

OLESIAK Krzysztof ¹

PROJEKTOWANIE I STROJENIE REGULATORÓW ROZMYTYCH TYPU MAMDANIEGO

W artykule przedstawiono podstawowe zasady projektowania regulatorów rozmytych typu Mamdaniego. Zaprezentowano ogólną formułę regulatora o strukturze MIMO. Dokonano wyboru funkcji przynależności dla poszczególnych zmiennych lingwistycznych. Określono wzajemne położenie oraz zakresy funkcji przynależności. Zdefiniowano bazę reguł regulatora rozmytego. Zaprezentowano dwie powierzchnie sterowania regulatora dla metody defuzyfikacji środek sum.

DESIGN AND TUNING OF MAMDANI FUZZY CONTROLLERS

The article presents the basic principles of designing Mamdani fuzzy controllers. The general formula of the controller with MIMO structure was presented. The choice of membership functions for individual linguistic variables was performed. The relative position of membership functions and the ranges have been determined. The rule base of the fuzzy logic controller was defined. Two control surfaces of the controller for the Center of Sums Method were presented.

1. WSTĘP

Zagadnienie projektowania regulatorów rozmytych typu Mamdaniego ma złożony charakter. Wynika to zarówno z różnorodnych możliwości dotyczących wyboru określonych funkcji przynależności dla zmiennych procesu regulacji, jak również z samej specyfiki reguł lingwistycznych opisujących algorytm działania regulatora. Dodatkową kwestią jest ilość wejść i wyjść określająca w pewien sposób rozmiar bazy reguł, które dla struktury typu Mamdaniego posiadają poprzedniki i następniki w postaci rozmytej [2, 8].

Strojenie zaprojektowanego regulatora rozmytego wymaga modyfikacji kształtu funkcji przynależności oraz ich wzajemnego położenia dla poszczególnych zmiennych procesu regulacji. W konsekwencji procedura strojenia pozwala na kształtowanie nieliniowej powierzchni sterowania regulatora. Powierzchnia ta jest złożona z segmentów przestrzennych, które mogą posiadać różne stopnie wypukłości. Ilość segmentów wzrasta wraz z ilością funkcji przynależności opisujących zmienne procesu regulacji. Z jednej strony pozwala to na bardzo elastyczne kształtowanie nieliniowej powierzchni sterowania,

¹ Politechnika Częstochowska, Wydział Elektryczny, ul. Armii Krajowej 17, 42-200 Częstochowa, tel.: + 48 34 32 50 884, fax: + 48 34 32 50 821, e-mail: kolesiak@el.pcz.czest.pl

ale jednocześnie powoduje wzrost ilości parametrów i tym samym zwiększenie trudności ze strojeniem regulatora [3, 8, 10].

2. STRUKTURA REGULATORA ROZMYTEGO TYPU MAMDANIEGO

Regulatory typu Mamdaniego występują zarówno w najprostszyc odmianach SISO (single-input single-output), jak również w bardziej złożonych, czyli MISO (multiple-input single-output) oraz MIMO (multiple-input multiple-output). Wybór konkretnej struktury jest głównie uwarunkowany wymaganiami procesu regulacji. Regulator Mamdaniego o strukturze MIMO można opisać za pomocą następującego ogólnego zbioru reguł [1, 3, 4]:

$$\begin{aligned}
 R^{(1)} : & \text{IF } (x_1 \text{ is } LX_1^{(1)}) \text{ AND... AND } (x_n \text{ is } LX_n^{(1)}) \\
 & \text{THEN } (u_1 \text{ is } LU_1^{(1)}) \text{ AND... AND } (u_m \text{ is } LU_m^{(1)}) \\
 & \dots\dots\dots \\
 R^{(k)} : & \text{IF } (x_1 \text{ is } LX_1^{(k)}) \text{ AND... AND } (x_n \text{ is } LX_n^{(k)}) \\
 & \text{THEN } (u_1 \text{ is } LU_1^{(k)}) \text{ AND... AND } (u_m \text{ is } LU_m^{(k)})
 \end{aligned}
 \tag{1}$$

