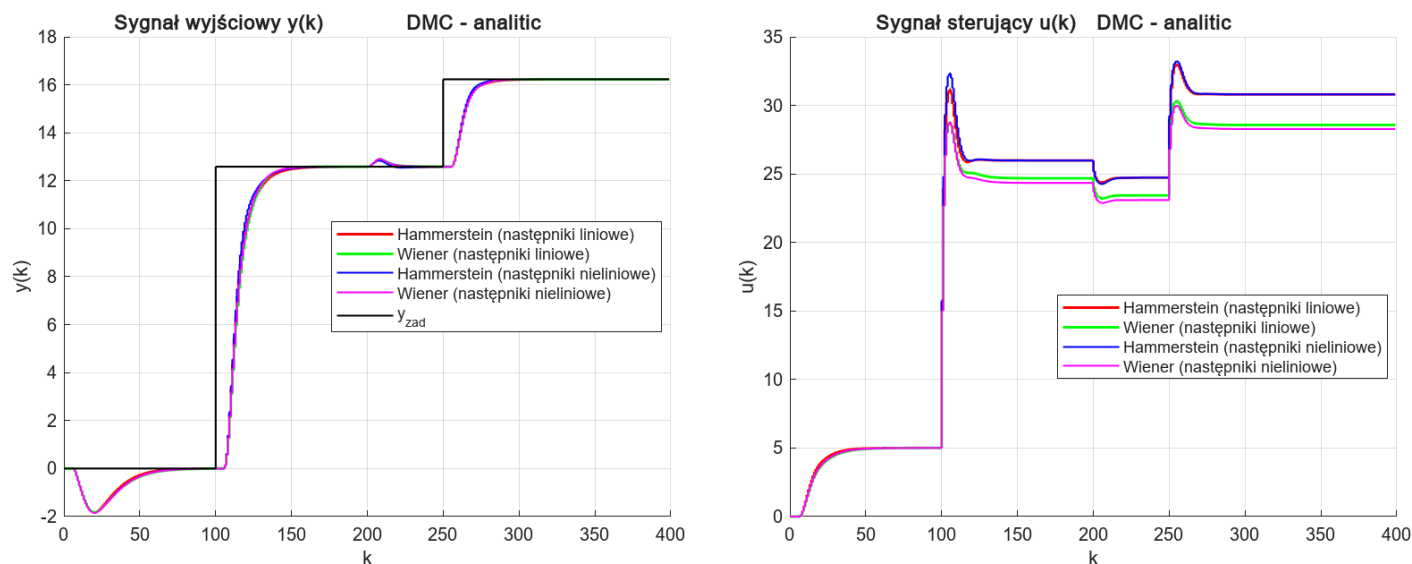
4.4.2. Porównanie

| Regulator | Model Hammersteina | | Model Wienera | |
|--------------|--------------------|-----------------------|--------------------|-----------------------|
| | Następniki liniowe | Następniki nieliniowe | Następniki liniowe | Następniki nieliniowe |
| DMC-analitic | 20,6110 | 20,4929 | 22,3904 | 22,4372 |
| DMC-numeric | 20,6110 | 20,4929 | 22,3903 | 22,4372 |
| DMC-SL | 20,5775 | 20,4629 | 22,3721 | 22,4176 |
| DMC-NPL | 20,4963 | 20,3846 | 22,3008 | 22,3467 |
| FDMC | 19,7474 | 19,6734 | 21,6466 | 21,7162 |

