

MODELLING OF DATA TRANSMISSION USING MATLAB FUZZY TOOLBOX SOFTWARE

Streszczenie

This article is a result of works on modelling data transmission rate in sea transportation. While working on this project fuzzy logic controller was developed. On the very beginning several membership functions and rules of inference were created on the basis of theoretical assumptions. After that appropriate method of aggregation was chosen. However during the research appears a need to review this assumptions and verify correct operation of mentioned controller. Fuzzy Toolbox software was used for this purpose. It is a library of Matlab Simulink created especially to design fuzzy controllers and drivers. The process of developing fuzzy logic controller is presented in this paper together with the conclusions and verification of initial assumptions.

INTRODUCTION

This article is an outcome of a project concentrating on modelling data transmissions in maritime transportation. As a part of this project a fuzzy logic controller was developed. At the very beginning membership functions were defined and inference rules selected (initial values base on theoretical assumptions). At the end the most appropriate method of aggregation was implemented. However the results were satisfactory there was a certain need to review the initial assumptions. To deal with that a Fuzzy Toolbox software was used. This is a library of Simulink designed to develop and test fuzzy controllers and systems. The whole process of creation of fuzzy controller with a great help of this software is presented in this paper. Conclusions coming from verification of the initial assumptions are also included and discussed.

1. FUZZY LOGIC AND FUZZY TOOLBOX MATLAB

Fuzzy logic is a section of mathematics that allows for a formal record of uncertainty. It is an example of multi-valued logic. The purpose of fuzzy set theory is a need to mathematically describe situations which are ambiguous and imprecise. Fuzzy systems are used where there is no sufficient knowledge of the mathematical model which describes a certain situation and where recreation of this model becomes for some reason impossible. For the purposes of classical logic, element which is member of the set should be described as follows [4], [7]:

$$\mu(A) = \begin{cases} 1, & \text{if } x \in A \\ 0, & \text{if } x \in \emptyset \end{cases} \quad (1)$$

In fuzzy logic, each element can be a member of the set to an extent other than 1 and 0.

$$\mu(A) = \begin{cases} 1, & \text{when membership criteria is fulfilled completely} \\ n \in (0,1), & \text{when membership criteria is partially fulfilled} \\ 0, & \text{when membership criteria is not fulfilled at all} \end{cases} \quad (2)$$

Fuzzy Logic Toolbox is a library of Matlab Simulink and provides a lot of functions and interfaces which facilitate the design of fuzzy controllers and systems. This library includes Simulink blocks that can be used as a model of designed fuzzy controller. The library includes a complete development environment for modelling and testing dynamic systems using fuzzy sets and fuzzy rules of

inference. It also has a number of tools useful when designing intelligent control systems, which operation is based on fuzzy logic elements. It also allows users to work on fuzzy logic blocks and schemes using a graphical editor or text commands [3]. Simulink environment provides an easy and understandable way to create a digital model of the proposed controller. Fuzzy Toolbox library makes possible to check the computational model and the potential adjustments to the initial assumptions. Verification of this assumptions is fairly easy and the results of any changes or corrections are visible straight away.

2. THEORETICAL ASSUMPTIONS OF FUZZY CONTROLLER MODEL

Fuzzy controller principles are based on an approximation of the actual process of implementing functions. Controller receives input values about the actual situations and processes this information to control the frequency of data transmission. In presented example input values describing a real situation are CPA and TCPA. A control value on output will determine the rate of data transmission between two objects. General assumption of presented method is that any two objects are more dangerous to each other, the more often they should exchange information.

