Takagi–Sugeno fuzzy systems as a method of acting opponents in games

Karolina Kęsik
Institute of Mathematics
Silesian University of Technology
Kaszubska 23, 44-100 Gliwice, Poland
Email: karola.ksk@gmail.com

***Abstract*—Controlling a hero in 2D or 3D games can occur on different rules. To increase the level of gameplay, there are objects in the entire game with which the player can interact. Such an action is quite often implemented using artificial intelligence, which has been growing in recent years. In this paper, we propose using a model of a fuzzy system to move objects in the games. Movement takes place on the principle of moving between two points, where the first one is the player's position and the second is destination point. The paper presents the mathematical model of the Takagi–Sugeno system and tests for the correctness of the movement.**

I. Introduction

The field of artificial intelligence is developing rapidly, which is visible in various applications around us. Especially in devices of the Internet of Things, whose times are coming thanks to the 5G network. Enabling communication between a huge number of devices results in acquiring and generating a large amount of digital information. This information must be processed so that it can be forwarded or take some decisions based on it. More often, such problems are solved using artificial intelligence (AI) techniques.

In practice, almost everyone uses a smartphone, on which various applications are installed. Especially often of these applications are games that use AI methods. Their practical use is based on the movement of opponents or the generation of boards/environment. The movement can be simulated using various tools such as artificial neural networks, fuzzy controllers and heuristic algorithms. In this paper, we show the simple use of the Takagi-Sugeno system to simulate the movement of objects.

A. Related works

Heuristic methods are used only to simulate player action [1], but also to create a game environment. An example is the creation of labyrinths [2]. The same algorithms can be used for play-testing in various games, especially in card ones [3]. Quite often, there is a need for the opponent's intelligence to be trained. This is a popular situation with games that have large amounts of different rules, such as chess [4] or go [5]. The second of these games is much more difficult, which made training AI quite problematic. In [6] deep learning of neural networks and tree searching were used to obtain impressive results. A year later, the authors presented a learning algorithm without the need for knowledge given by programmers [6]. Again in [7], the authors presented idea to train deep neural networks to play othello ie a game on a board with 64 fields and black and white verticals. An interesting idea is to extract knowledge from how others play the game [8], or using the idea of games in other areas, such as medicine. An example is creating decision systems using game theory and rough sets [9].

The idea of multi-agents and collective using many objects is a frequent use in games. The results of research on this topic are presented in [10], [11] using fuzzy logic. This field of science can also be used in solving various problems [12], [13].

II. Fuzzy logic

Fuzzy logic was proposed by L. A. Zadeh in 1965, where unlike classical sets, partial membership is allowed [14]. So the sets can be described using the adaptation function

$$\mu_A : X \to [0, 1]. \tag{1}$$

Fuzzy set A on space X can also be described by a set of ordered pairs $(x, \mu_A(x))$, where $\mu_A(x)$ is the degree of ownership of the x object to the fuzzy set A

$$A = \{(x, \mu_A(x)); x \in X, \mu_A \in [0, 1]\}. \tag{2}$$

Zadeh suggested another notation for fuzzy sets. For the discrete space X, we have

$$A = \sum_{x \in X} \mu_A(x)/x \tag{3}$$

Again, for continuous space X

$$A = \int_X \mu_A(x)/x \tag{4}$$

These markings should be treated as the sum of elements x having a degree of belonging $\mu_A(x)$. Mark / should be treated as a separator.

The most commonly used affiliation functions that describe the sets are as follows

- triangular

$$\mu_A(x; a, b, c) = \begin{cases} 0, & x \le a, \\ \frac{x-a}{b-a}, & a < x \le b, \\ \frac{c-x}{c-b}, & b < x \le c, \\ 0, & x > c \end{cases}, \quad (5)$$

where a, b, c are parameters $(a \le b \le c)$.