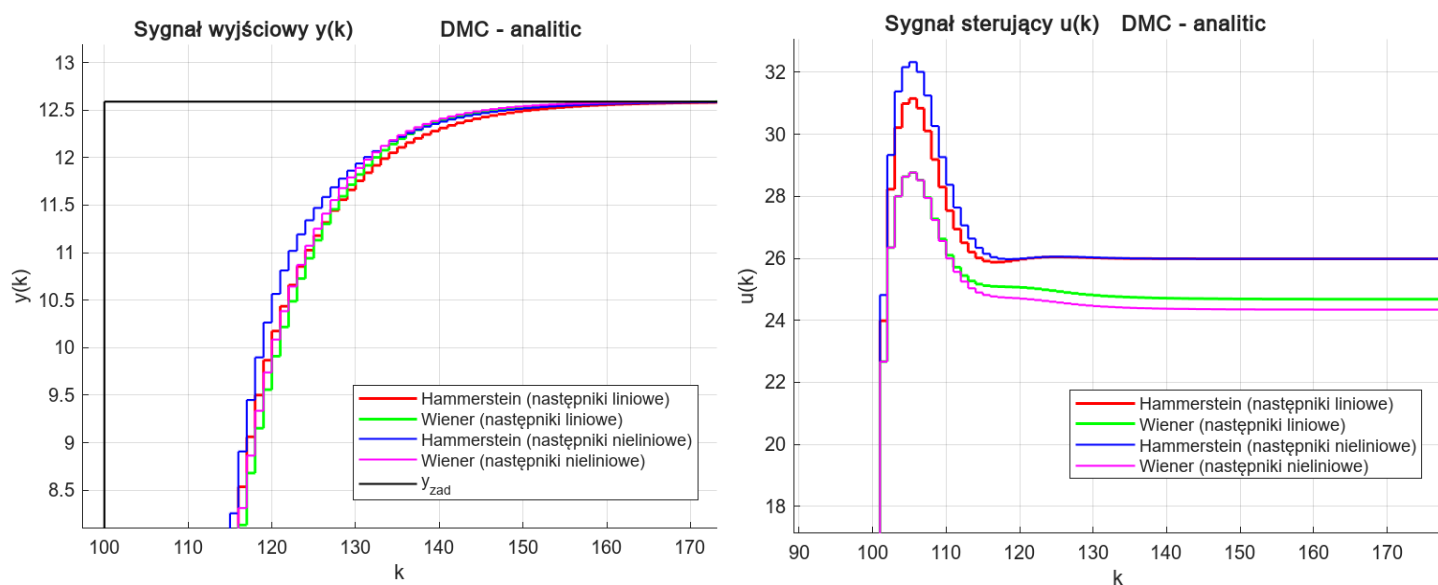
Tab. 4.6: Porównanie modeli.

| Regulator | Model Hammersteina | | | | Model Wienera | | | |
|--------------|--------------------|----------------|-----------------------|----------------|--------------------|----------------|-----------------------|----------------|
| | Następniki liniowe | | Następniki nieliniowe | | Następniki liniowe | | Następniki nieliniowe | |
| | E _y | E _u | E _y | E _u | E _y | E _u | E _y | E _u |
| DMC-analitic | 19,6366 | 0,9743 | 19,4785 | 1,0144 | 21,3817 | 1,0087 | 21,428 | 1,0092 |
| DMC-numeric | 19,6366 | 0,9743 | 19,4785 | 1,0144 | 21,3817 | 1,0087 | 21,428 | 1,0092 |
| DMC-SL | 19,5959 | 0,9816 | 19,4414 | 1,0215 | 21,3607 | 1,0114 | 21,4057 | 1,0119 |
| DMC-NPL | 19,527 | 0,9693 | 19,3757 | 1,0089 | 21,3023 | 0,998 52 | 21,3476 | 0,999 06 |
| FDMC | 18,2132 | 1,5342 | 18,1112 | 1,5622 | 20,0199 | 1,6267 | 20,0874 | 1,6288 |

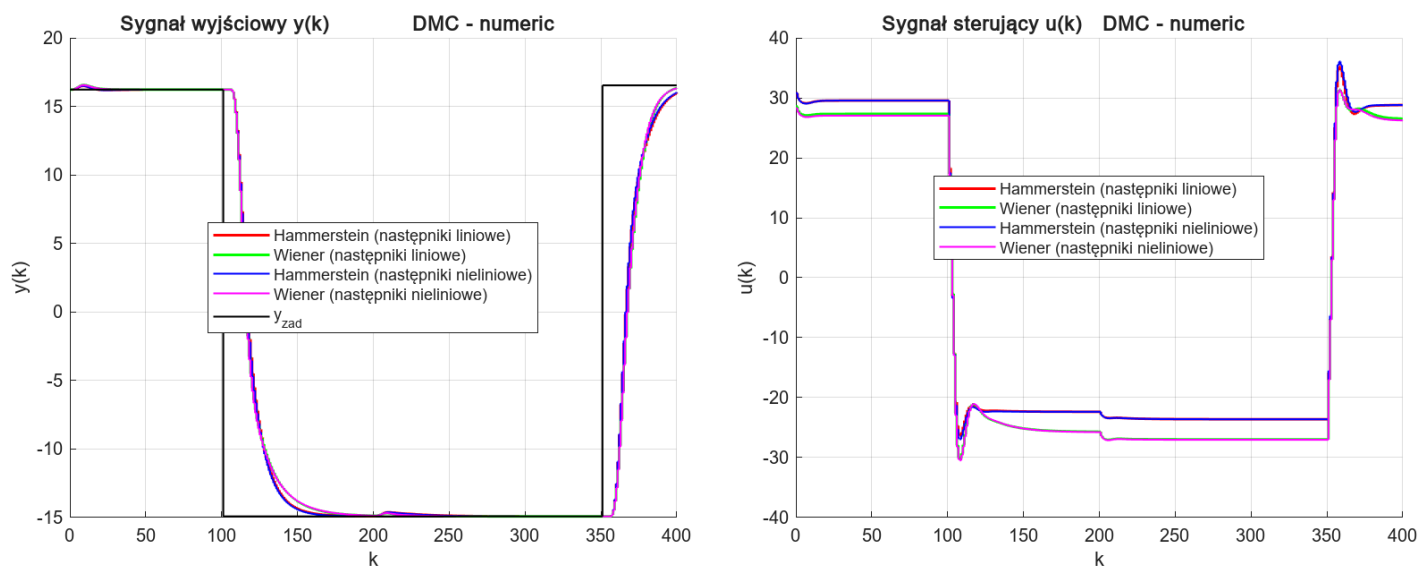
Tab. 4.7: Porównanie modeli.