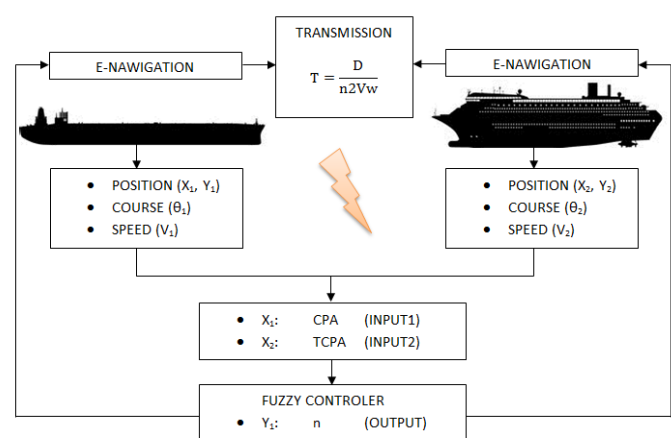


Fig. 1. The exchange of data between vessels using fuzzy controller.

2.1. Membership functions of CPA and TCPA

Membership function is a generalized form of the function characteristic for particular fuzzy set. In classic logic it takes value 1 when the element fully belongs to the set, or 0 if not. In fuzzy sets theory, fuzzy element may belong to the set to a certain extent, so the membership function can take values from 0 to 1. Membership function answers to the question - how much an element belongs to a fuzzy set.

The membership functions should meet the following recommendation [8], [9], [10]:

- At the borders of membership functions an element should belong 0.5 to each of them.
- Membership functions should fulfil the requirement of the division of unity. Point located in the transition zone and belonging to the extent p to one area should belong to the extent of $1-p$ to the next area.
- Interval of rise (fall) membership function should be determined within the scope of error associated with the estimation of input values $[-\sigma_{CPA}, +\sigma_{CPA}]$ or $[-\sigma_{TCPA}, +\sigma_{TCPA}]$.
- Asymptotically membership function should aim to 1 within the nucleus (core) and get close to 0 near the base.

In addition, the intuitive expectations with respect to the shape of the membership function are expressed by Schwab axioms [8], [9], [10]:

- The membership functions $\mu(x)$ should be continuous over the range of X .
- The first derivative of membership function $d\mu(x)/dx$ must be continuous in the whole range of X .
- The second derivative of membership function $d^2\mu(x)/dx^2$ should be also continuous throughout the range of X .

Taking into account these recommendations a sigmoidal shaped function was chosen to represent CPA. In described example, the steepness of the function is different for the left and right side of the internal functions. Figure 2 shows a set of four sigmoidal membership functions, in order from the left side representing the following linguistic values: *very close*, *near*, *far* and *very far away*. The external membership functions are single sigmoidal functions the internal ones consists of two such functions.

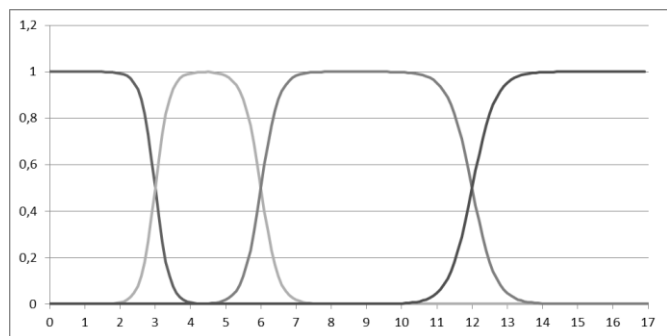


Fig. 2. The membership functions of CPA.

Taking into account the above recommendations and the fact that the linear approximation of TCPA error is sufficient for the purposes of this method, trapezoidal shaped membership functions was used to represent TCPA. Transition zones between successive membership functions will be determined by TCPA errors. The error value depends on several parameters, but not the TCPA itself, so theoretically increase of TCPA does not affect the value of the error.

