

Wydział Elektroniki i Technik Informacyjnych
Politechnika Warszawska

Pracownia dyplomowa magisterska
(semestr zimowy 24/25Z)

Przegląd literatury

Autor:
Wojciech Rogalski

Promotor:
dr hab. inż. Piotr Marusak

Warszawa, 2025

Spis treści

1. Wstęp	3
2. Algorytmy regulacji predykcyjnej	4
2.1. DMC	6
2.1.1. Algorytm DMC w wersji analitycznej	7
2.1.2. Algorytm DMC w wersji numerycznej	8
2.2. Inne warianty DMC	9
3. Regulacja rozmyta	10
3.1. Modele Takagi-Sugeno	12
3.2. Podejście PDC	13
4. Modele Hammersteina i Wienera	14
Bibliografia	15
Spis rysunków	17

1. Wstęp

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]:

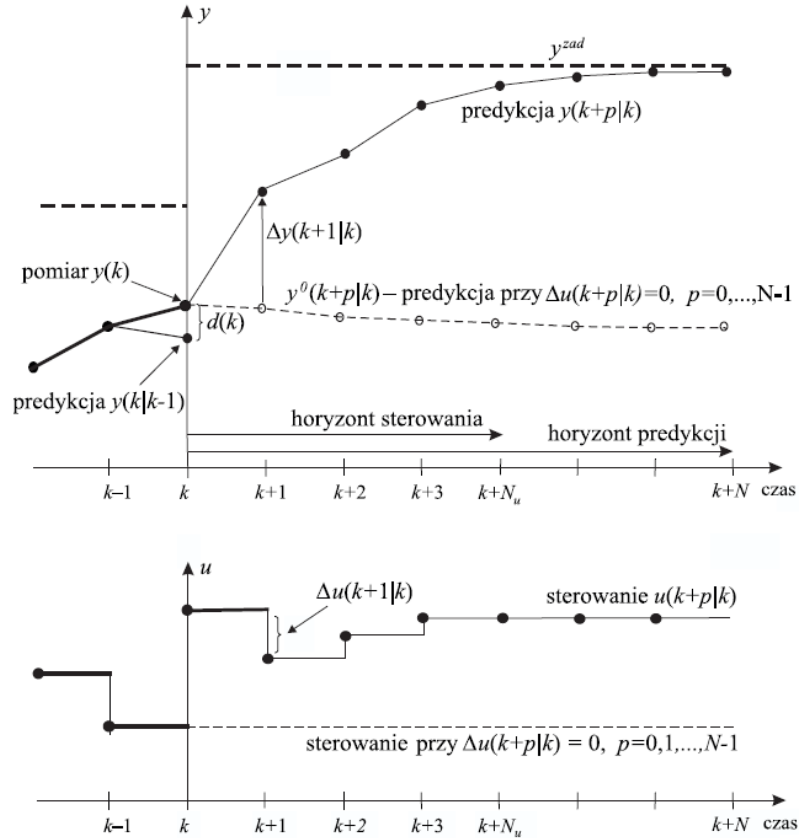
- możliwość uwzględnienia ograniczeń sygnałów zarówno wejściowych, jak i wyjściowych
- dobra jakość regulacji także w przypadku obiektów o trudnej dynamice - duże opóźnienia
- efektywne regulowanie procesu wielowymiarowego
- możliwość zaprojektowania nieliniowego algorytmu regulacji predykcyjnej
- możliwość 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).

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. 2.2



Rys. 2.2: 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 predykcji¹
- $\Delta y(k+p|k)$ - 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 predykcji 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:

$$y(k+p|k) = y^0(k+p|k) + \Delta y(k+p|k) \quad p = 1 \dots N \quad (2.3)$$

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

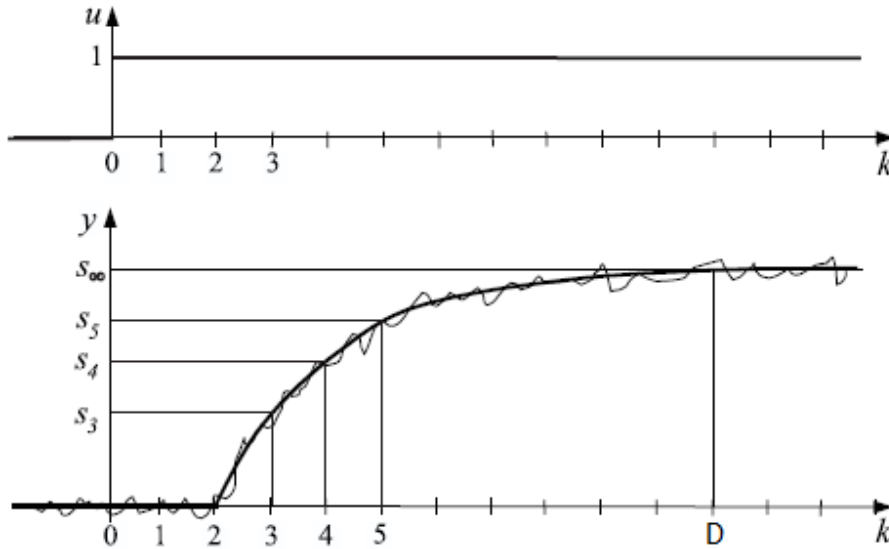
Ze względu na praktyczną naturę algorytmu DMC do dalszych rozważań postanowiono przyjąć właśnie ten algorytm.

2.1. DMC

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

$$s_k = s_{k+1} = s_{k+2} = \dots = s_D = s_\infty \quad (2.4)$$



Rys. 2.3: Odpowiedź skokowa [19].

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

$$y^{mod}(k + p|k) = y^{mod}(0) + \sum_{j=1}^{k+p} s(j) \Delta u(k + p - j) \quad (2.5)$$

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

$$\hat{y}(k+p|k) = y^{mod}(k+p|k) + d(k+p|k) \quad (2.6)$$

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

$$d(k) = y(k) - y^{mod}(k|k-1) \quad (2.7)$$

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

$$d(k+1|k) = d(k+2|k) = \dots = d(k+N|k) = d(k) \quad (2.8)$$

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

$$\begin{cases} \hat{y}(k+p|k) = y^0(k+p|k) + \Delta y(k+p|k) \\ y^0(k+p|k) = y(k) + \sum_{j=1}^k (s(j+p) - s(j)) \Delta u(k-j) \\ \Delta y(k+p|k) = \sum_{j=1}^p s(j) \Delta u(k+p-j) \end{cases} \quad (2.9)$$

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.

2.1.1. Algorytm DMC w wersji analitycznej

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

$$Y^{zad} = \begin{bmatrix} y^{zad}(k+1|k) \\ \vdots \\ y^{zad}(k+N|k) \end{bmatrix}_{N \times 1}, \quad \hat{Y} = \begin{bmatrix} \hat{y}(k+1|k) \\ \vdots \\ \hat{y}(k+N|k) \end{bmatrix}_{N \times 1}, \quad \Delta U(k) = \begin{bmatrix} \Delta u(k|k) \\ \vdots \\ \Delta u(k+N_u-1|k) \end{bmatrix}_{N_u \times 1} \quad (2.10)$$

Oraz macierzy wagowych:

$$\Psi = \begin{bmatrix} \psi_1 & & \\ & \ddots & \\ & & \psi_N \end{bmatrix}_{N \times N}, \quad \Lambda = \begin{bmatrix} \lambda_0 & & \\ & \ddots & \\ & & \lambda_{N_u-1} \end{bmatrix}_{N_u \times N_u} \quad (2.11)$$

Tak zdefiniowane wektory pozwalają zapisać funkcję celu w postaci:

$$J(k) = \min_{\Delta U(k)} \{ \|Y^{zad} - \hat{Y}(k)\|_{\Psi}^2 + \|\Delta U(k)\|_{\Lambda}^2 \} \quad (2.12)$$

Odwolując się do równania 2.9 prognozowaną trajektorię wyjść w wersji macierzowo-wektorowej można zapisać w postaci:

$$\begin{cases} \hat{Y} = Y^0(k) + \Delta Y(k) \\ Y^0(k) = Y(k) + M^P \Delta U^P(k) \\ \Delta Y(k) = M \Delta U(k) \end{cases} \quad (2.13)$$

gdzie:

$$M^P = \begin{bmatrix} s_2 - s_1 & s_3 - s_2 & \cdots & s_D - s_{D-1} \\ s_3 - s_1 & s_4 - s_2 & \cdots & s_D - s_{D-1} \\ \vdots & \vdots & \ddots & \vdots \\ s_{N+1} - s_1 & s_{N+2} - s_2 & \cdots & s_{N+D-1} - s_{D-1} \end{bmatrix}_{Nx(D-1)} \quad (2.14)$$