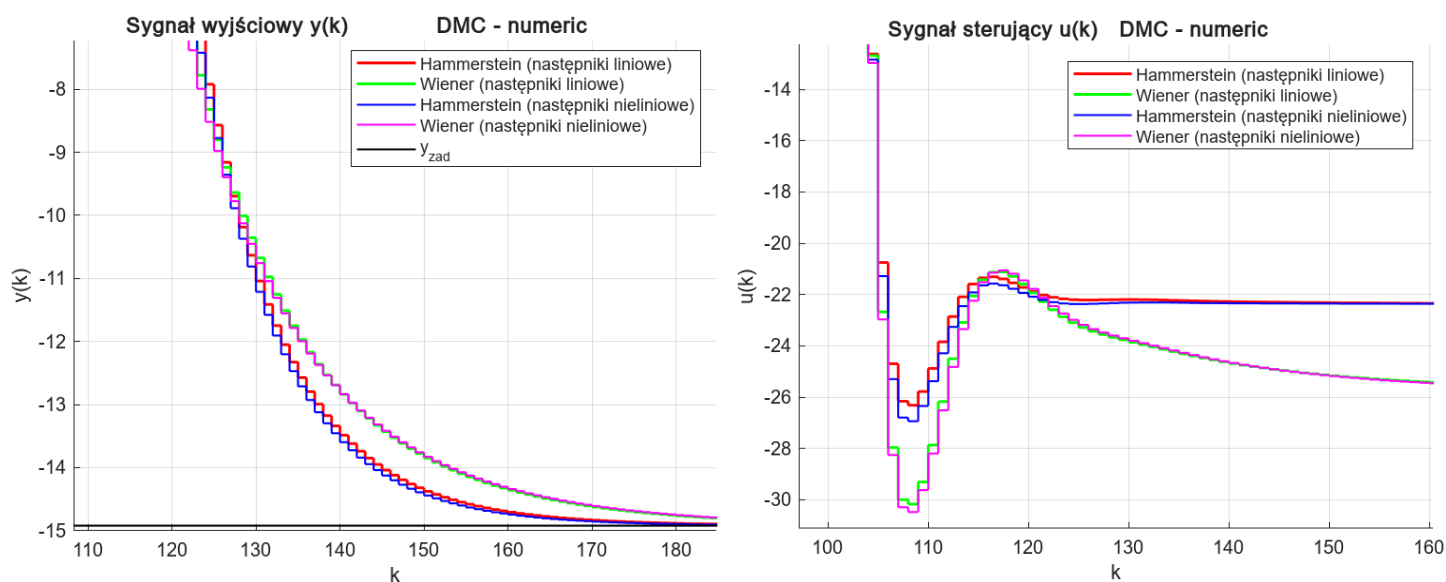
Rys. 4.23: Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli.



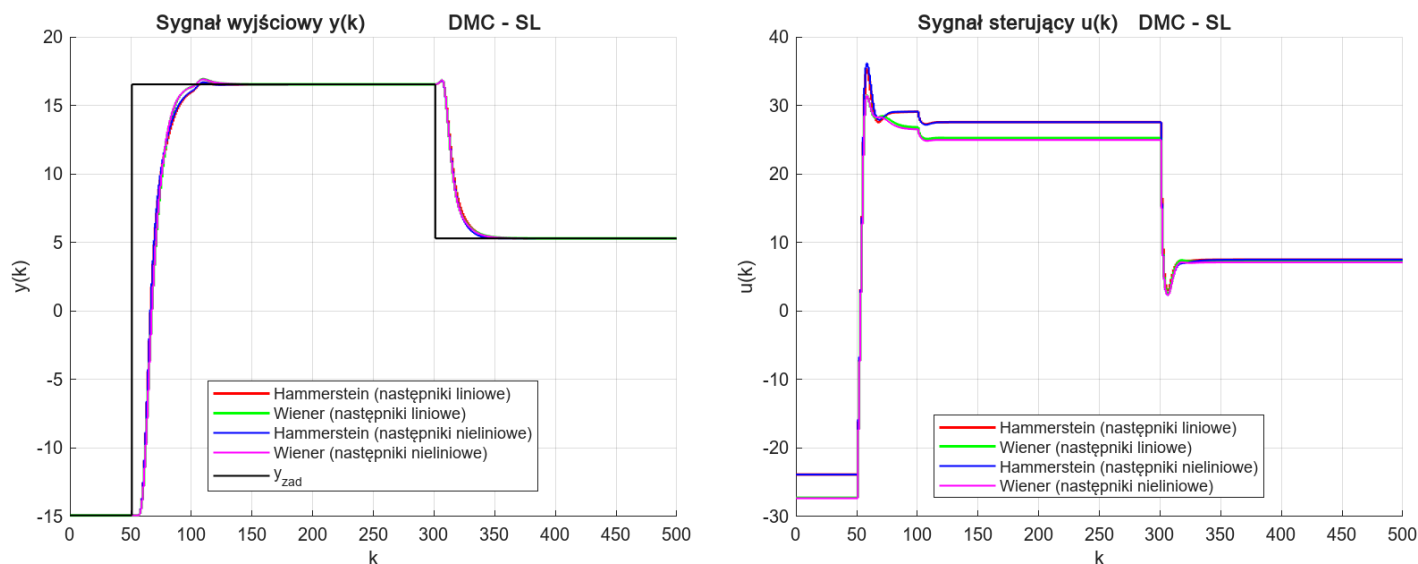
Rys. 4.24: Wyszczególnienie interesujących fragmentów przebiegów.