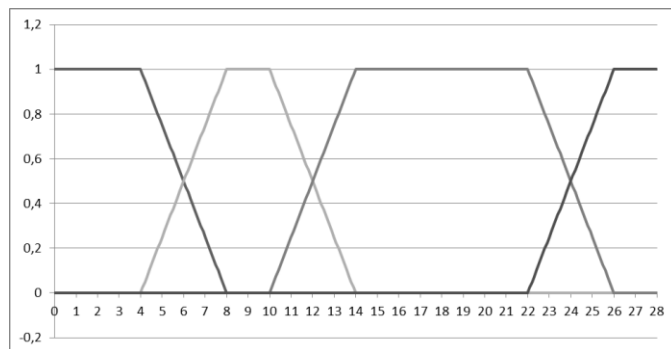


Fig. 3. The membership functions of TCPA.

Figure 3 shows a set of four polygonal and trapezoidal membership functions. In order from left to right they represent as follows: *very fast*, *fast*, *slow*, *very slow*.

2.2. Interference rules

Let the input values be CPA and TCPA, an output value y_k which in next step is used to calculate the coefficient ψ . This coefficient directly affects the transmission rate in presented method. Fuzzy controllers are designed this way that they try to imitate human reasoning. That is why reasoning part of controller is a database of interference rules which processing fuzzy input values into fuzzy output value [1], [5], [11].

Tab. 1. Inference rules for the developed method

| | | CPA | | | |
|------|-----------|---------------|---------------|-----------|---------------|
| | | very close | close | far away | very far away |
| TCPA | very fast | very frequent | very frequent | frequent | normal |
| | fast | very frequent | frequent | normal | rare |
| | slow | frequent | normal | rare | very rare |
| | very slow | normal | rare | very rare | very rare |

The rules allows to deduce the truth of the successor based on its predecessor, the implication is treated as a fuzzy relationship. The fuzzy controller rules vary depending on the situation. Interference rules represents knowledge about the possible values of linguistic variables, processes occurring when manoeuvring the vessel, COLREG rules, etc. In inference block each rule which conditions are met is activated. Overall, basing on the condition we are finding a new fuzzy set which is an effect of activation of certain interference rules. Table 1 presents a set of inference rules that have been programmed in the described controller. As action between fuzzy sets fuzzy operator (AND) was used, which was calculated using T-MIN operator [5], [12], [13].

2.3. The output membership function and aggregation rules

As a result of interference a new membership function $\mu(s)$ shown in figure 4 was obtained. It is necessary to help in defuzzification process.

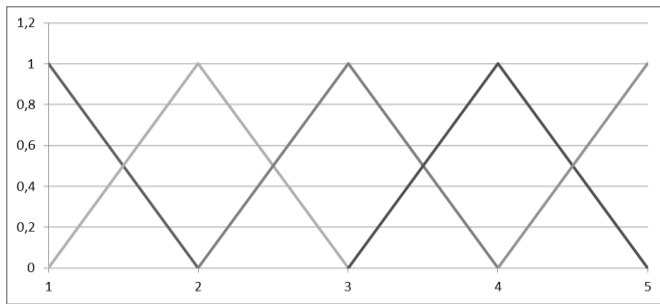


Fig. 4. Output membership function of the fuzzy values.

To control the rate of data transmission a value of coefficient ψ is needed. In presented method value y_k is an output of fuzzy controller. This value is used to calculate coefficient ψ but first aggregation has to be done to obtain y_k . In the theory of fuzzy sets several methods of aggregation can be used. They are generally divided into two groups Centre of Area (COA) and Medium of Maximum (MOM). An example of the first group is a method of centre of gravity of fuzzy set. An example of the second type is the method of the first maximum [6], [14].