gdzie: x_1, \dots, x_n - wejściowe zmienne lingwistyczne,
 u_1, \dots, u_m - wyjściowe zmienne lingwistyczne,
 $LX_1^{(1)}, \dots, LX_n^{(1)}, LX_1^{(k)}, \dots, LX_n^{(k)}$ - wartości lingwistyczne zmiennych wejściowych,
 $LU_1^{(1)}, \dots, LU_m^{(1)}, LU_1^{(k)}, \dots, LU_m^{(k)}$ - wartości lingwistyczne zmiennych wyjściowych,
 k - ilość reguł.

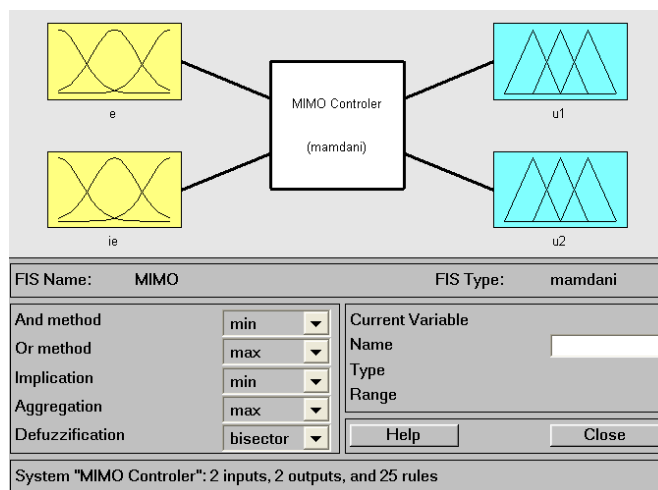
Do realizacji regulatora rozmytego można zastosować przybornik Fuzzy Logic Toolbox [6, 7], który jest opcjonalnym elementem składowym programu Matlab/Simulink. Projektowanie regulatora rozpoczynamy od określenia typu oraz struktury, która determinuje ilość wejść i wyjść zmiennych lingwistycznych. Następnie dokonujemy wyboru metod realizujących operacje iloczynu i sumy zbiorów rozmytych. Do tego celu stosuje się następujące zależności [5, 9]:

$$\forall x \in X : \mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x)) \tag{2}$$

$$\forall x \in X : \mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x)) \tag{3}$$

gdzie: x - element przestrzeni rozważań,
 X - przestrzeń rozważań;
 μ_A, μ_B - funkcje przynależności zbiorów rozmytych A i B,
 $\mu_A(x), \mu_B(x)$ - stopnie przynależności elementu x do zbiorów A i B.

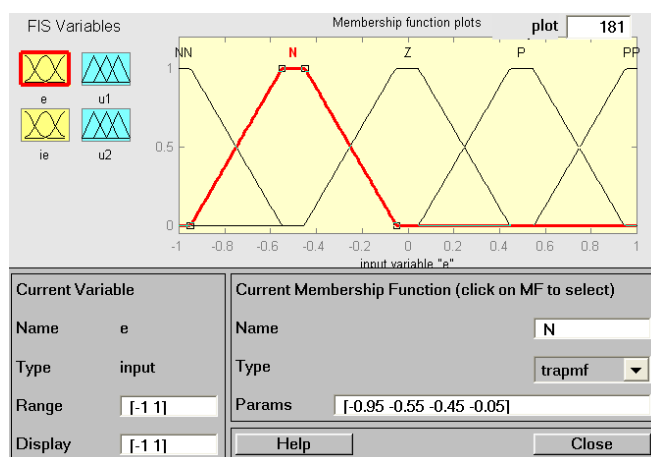
Następnym etapem projektowania jest wybór algorytmów implikacji i agregacji, a także wstępne określenie metody defuzyfikacji. W konsekwencji otrzymujemy przykładowy regulator rozmyty Mamdaniego o strukturze MIMO zaprezentowany na rys.1.