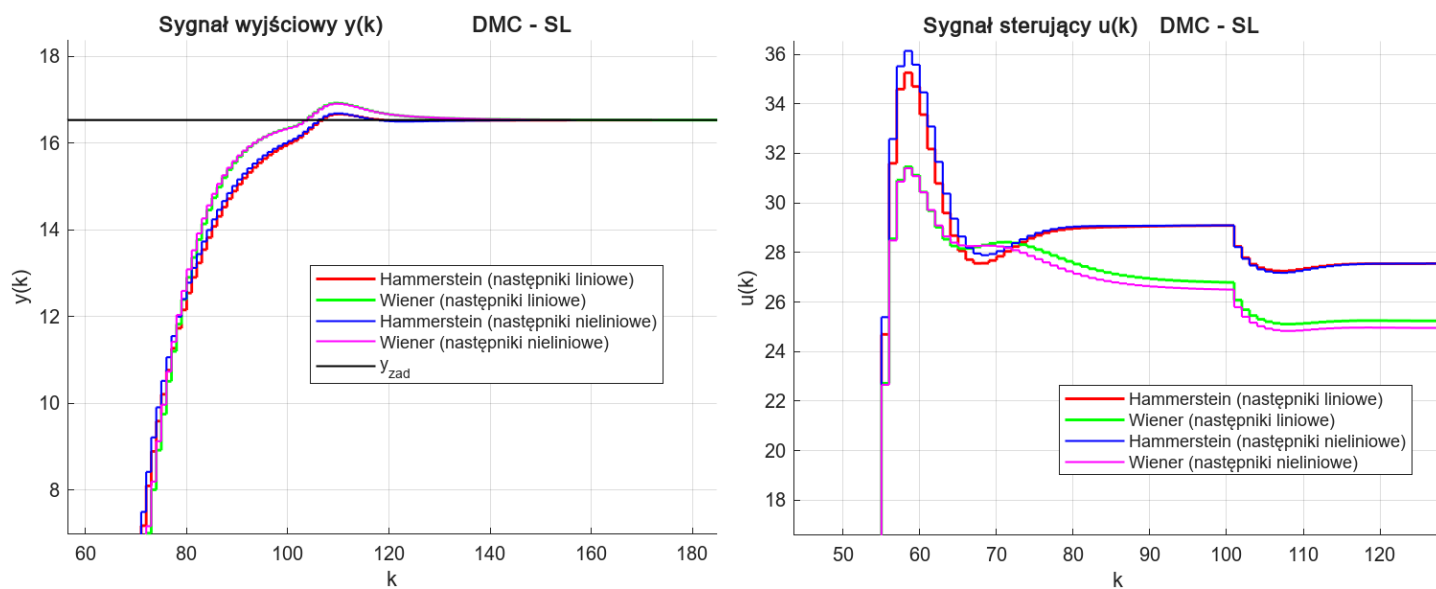
Rys. 4.25: Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli.