- Gaussian

$$\mu_A(x; m, \sigma) = e^{-\frac{(x-m)^2}{2\sigma^2}}, \quad (6)$$

where m, σ are parameters. For $x = m$, this function assumes the value 1 and the parameter $\sigma > 0$ determines the width of the fuzzy set.

- inverse trapezoid

$$\mu_A(x; a, b, c, d) = \begin{cases} \frac{b-x}{b-a} & a < x \le b, \\ 0 & b < x \le c, \\ \frac{x-c}{d-c} & c < x < d \end{cases}, \quad (7)$$

where a, b, c and d are parameters $(a \le b \le c \le d)$.

In addition, we distinguish two types of fuzzy sets. Type-1 fuzzy sets on the one hand describe uncertain terms, and on the other hand, the values of the membership function are precise, because they are specific numbers. The idea was born that the values of the membership function were fuzzy sets and were called fuzzy sets of type-2. Fuzzy sets that represent the values of membership functions are described precisely. Hence, the further generalization that they can be represented again by fuzzy sets. This reasoning can be continued indefinitely, creating the concept of fuzzy sets of type-m.

Apart from the harvest itself, we define the concept of a fuzzy number $\tilde{A}$, which is a convex and normal fuzzy set with a limited medium, defined on the axis of real numbers $\mathbb{R}$.

A. Fuzzy system

A fuzzy system is a set of interrelated elements separated from the environment. Characteristic for the system is the fact that internal connections are dominant in relation to connections with elements from outside the system. If connections with elements outside the system occur, then we understand them as inputs and outputs from the system. We say then that the system processes the input values to the output ones. At the entrance of such a system we define

- linguistic values (in the form of fuzzy sets),
- numerical values - where is the need to apply some fuzzy operations.

Each fuzzy system contains a knowledge base written in the form of fuzzy conditional rules *if–then* and so-called an approximate motor of approximation based on fuzzy set theory and fuzzy reasoning [15]. The aforementioned motor combines parts of knowledge contained in the rules *if-then* and with approximate inference methods transforms the values of input variables of the system into the values of its output variables.