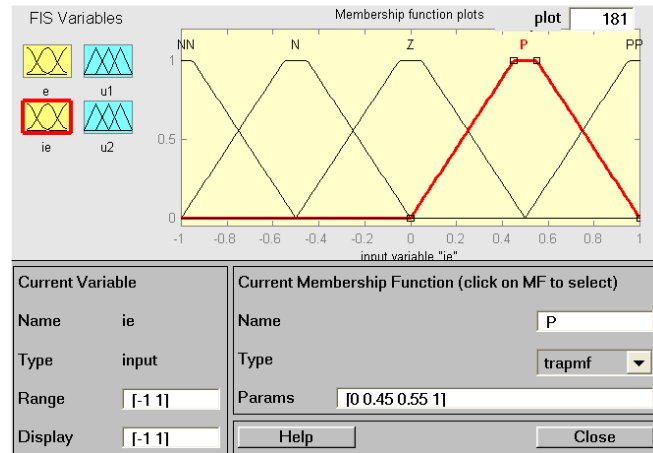
Rys. 1. Widok ogólny regulatora rozmytego Mamdaniego o strukturze MIMO wraz z modułami zmiennych lingwistycznych

3. FUNKCJE PRZYNALEŻNOŚCI ZMIENNYCH LINGWISTYCZNYCH

Projektowany regulator posiada dwa moduły wejściowych zmiennych lingwistycznych e oraz ie , a także dwa moduły wyjściowych zmiennych lingwistycznych $u1$ oraz $u2$. W odniesieniu do każdego modułu należy dokonać wyboru typu funkcji przynależności, ich ilości oraz wzajemnego położenia, które określa wysokość przecięcia funkcji sąsiednich. W przypadku wejściowych zmiennych lingwistycznych zastosowano funkcje trapezowe, których wykresy zaprezentowano na rys. 2 i 3.



Rys. 2. Trapezowe funkcje przynależności wejściowej zmiennej lingwistycznej e



Rys. 3. Trapezowe funkcje przynależności wejściowej zmiennej lingwistycznej *ie*

Wejściowe zmienne lingwistyczne przedstawione na rys. 2 i 3 zawierają 5 trapezowych funkcji przynależności określających błąd *e* oraz całość błędu *ie*. Funkcje oznaczono w sposób następujący: NN - średnia ujemna, N - mała ujemna, Z - zerowa, P - mała dodatnia, PP - średnia dodatnia. Ogólną zależność opisującą funkcję trapezową można przedstawić w następującej postaci [5, 9]:

$$\mu(x) = \begin{cases} 0 & \text{dla } x = a \\ 1 & \text{dla } x = b \\ 1 & \text{dla } x = c \\ 0 & \text{dla } x = d \end{cases} \quad (4)$$

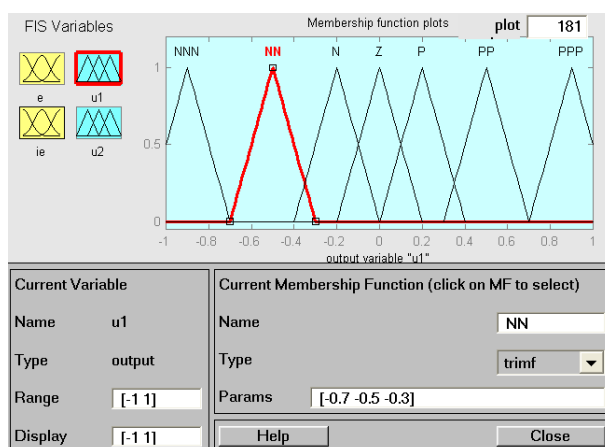
gdzie: *a*, *b*, *c*, *d* - punkty charakterystyczne funkcji przynależności.