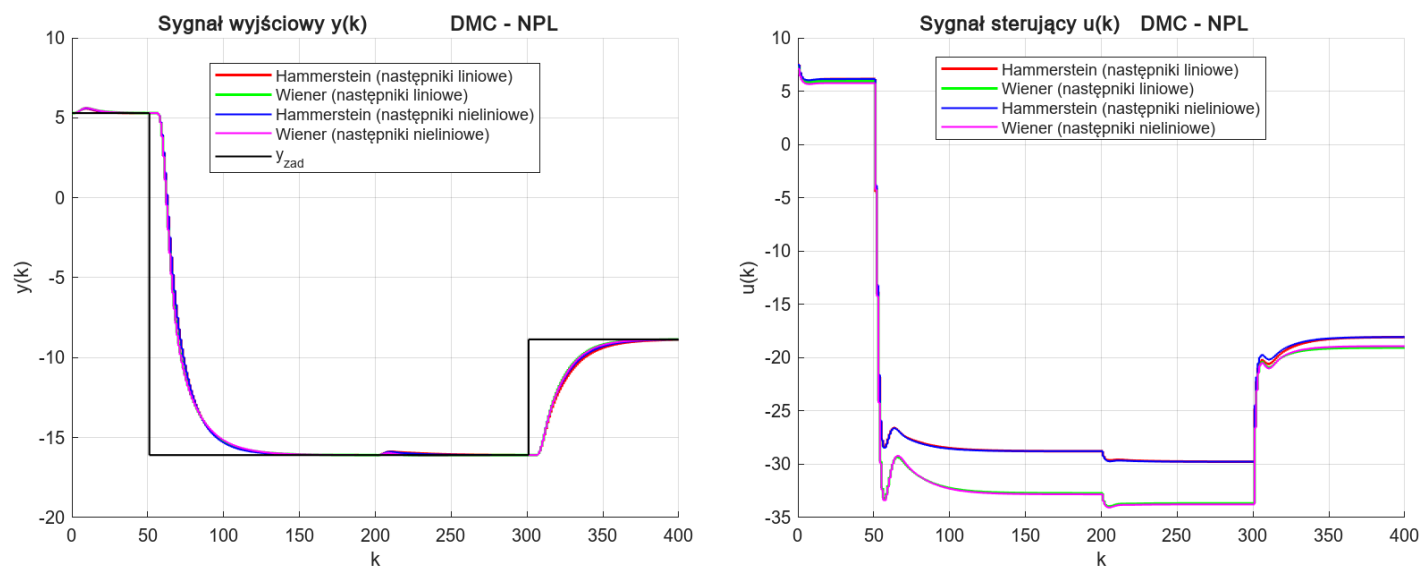
Rys. 4.26: Wyszczególnienie interesujących fragmentów przebiegów.



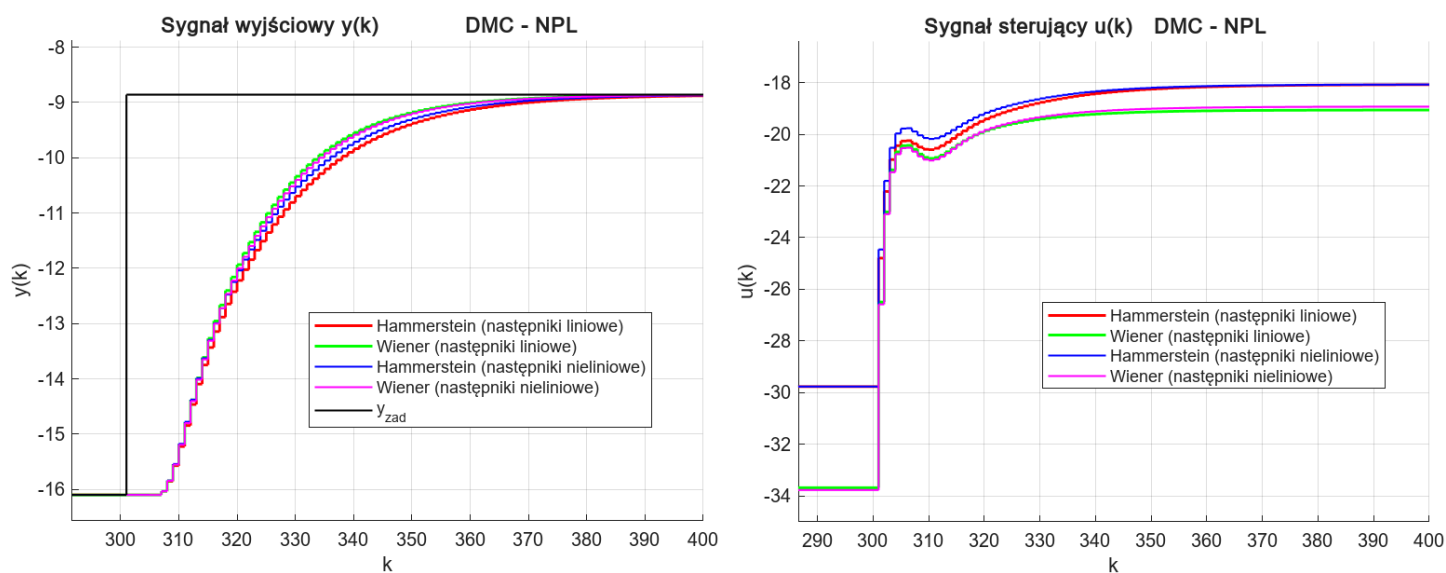
Rys. 4.27: Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli.