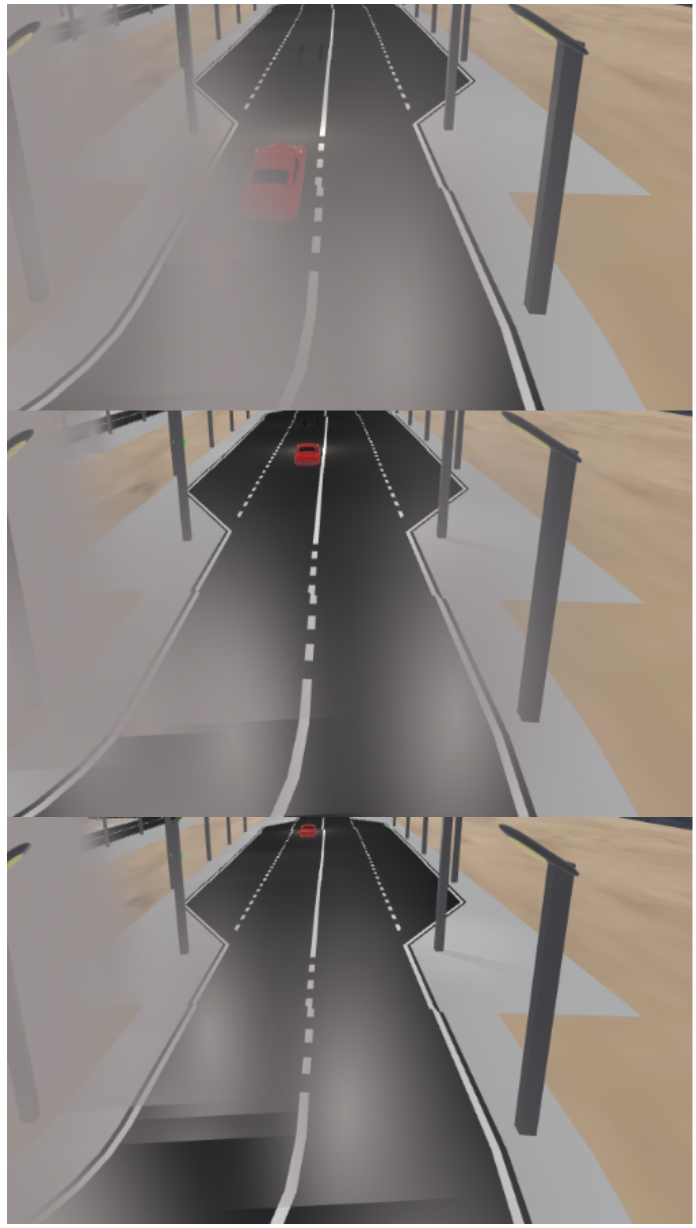


Figure 1: Sample screenshots using Takagi–Sugeno to drive a car on a short distance.

III. Takagi–Sugeno fuzzy systems

The knowledge base of this system [16] is a set of *if-then* rules

$$\begin{aligned} \mathcal{R} &= \left\{ \mathcal{R}^{(i)} \right\}_{i=1}^{l} \\ &= \left\{ \text{if } \wedge_{n=1}^{N} x_{0n} \text{ is } A_n^{(i)}, \text{ then } y = f_i(x_0) \right\}_{i=1}^{l}, \end{aligned} \quad (8)$$

where l is the amount of rules in knowledge base, x_{0n} is input singleton, $x_0 = [x_{01}, x_{02}, \ldots, x_{0N}]^T$, $A_n^{(i)}$ is the linguistic value and $y = f(x_0)$ is a function in the conclusion i-th *if-then* rule.

Each of the conditional rules leads to the numeric type output value, and output is determined as a weighted average calculate using the following formula

$$y_0 = \frac{\sum_{i=1}^{I} F^{(i)}(x_0), \quad y^{(i)}(x_0)}{\sum_{j=1}^{I} F^{(j)}(x_0)} \tag{9}$$

where $y^{(i)}(x_0) = f_i(x_0)$ determines the position of the singleton in conclusion of i-th rule or or numerical result of inference using i-th rule, $F^{(i)}(x_0)$ is activation degree of i-th rule defined as

$$F^{(i)}(x_0) = \mu_{A^{(i)}}(x_{01}) \star_T \mu_{A_2^{(i)}}(x_{02}) \star_T \ldots \star_T \mu_{A_{\mathcal{N}}^{(i)}}(x_{0\mathcal{N}}), \tag{10}$$

where $\star_T$ means t-norm.

A function $y = f_i(x_0)$ in conclusion of i-th rule can be understand as fuzzy singleton with the following membership function

$$\mu_{B^{(i)}}(y) = \delta_{y,y^{(i)}(x_0)} = \begin{cases} 1, & y = y^{(i)}(x_0), \\ 0, & y \neq y^{(i)}(x_0). \end{cases} \tag{11}$$

```
Start;
Define the object;
Define the goal;
Create rules;
Create a system based on previously created rules;
Create a set composed of angles and the current angle
  of the object, distances and distance between the
  target and the current position;
while the target did not reach its destination do
    Calculate the angle value using the system and the
      current object parameters;
    Correct the angle of the object;
    Move the object forward;
end
Stop;
```

Algorithm 1: The code of moving the object in the game using the Takagi–Sugeno system.