W odniesieniu do wyjściowej zmiennej lingwistycznej *u1* zastosowane trójkątne funkcje przynależności rozmieszczone nierównomiernie w znormalizowanym przedziale [-1, 1]. Do opisu funkcji trójkątnej można wykorzystać zależność (4) przy spełnieniu warunku *b* = *c* dla punktów charakterystycznych. W przypadku zmiennej lingwistycznej *u2* wykorzystano równomiernie rozmieszczone funkcje przynależności Gaussa, które przyjmują następującą postać ogólną [3, 4]:

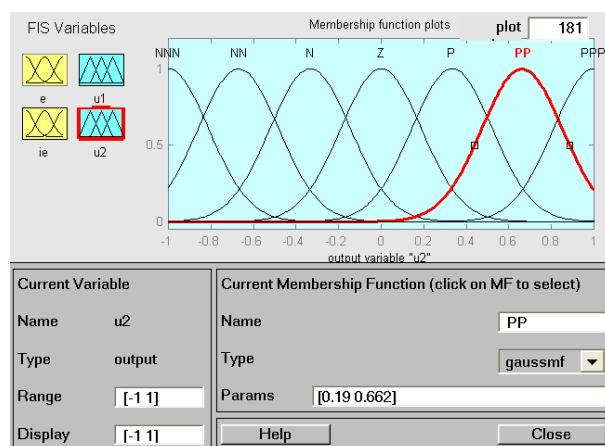
$$\mu(x) = e^{-\left(\frac{x-m}{w}\right)^2} \quad (5)$$

gdzie: *m* - wartość modalna,
w - szerokość.

Zastosowano łącznie 7 funkcji, dodatkowo wprowadzając oznaczenia: NNN - duża ujemna oraz PPP - duża dodatnia. Ilustracje graficzne wyjściowych zmiennych lingwistycznych przedstawiono na rys. 4 i 5.



Rys. 4. Trójkątne funkcje przynależności wyjściowej zmiennej lingwistycznej u_1



Rys. 5. Funkcje przynależności Gaussa wyjściowej zmiennej lingwistycznej u_2

4. BAZA REGUŁ I ALGORYTM DEFUZYFIKACJI

Po wprowadzeniu danych dotyczących wejściowych i wyjściowych funkcji przynależności należy przystąpić do zdefiniowania bazy reguł projektowanego regulatora rozmytego. Wykorzystano do tego celu standardową tablicę Mac Vicara-Whelana [1, 9], która została zmodyfikowana przy uwzględnieniu ilości wejściowych i wyjściowych funkcji przynależności opisujących zmienne lingwistyczne. Opracowaną bazę reguł przedstawiono w tabeli 1.

Tab. 1. Baza reguł regulatora rozmytego Mamdaniego o strukturze MIMO

Błąd e	Całka błędu ie				
	NN	N	Z	P	PP
NN	NNN	NNN	NN	N	Z
N	NNN	NN	N	Z	P
Z	NN	N	Z	P	PP
P	N	Z	P	PP	PPP
PP	Z	P	PP	PPP	PPP

Zdefiniowaną bazę reguł określającą wzajemne zależności pomiędzy wejściowymi i wyjściowymi zmiennymi lingwistycznymi należy wprowadzić z wykorzystaniem edytora reguł do modelu projektowanego regulatora rozmytego. W odniesieniu do wszystkich reguł zastosowano współczynnik wagowy równy jedności oraz metodę łączenia poprzedników reguł z wykorzystaniem formuły iloczynu. Kolejność wprowadzania poszczególnych reguł z punktu widzenia charakterystyki sterowania nie ma znaczenia, niemniej zazwyczaj są one wprowadzane według kolejności wierszy lub kolumn. Odpowiednie uporządkowanie wpisywanych reguł ułatwia późniejsze sprawdzenie ich poprawności.