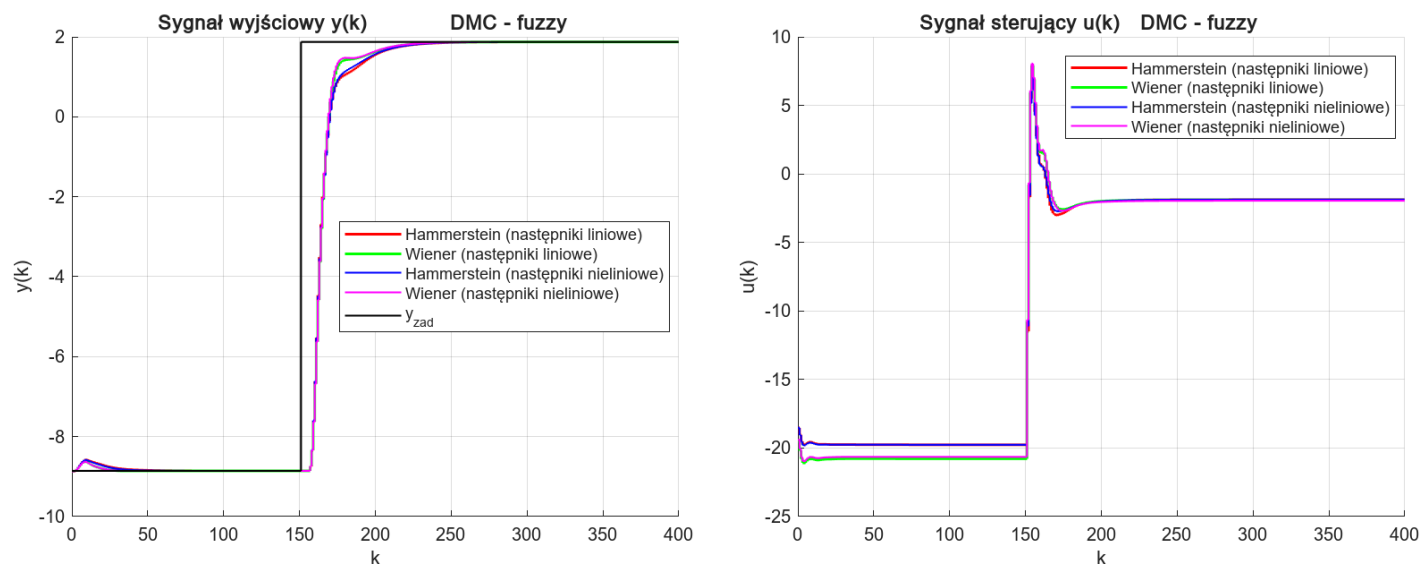
Rys. 4.28: Wyszczególnienie interesujących fragmentów przebiegów.



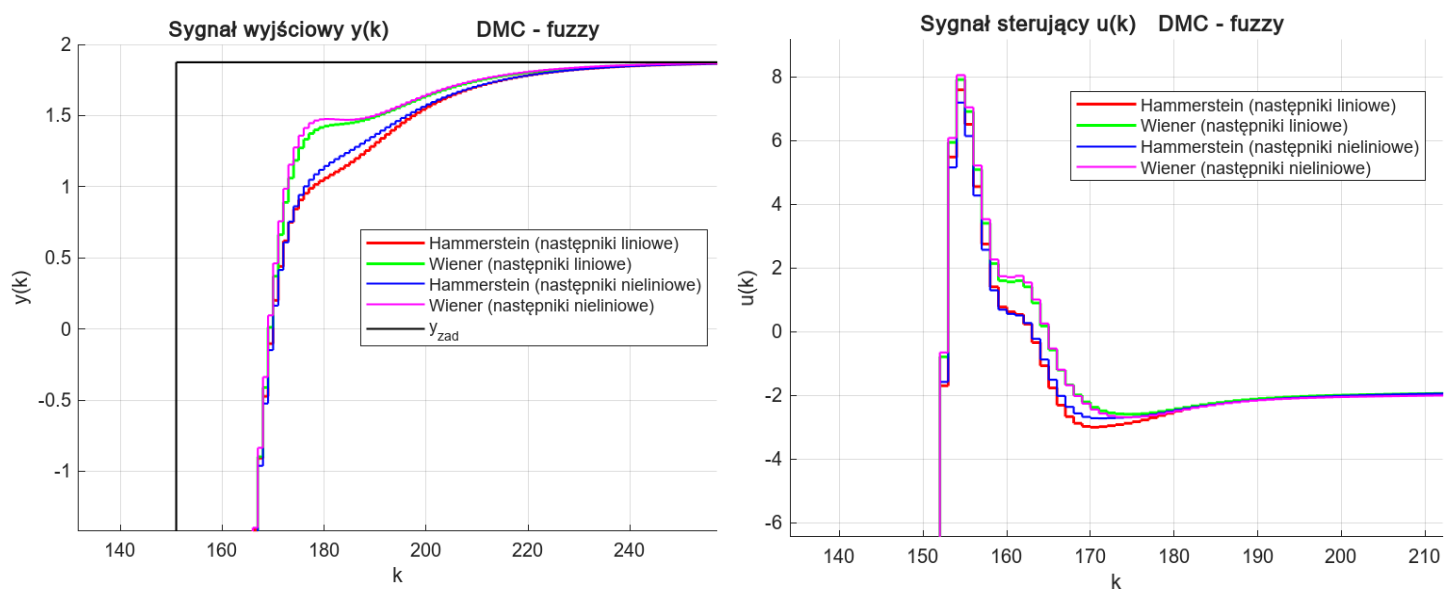
Rys. 4.29: Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli.