In the method of the first maximum result of aggregation shall be the smallest y value corresponding to the maximum degree of membership functions $\mu(y)$. As shown in figure 5, with increasing activation of a right set, its representative $y_k = y$, it will move toward the maximum of this set. With the decrease of the degree of activation of the same set its representative move toward the left, but only until degree of activation of right set will be greater than the left set. The return value of this method is y_k such that:

$$\mu(y_k) = \max_{y \in Y}(\mu(y)) \text{ oraz } \bigvee_{y < y_0} \mu(y) < \mu(y_k) \quad (3)$$

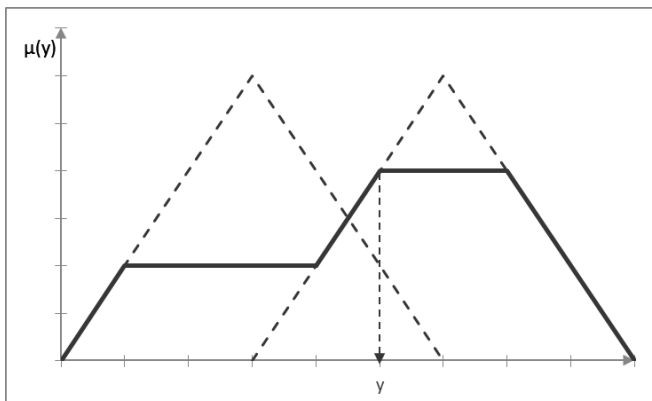


Fig. 5. The method of the first maximum [15]

On the other hand, using centre of gravity method y_k can be calculated with formula:

$$y_k = \frac{\int y\mu(y) dx}{\int \mu(y) dx} \quad (4)$$

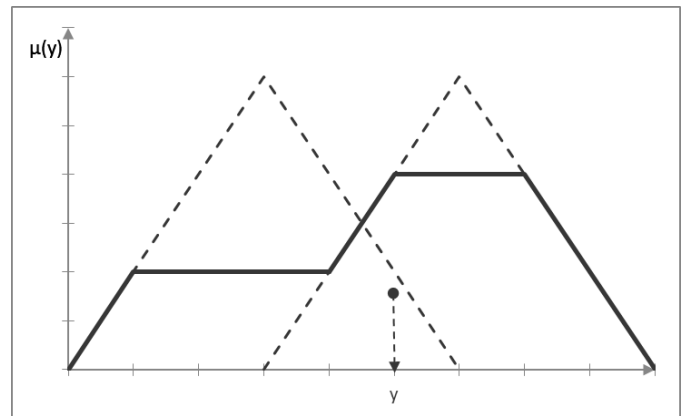


Fig. 6. The method of center of gravity

The above-described method of determining y_k imposes a relatively high computational complexity. Very often in the literature a simplified method can be found. The advantage of this method of calculation is to increase productivity by simplifying computations while preserving the advantages of the centre of gravity method.

$$y_k = \frac{\sum_{i=1}^n y_i \mu(y_i)}{\sum_{i=1}^n \mu(y_i)} \quad (5)$$

3. CREATING A MODEL OF FUZZY CONTROLLER USING MATLAB SOFTWARE

This chapter presents step-by-step process of development a fuzzy controller model in Matlab Fuzzy Toolbox software. At the beginning membership functions are programmed. Then database of inference rules is created as outlined in the previous section and shown in table 1. Later on a method of aggregation is chosen. Finally analysis and verification of developed method is carried out by simulation of different initial situations. Different input data is chosen (CPA and TCPA) to test and verify proper operation of the controller. Extreme input values and some closer to the middle of fuzzy sets were used for this test. The evaluation shall assess the proper application of membership functions, rules of inference and aggregation method.

3.1. Fuzzy Inference System Editor

It is used for setting the basic parameters of the proposed controller, such as the number of input and output variables, selection of operators that provide specific functions etc. There are two models available: Mamdani and Takagi-Sugeno.

Inference rules applied in our method has the following form:

$$IF (CPA = ...) AND (TCPA = ...) THEN (T = ...)$$

A natural consequence in presented controller is to use Mamdani. In the developed model two input values CPA and TCPA were introduced. The result of fuzzy system is output y_k which later on is used to calculate the ψ coefficient.

3.2. Membership Function Editor

It is used to display and model membership functions. It allows to set range, value and names labels for each parameter. In this example four sigmoidal membership functions for variable CPA and the four trapezoidal functions for variable TCPA were created.