IV. Takagi–Sugeno fuzzy system for games

Takagi–Sugeno systems can be responsible for the movement of certain objects in 2D or even 3D games. The main idea is that the movement takes place between two points – the location of the object, and the planned destination position. To make this possible, a base of rules must be created. As previously stated, the rules are based on *if-then*. The simplest way is to create two variables – *distance* and *angle*. The first one is responsible for the moving distance and contains three linguistic variables from the following set $\{left, straight, right\}$ and a triangular membership function,

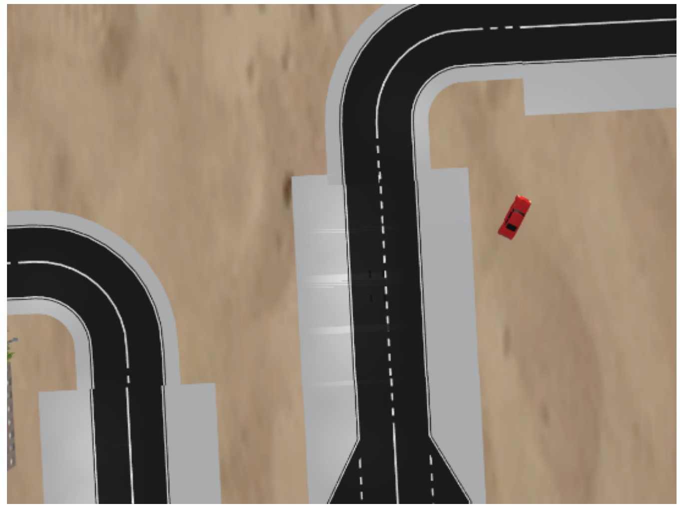


Figure 2: The movement of the car depending on the destination located in a very far distance ahead of the car on the black surface.

and the second one is a movement angle. The variable has one of two membership functions – triangular or inverse trapezoid, where the points of the bend will be the angles given in the numerical angles from the range $\langle 0, 360 \rangle$. Rules were created based on these two variables and given functions. Each rule takes input and output values and returns value of the weight. The created rule model works on the principle of adjusting the values of variables.

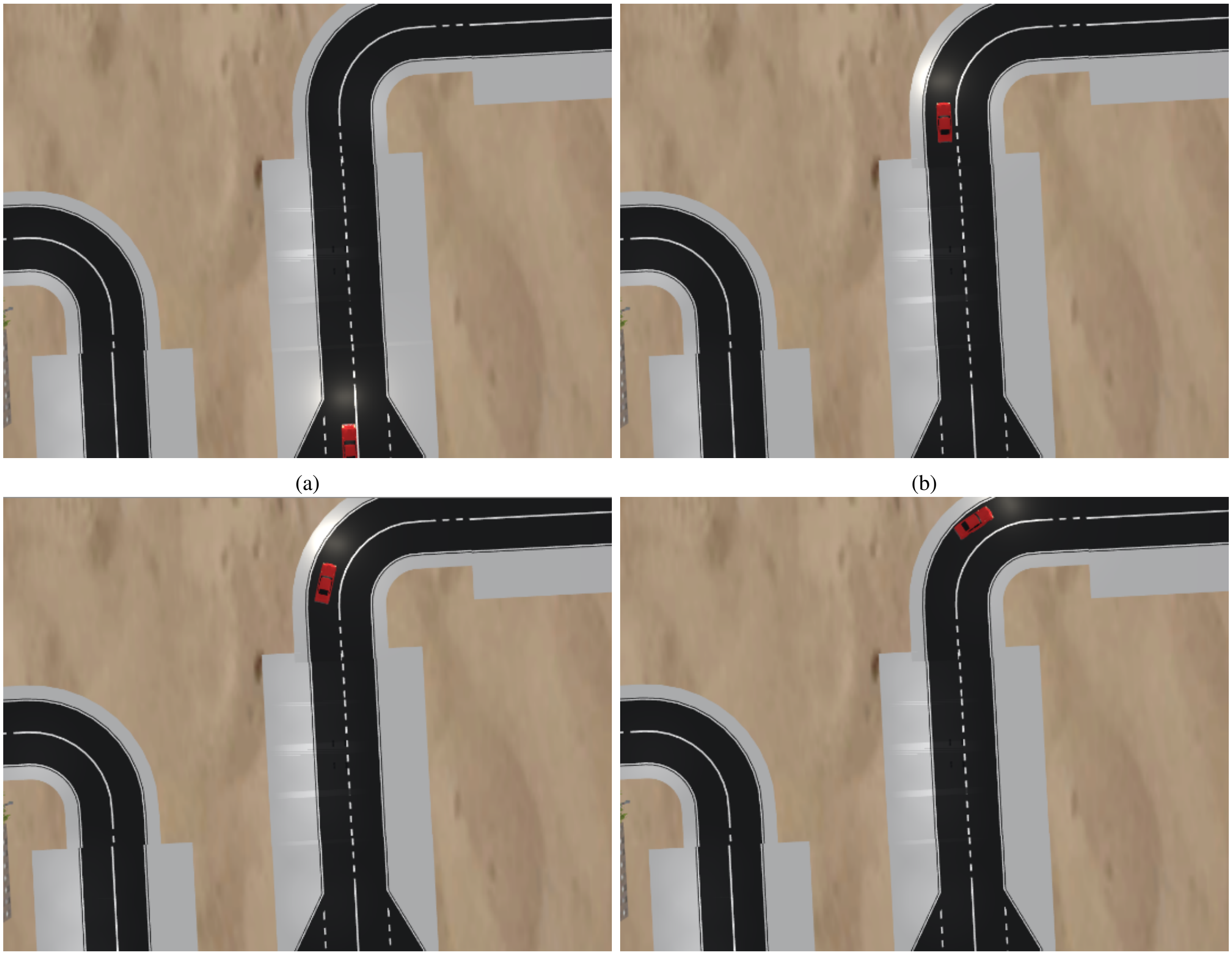
Using the Takagi–Sugeno system described in the previous section, it was implemented and described in Algorithm 1. Examples of screenshots from the car's movement on the surface are shown on Fig. 1.