Rys. 4.30: Wyszczególnienie interesujących fragmentów przebiegów.



Rys. 4.31: Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli.



Rys. 4.32: Wyszczególnienie interesujących fragmentów przebiegów.

4.5. Podsumowanie

Bibliografia

- [1] A. Janczak. *Identification of Nonlinear Systems Using Neural Networks and Polynomial Models*. Zielona Góra, Springer, 2005.
- [2] L. Ljung. *System Identification. Theory for the User*, wyd. 2. Szwecja, Prentice Hall, 1999.
- [3] L. Ljung. *Above: A Harrunerstein model. Below: A Wiener model*, [ilustracja].
W: *System Identification. Theory for the User*, wyd. 2. Szwecja, Prentice Hall, 1999.
- [4] M. Ławryńczuk. *Modelowanie i identyfikacja*. Warszawa, 2022.
- [5] M. Ławryńczuk. *Sterowanie procesów ciągłych*. Warszawa, 2022.
- [6] M. Ławryńczuk, P. Marusak. *Sztuczna inteligencja w automatyce*. Warszawa, 2009–2018.
- [7] M. Ławryńczuk, P. Marusak. *Regulator typu Takagi–Sugeno otrzymany za pomocą podejścia PDC*, [ilustracja]. W: *Sztuczna inteligencja w automatyce*. Warszawa, 2009–2018.
- [8] J. M. Maciejowski. *Predictive Control with Constraints*. Harlow, Prentice Hall, 2002.
- [9] K. Malinowski, P. Tatjewski. *Podstawy Automatyki*, wyd. 2. Warszawa, 2016.
- [10] P. Marusak. *Regulacja predykcyjna obiektów nieliniowych z zastosowaniem techniki DMC i modelowania rozmytego*. Warszawa, Wydział Elektroniki i Technik Informacyjnych, 2002.
- [11] P. Marusak, J. Pułaczewski, P. Tatjewski. *Algorytmy DMC z uwzględnieniem ograniczeń sterowania*, vol. 1. Opole, 1999.
- [12] P. Marusak, P. Tatjewski. *Fuzzy Dynamic Matrix Control algorithms for nonlinear plants*, vol. 2. Międzyzdroje, 2000.
- [13] K. Mehra. *Takagi-Sugeno Fuzzy Modeling for Process Control*. Newcastle, 2008.
- [14] H. Moodi, M. Farrokhi. *Robust observer-based controller design for Takagi–Sugeno systems with nonlinear consequent parts*. W: *Fuzzy Sets and Systems*. Amsterdam, Elsevier B.V., 2015.
Nr 273, s. 141-154, ISSN 0165-0114.
- [15] K. Rykaczewski. *Systemy rozmyte i ich zastosowania*. Toruń, 2006.
- [16] P. Tatjewski. *Sterowanie zaawansowane obiektów przemysłowych. Struktury i algorytmy*.
Warszawa, Akademicka Oficyna Wydawnicza EXIT, 2016.
- [17] P. Tatjewski. *Warstwowa struktura sterowania obiektem przemysłowym*, [ilustracja].
W: *Sterowanie zaawansowane obiektów przemysłowych. Struktury i algorytmy*.
Warszawa, Akademicka Oficyna Wydawnicza EXIT, 2016.
- [18] P. Tatjewski. *Zasada regulacji predykcyjnej*, [ilustracja].
W: *Sterowanie zaawansowane obiektów przemysłowych. Struktury i algorytmy*.
Warszawa, Akademicka Oficyna Wydawnicza EXIT, 2016.