Kolejnym etapem projektowania regulatora rozmytego jest wybór metody defuzyfikacji, która umożliwia przekształcenie zbioru rozmytego, opisanego odpowiednią funkcją przynależności, na wartość skalarną. Przybornik Fuzzy Logic Toolbox oferuje łącznie pięć metod defuzyfikacji, w tym metody ciągłe oraz nieciągłe [7]. W odniesieniu do pojęcia defuzyfikacji ciągłość metody można zdefiniować przy pomocy następującej ogólnej zależności [1, 4, 9]:

$$\forall \varepsilon > 0 \exists (\delta > 0 \wedge d > 0):$$

$$|x_1(d) - x_1(d-1)| < \delta \wedge \dots \wedge |x_n(d) - x_n(d-1)| < \delta \quad (6)$$

$$\Rightarrow |u_1(d) - u_1(d-1)| < \varepsilon \wedge \dots \wedge |u_m(d) - u_m(d-1)| < \varepsilon$$

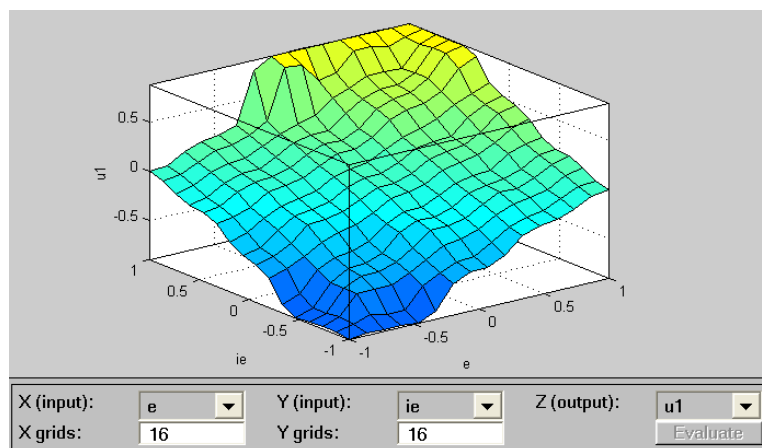
gdzie: δ - zmiana wartości sygnałów wejściowych,
 ε - zmiana wartości sygnałów wyjściowych,
 d - wybrany krok dyskretyzacji,
 x_1, \dots, x_n - wejściowe zmienne lingwistyczne,
 u_1, \dots, u_m - wyjściowe zmienne lingwistyczne.

Wyznaczenie powierzchni sterowania regulatora rozmytego przy zastosowaniu metody środka sum (Center of Sums) jest realizowane dla przypadku dyskretnego w oparciu o następującą zależność [1, 3]:

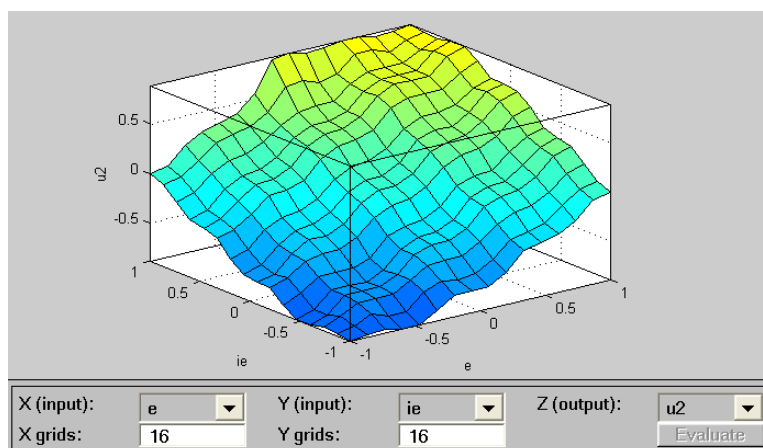
$$u = \frac{\sum_{i=1}^m u_i \cdot \sum_{k=1}^n \mu_{\text{CLU}(k)}(u_i)}{\sum_{i=1}^m \sum_{k=1}^n \mu_{\text{CLU}(k)}(u_i)} \quad (7)$$

gdzie: $\mu_{CLU(k)}$ - ściśnięta funkcja przynależności dla każdej k-tej reguły .

Graficzne ilustracje zastosowania metody środka sum w procesie defuzyfikacji w odniesieniu do projektowanego regulatora Mamdaniego o strukturze MIMO przedstawiono na rys. 6 i 7.