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

Wektor $\Delta U^P(k)$ opisuje wartości przyrostów sterowań z poprzednich chwil, natomiast macierz M nazywana jest macierzą dynamiczną. Uwzględniając postać funkcji kryterialnej (2.12) oraz przyjmując odpowiednie założenia $\Psi \geq 0$ oraz $\Lambda \geq 0$ wektor kolejnych przerostów sterowań dany jest wzorem:

$$\Delta U(k) = K(Y^{zad} - Y^0) \quad (2.15)$$

gdzie

$$K = (M^T \Psi M + \Lambda)^{-1} M^T \Psi \quad (2.16)$$

Zważywszy na fakt, że w każdej iteracji algorytmu wykorzystywany jest tylko pierwsza z wyznaczonych wartości przyrostów sterowań, równanie 2.15 można uprościć i zapisać w postaci:

$$\Delta u(k) = \bar{K}_1(Y^{zad} - Y^0(k)) \quad (2.17)$$

gdzie \bar{K}_1 jest pierwszym wierszem macierzy K . Równanie 2.17 opisuje analityczną postać algorytmu DMC. Kolejnym istotnym aspektem jest rzutowanie wyznaczanych wartości na ograniczenia:

$$\begin{aligned} u^{min} &\leq u(k+1|k) \leq u^{max} \\ -\Delta u^{max} &\leq \Delta u(k|k) \leq \Delta u^{max} \\ y^{min} &\leq \hat{y}(k|k) \leq y^{max} \end{aligned} \quad (2.18)$$

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.

2.1.2. 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:

$$\min\{J(x) = \frac{1}{2}x^T Hx + f^T x\} \quad (2.19)$$

przy ograniczeniach:

$$\begin{aligned} x_{min} &\leq x \leq x_{max} \\ Ax &\leq b \end{aligned} \quad (2.20)$$

gdzie:

$$\begin{aligned}
 x &= \Delta U(k), \quad x_{min} = -\Delta U_{max}, \quad x_{max} = \Delta U_{max} \\
 H &= 2(M^T \Psi M + \Lambda) \\
 f &= -2M^T \Psi (Y^{zad}(k) - Y^0(k)) \\
 A &= \begin{bmatrix} -J \\ J \\ -M \\ M \end{bmatrix}, \quad b = \begin{bmatrix} -U_{min} + U(k-1) \\ U_{max} - U(k-1) \\ -Y_{min} + Y^0(k) \\ Y_{max} - Y^0(k) \end{bmatrix} \\
 J &= \mathbb{I}_{N_u \times N_u}, \quad Y_{min} = [y_{min}, \dots, y_{min}]^T, \quad Y_{max} = [y_{max}, \dots, y_{max}]^T
 \end{aligned} \tag{2.21}$$

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.

2.2. 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].

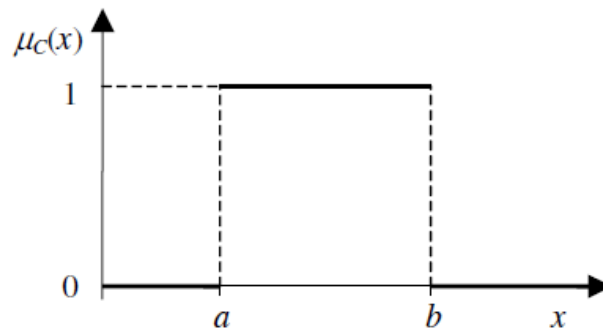
DMC-NO

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. Regulacja rozmyta

Tytułowy typ regulacji zrywa ze standardową procedurą podejmowania decyzji w sposób binarny, tzn. 1/0 [10]. Na Rys. 3.1 przedstawiono standardową funkcję przynależności do zbioru, którą można opisać wzorem:

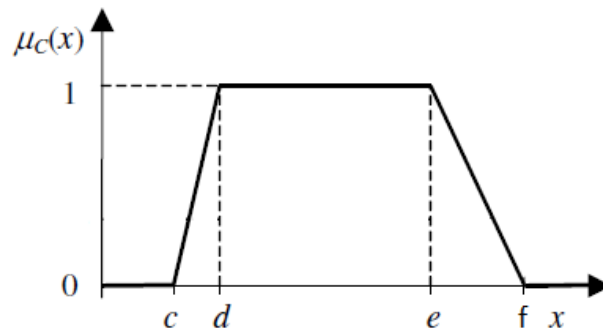
$$\mu_C = \begin{cases} 1, & x \in [a, b] \\ 0, & x \notin [a, b] \end{cases} \quad (3.1)$$



Rys. 3.1: Funkcja przynależności zbioru ostrego.