3.3. Rule Editor and Rule Viewer

Rule Editor is used to create interference rules database. The user selects a linguistic value for each set and then the output value which is chosen if the rule is activated. The user can select one of two operators AND or OR, depending on what action is to be executed by the programmed rule. In this example, sixteen rules were created in accordance with table 1. In each rule operator AND was used.

Rule Viewer is useful tool which enables preview for specific rules in the process of inference. Input value can be changed and user can observe what influence does it have on each rule. Results of individual rules are accumulated in one fuzzy set. Later on it is converted to a specific numerical value, which is the output value of fuzzy inference system.

3.4. Surface Viewer

It is used to visualize the output parameter depending on the selected two input parameters. Figure 7 shows surface which represents output for fuzzy controller used in presented method. Local minimum indicates that for some input values of the CPA and TCPA (edge of the fuzzy set), fuzzy controller gives back smaller value than theoretically should. In this example, to get rid of the inaccuracies it was enough to change the method of aggregation rules.

CONCLUSIONS

Matlab Toolbox Fuzzy turns out to be a great tool for designing and testing controllers based on fuzzy logic. User can create a controller and test it computationally in safe environment. A number of charts and diagrams can be generated. This is an easy way to verify the correctness of calculation and compliance with the initial assumptions. With other components can keep track of the implementation of the various rules and in any way match the aggregation method to find the right one. Designed controller can be stored and used in other Simulink schemes and projects. In this example Fuzzy Toolbox software helped to find local minima and change aggregation method to get rid of them. Without this software tests of presented model would take lot of tedious hours of computational verification.

REFERENCES

1. Badurek J., „Logika rozmyta w bazach danych,” *Informatyka*, nr

- 1, pp. 14–17, 1999.
2. Biacino L. i Gerla G., „Fuzzy logic, continuity and effectiveness,” *Archive for Mathematical Logic*, nr 41(7), pp. 643–647, 2002.
3. Bodnar A., Honczarenko J., Jarczoch A. i Pawlukowicz P., „Application of fuzzy logic for controlling robotized machining systems,” *Prace Naukowe Instytutu Technologii Maszyn i Automatykacji Politechniki Wrocławskiej. Konferencje*, tom 69, nr 31, pp. 114–120, 1998.
4. Brown M., *An Introduction to Fuzzy and Neurofuzzy Systems*, 1996.
5. Czystzak T., „Analiza wybranych metod generowania bazy reguł systemów logiki rozmytej,” *Prace Naukowe Katedry Budowy Maszyn / Politechnika Śląska*, nr 4, pp. 117–126, 2005.
6. Czystzak T., Chowaniec K., Urbańczyk T. i Sokołowski A., „System wnioskowania rozmytego o strukturze Mamdani – analiza metod wyostrzania,” *Prace Naukowe Katedry Budowy Maszyn / Politechnika Śląska*, nr 4, pp. 127–138, 2005.
7. Dybiec E. i Tran C., „Logika rozmyta: początek, rozwój i możliwości zastosowania,” *Inżynieria Morska i Geotechnika*, nr 5, pp. 418–422, 2001.
8. Filipowicz W., „Fuzzy Evidence Reasoning and Position Fixing,” *Frontiers in Artificial Intelligence and Applications – Knowledge-Based Intelligent Engineering Systems*, pp. 1181–1190, 2012.
9. Filipowicz W., „Fuzzy Reasoning Algorithms for Position Fixing,” *PAK, Pomiary Automatyka Kontrola*, nr R.56 nr12, pp. 1491–1495, 2010.
10. Filipowicz W., „New Approach towards Position Fixing,” *Annual of Navigation*, nr 10, pp. 41–54, 2010.
11. Fulmański P. i Grzanek M., „Logika rozmyta i SQL,” *Zeszyty Naukowe Wydziału ETI Politechniki Gdańskiej. Technologie Informacyjne*, nr 11, pp. 857–864, 2006.
12. Kowalczyk-Niewiadomy A. i Pelikant A., „Przetwarzanie zapytań w metajęzyku naturalnym za pomocą algorytmów rozmytych,” *Studia Informatica*, tom 31, nr 2A, pp. 479–488, 2012.
13. Kuchta M., „Optymalizacja bazy reguł systemu logiki rozmytej z zastosowaniem sztucznych systemów immunologicznych,” *Prace Naukowe Katedry Budowy Maszyn / Politechnika Śląska*, nr 1, pp. 189–198, 2009.
14. Novák V., „On fuzzy type theory,” *Fuzzy Sets and Systems*, nr 149(2), pp. 235–273, 2005.
15. Ross T. J., *Fuzzy logic with engineering applications*, John Wiley&Sons, 2004.