Rys. 6. Zależność sygnału wyjściowego u_1 od sygnałów wejściowych e oraz ie regulatora rozmytego dla algorytmu defuzyfikacji bisector



Rys. 7. Zależność sygnału wyjściowego u_2 od sygnałów wejściowych e oraz ie regulatora rozmytego dla algorytmu defuzyfikacji bisector

5. WNIOSKI

Powierzchnie sterowania, które uzyskano w przypadku projektowanego regulatora rozmytego Mamdaniego o strukturze MIMO wykazują różnice w kształcie wynikające z tego, iż dla pierwszej wyjściowej zmiennej lingwistycznej zastosowano nierównomierne rozmieszczenie funkcji przynależności, a w odniesieniu do drugiej zmiennej lingwistycznej wykorzystano funkcje rozłożone równomiernie w znormalizowanym przedziale. Pewien wpływ na kształt powierzchni sterowania regulatora ma również zróżnicowanie wyjściowych funkcji przynależności, poprzez zastosowanie funkcji trójkątnych oraz Gaussa.

Wykorzystanie przybornika Fuzzy Logic Toolbox zawartego w pakiecie Matlab pozwala na projektowanie regulatorów rozmytych typu Mamdaniego w postaci następujących struktur: SISO, MISO oraz MIMO. W rozwiązaniach praktycznych najczęściej stosuje się dwie pierwsze struktury regulatorów, natomiast trzecia ma mniejszy zakres zastosowania ze względu na znaczny stopień skomplikowania bazy reguł.

Oprogramowanie umożliwia zmiany takich parametrów jak: rodzaj funkcji przynależności, lokalizacja funkcji w wybranym przedziale, wysokość przecięcia sąsiednich funkcji, ilość reguł oraz zastosowana metoda defuzyfikacji. Wprowadzając zmiany i modyfikacje wybranych parametrów można badać ich wpływ na kształt segmentów nieliniowej powierzchni sterowania regulatora.

Ze względu na szeroki zakres możliwości konfiguracyjnych przybornik Fuzzy Logic Toolbox pakietu Matlab stanowi bardzo pomocne i jednocześnie efektywne narzędzie do projektowania, strojenia oraz badania regulatorów rozmytych.

6. BIBLIOGRAFIA

- [1] Driankov D., Hellendoorn H., Reinfrank M.: *Wprowadzenie do sterowania rozmytego*, Warszawa, WNT 1996.
- [2] Fan X., Zhang N., Li N.: *Structure analysis of typical fuzzy controllers with unequally spaced fuzzy sets for input and output variables*, Chinese Journal of Electronics 10, 2001, pp. 326–331.
- [3] Kacprzyk J.: *Wieloetapowe sterowanie rozmyte*, Warszawa, WNT 2001.
- [4] Passino K. M., Yurkovich S.: *Fuzzy Control*, California, Menlo Park, Addison Wesley Longman 1998.
- [5] Piegat A.: *Modelowanie i sterowanie rozmyte*, Warszawa, Akademicka Oficyna Wydawnicza EXIT 1999.
- [6] Sivanandam S.N., Sumathi S., Deepa S. N.: *Introduction to Fuzzy Logic using MATLAB*, Berlin, Springer-Verlag 2006.
- [7] The Math Works: *Fuzzy Logic Toolbox for use with Matlab - User's Guide*, 2002.
- [8] Tu K. Y., Lee T. T., Wang W. J.: *Design of a multi-layer fuzzy logic controller for multi-input multi-output systems*, Fuzzy Sets and Systems 111, 2000, pp. 199-214.
- [9] Yager R. R., Filev D. P.: *Podstawy modelowania i sterowania rozmytego*, Warszawa, WNT 1995.
- [10] Ying H. : *TITO Mamdani Fuzzy PI/PD Controllers as Nonlinear, Variable Gain PI/PD controllers*, International Journal of Fuzzy Systems, Vol. 2, No. 3, 2000, pp. 191-196.