-
- [19] P. Tatjewski. *Przykład odpowiedzi wyjścia obiektu y na skok sterowania u* , [ilustracja].
W: *Sterowanie zaawansowane obiektów przemysłowych. Struktury i algorytmy*.
Warszawa, Akademicka Oficyna Wydawnicza EXIT, 2016.
- [20] A. Piegat. *Modelowanie i sterowanie rozmyte*.
Warszawa, Akademicka Oficyna Wydawnicza EXIT, 1999.

Spis rysunków

| | | |
|------|--|----|
| 3.1 | Warstwowa struktura sterowania [17]. | 5 |
| 3.2 | Zasada działania regulacji predykcyjnej [18]. | 6 |
| 3.3 | Odpowiedź skokowa [19]. | 7 |
| 3.4 | Funkcja przynależności zbioru ostrego. | 11 |
| 3.5 | Funkcja przynależności zbioru rozmytego. | 11 |
| 3.6 | Przykładowe zbiory rozmyte. | 12 |
| 3.7 | Struktura regulatora rozmytego otrzymanego podejściem PDC [7]. | 14 |
| 3.8 | Model Hammersteina [3]. | 15 |
| 3.9 | Model Wienera [3]. | 15 |
| 4.1 | Obiekt regulacji automatycznej. | 17 |
| 4.2 | Charakterystyka statyczna. | 18 |
| 4.3 | Porównanie odpowiedzi obiektu nieliniowego i zlinearyzowanego w punkcie pracy na wymuszenia skokowe sygnału F_1 | 19 |
| 4.4 | Porównanie odpowiedzi obiektu nieliniowego i zlinearyzowanego w punkcie pracy na wymuszenia skokowe sygnału F_D | 20 |
| 4.5 | Reprezentacja graficzna modelu Hammersteina. | 21 |
| 4.6 | Zbiory rozmyte - następniiki liniowe. | 22 |
| 4.7 | Zbiory rozmyte - następniiki nieliniowe. | 23 |
| 4.8 | Wykres funkcji $\sinh()$ | 24 |
| 4.9 | Porównanie modelu Hammersteina z następniikami liniowymi i nieliniowymi - pierwsza sekwencja. | 25 |
| 4.10 | Porównanie modelu Hammersteina z następniikami liniowymi i nieliniowymi - druga sekwencja. | 26 |
| 4.11 | Porównanie modelu Hammersteina z następniikami liniowymi i nieliniowymi - trzecia sekwencja. | 27 |
| 4.12 | Porównanie modelu Hammersteina z następniikami liniowymi i nieliniowymi - czwarta sekwencja. | 28 |
| 4.13 | Porównanie modelu Hammersteina z następniikami liniowymi i nieliniowymi - piąta sekwencja. | 29 |
| 4.14 | Reprezentacja graficzna modelu Wienera. | 31 |
| 4.15 | Zbiory rozmyte - następniiki liniowe. | 32 |
| 4.16 | Zbiory rozmyte - następniiki nieliniowe. | 32 |
| 4.17 | Porównanie modelu Wienera z następniikami liniowymi i nieliniowymi - pierwsza sekwencja. | 33 |
| 4.18 | Porównanie modelu Wienera z następniikami liniowymi i nieliniowymi - druga sekwencja. | 34 |
| 4.19 | Porównanie modelu Wienera z następniikami liniowymi i nieliniowymi - trzecia sekwencja. | 35 |
| 4.20 | Porównanie modelu Wienera z następniikami liniowymi i nieliniowymi - czwarta sekwencja. | 36 |
| 4.21 | Porównanie modelu Wienera z następniikami liniowymi i nieliniowymi - piąta sekw encja. | 37 |
| 4.22 | Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli. | 41 |
| 4.23 | Wyszczególnienie interesujących fragmentów przebiegów. | 41 |
| 4.24 | Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli. | 42 |
| 4.25 | Wyszczególnienie interesujących fragmentów przebiegów. | 42 |
| 4.26 | Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli. | 43 |
| 4.27 | Wyszczególnienie interesujących fragmentów przebiegów. | 43 |
| 4.28 | Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli. | 44 |
| 4.29 | Wyszczególnienie interesujących fragmentów przebiegów. | 44 |
| 4.30 | Porównanie sygnału wyjściowego oraz sterującego poszczególnych modeli. | 45 |
| 4.31 | Wyszczególnienie interesujących fragmentów przebiegów. | 45 |

Spis tabel

| | | |
|-----|---|----|
| 4.1 | Wartość błędu średnio kwadratowego w zależności od wielkości wymuszenia F_1 | 19 |
| 4.2 | Wartość błędu średnio kwadratowego w zależności od wielkości wymuszenia F_D | 20 |
| 4.3 | Współczynniki hiperbolicznych następników reguł. | 24 |
| 4.4 | Porównanie modeli. | 30 |
| 4.5 | Porównanie modeli. | 38 |
| 4.6 | Porównanie modeli. | 40 |
| 4.7 | Porównanie modeli. | 40 |