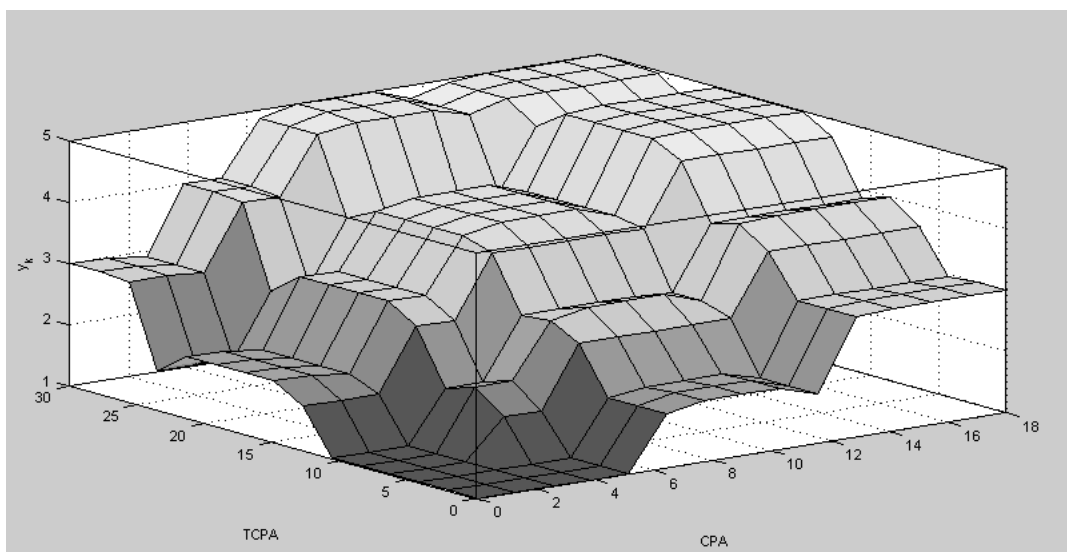


Fig. 7. Surface view for presented method.

MODELOWANIE TRANSMISJI DANYCH Z WYKORZYSTANIEM MATLAB FUZZY TOOLBOX

Streszczenie

Ten artykuł powstał podczas prac nad modelowaniem częstości transmisji danych w transporcie morskim. Podczas prac przy tym projekcie stworzono sterownik wykorzystujący logikę rozmytą. Na podstawie teoretycznych założeń stworzono funkcje przynależności i reguły wnioskowania oraz wybrano metodę agregacji. Pojawiła się jednak potrzeba weryfikacji założeń i sprawdzenia prawidłowości działania sterownika. W tym celu użyto narzędzia Fuzzy Toolbox, które jest biblioteką Matlab Simulink stworzoną w celu projektowania regulatorów i sterowników rozmytych. Proces tworzenia sterownika rozmytego za pomocą tegoż oprogramowania oraz wnioski płynące z weryfikacji założeń początkowych zostały przedstawione w niniejszym opracowaniu.

Autor:

Filipkowski Damian - Akademia Morska w Gdyni, Katedra Nawigacji, d.filipkowski@wn.am.gdynia.pl