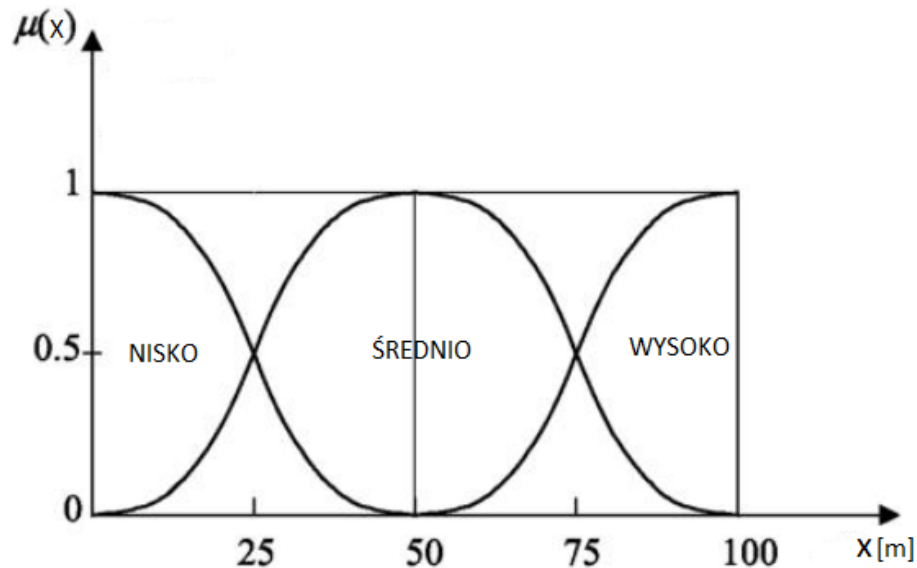
Natomiast istota wnioskowania rozmytego polega na zgromadzeniu wiedzy eksperckiej, operatora procesu, wspartej odpowiedzią skokową obiektu, które pozwolą na sformułowanie zbiorów rozmytych (Rys. 3.3) oraz bazy reguł definiującej regulator rozmyty [5, 6]. Funkcja przynależności zbioru rozmytego została zaprezentowana na Rys. 3.2 i w tym przypadku dana jest wzorem:

$$\mu_C = \begin{cases} 0, & x \leq c \vee x \geq f \\ \frac{x-c}{d-c}, & c \leq x \leq d \\ 1, & d \leq x \leq e \\ \frac{f-x}{f-e}, & e \leq x \leq f \end{cases} \quad (3.2)$$



Rys. 3.2: 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, sigmoidealne.



Rys. 3.3: Przykładowe zbiory rozmyte.

Ważną kwestią w rozumieniu systemów rozmytych, jest pośrednik między zmienną numeryczną, a zmienną symboliczną - zmienna lingwistyczna. Na Rys. 3.3 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:

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } < \text{nastpnik} > \quad (3.3)$$

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

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y = y_a \quad (3.4)$$

2. Następnik rozmyty

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y \text{ jest } Y_a \quad (3.5)$$

3. Następnik funkcyjny

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y = f(x_1, x_2, \dots, x_k) \quad (3.6)$$

Szczególnie istotne oraz wykorzystywane w praktyce są dwie ostatnie wymienione metody. Następniiki 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.

3.1. 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 3.7.

$$y = \frac{\sum_{j=1}^r w^j y^j}{\sum_{j=1}^r w^j} \quad (3.7)$$

gdzie:

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

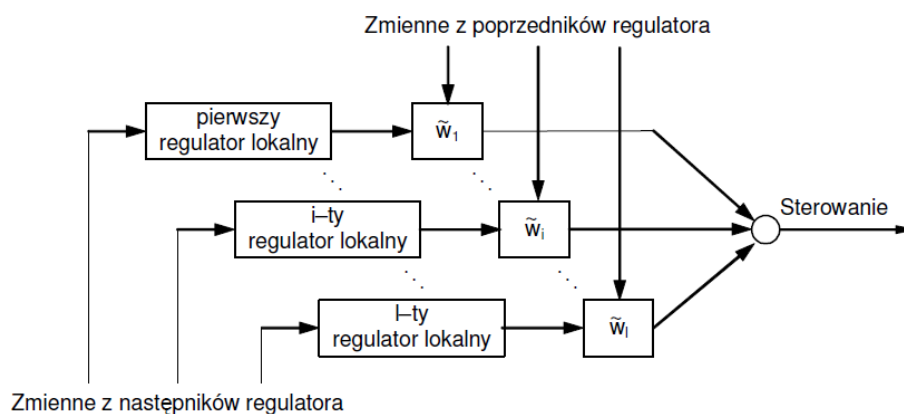
Ogólnie modele rozmyte służą do aproksymacji funkcji nieliniowych stąd w przypadku modeli Takagi-Sugeno następniiki funkcyjne występują najczęściej w postaci funkcji wielomianowych pierwszego rzędu [5, 6]:

$$\text{JEŚLI } < \text{poprzednik} > \text{ TO } y = a_0 + \sum_{j=1}^n a_j x_j \quad (3.8)$$

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.

3.2. Podejście PDC

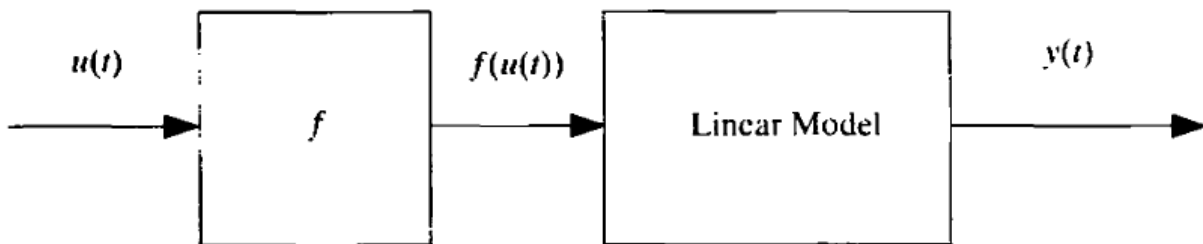
Podejście PDC (*Parallel Distributed Compensation*) zakłada dedukcję regulatora za pomocą modelu Takagi-Sugeno obiektu. Istota równoległej kompensacji rozproszonej polega na dobraniu dla każdego następnika rozmytego lokalnego regulatora liniowego. Zatem w wyniku takiego podejścia otrzymuje się tyle regulatorów 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. 3.4.



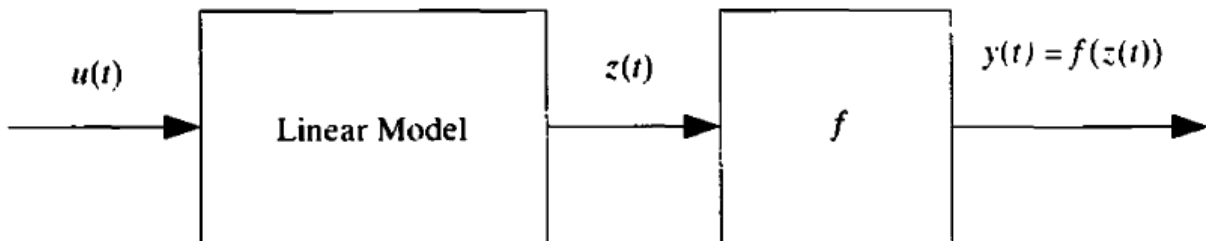
Rys. 3.4: Struktura regulatora rozmytego otrzymanego podejściem PDC [7].

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. 4.1) bądź wyjściu (Rys. 4.2) [1].



Rys. 4.1: Model Hammersteina [3].



Rys. 4.2: 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].

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. Mehran. *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

2.1	Warstwowa struktura sterowania [17].	4
2.2	Zasada działania regulacji predykcyjnej [18].	5
2.3	Odpowiedź skokowa [19].	6
3.1	Funkcja przynależności zbioru ostrego.	10
3.2	Funkcja przynależności zbioru rozmytego.	10
3.3	Przykładowe zbiory rozmyte.	11
3.4	Struktura regulatora rozmytego otrzymanego podejściem PDC [7].	13
4.1	Model Hammersteina [3].	14
4.2	Model Wienera [3].	14