V. Experiments

Proposed system was tested with car placed on the road with two lanes. The car was supposed to fit in the lanes at the turn. During the testing, different target distances were used to see which option would be most advantageous. The distant goal turned out to be ineffective, what can be seen in Fig. 2. The car left the road choosing the shortest path. Then the target distances were reduced and set at a distance of $1 - 4$ unit in UNITY. The effect for one field was the best, what is visible on Fig. 3.

VI. Conclusions

In this paper, the idea of using the Takagi-Sugeno system to make movement of objects in games. The simplicity of implementation and creation of rules means that such systems are accurate under additional conditions, as noted in the previous section. Setting close goals allows to move objects on arcs, which is the correct movement on the road. Unfortunately, this is not an ideal solution which is visible by finding new goals and setting them. However, it is a solution that does not require training as opposed to neural networks[17].



(a) (b) (c) (d)

Figure 3: The movement of the car depending on the destination located very close to the car on a black surface.

REFERENCES

[1] A. Winnicka, "The opponent's movement mechanism in simple games using heuristic method," *Symposium for Young Scientists in Technology, Engineering and Mathematics (SYSTEM 2018)*, pp. 12–16, 2018.

[2] D. Połap, M. Wozniak, C. Napoli, and E. Tramontana, "Is swarm intelligence able to create mazes?" *International Journal of Electronics and Telecommunications*, vol. 61, no. 4, pp. 305–310, 2015.

[3] P. Garca-Snchez, A. Tonda, A. M. Mora, G. Squillero, and J. J. Merelo, "Automated playtesting in collectible card games using evolutionary algorithms," *Knowledge-Based Systems*, vol. 153, no. C, pp. 133–146, 2018.

[4] O. E. David, N. S. Netanyahu, and L. Wolf, "Deepchess: End-to-end deep neural network for automatic learning in chess," in *International Conference on Artificial Neural Networks*. Springer, 2016, pp. 88–96.

[5] X. Zhao, Z. Ma, B. Li, Z. Zhang, and H. Liu, "Elm-based convolutional neural networks making move prediction in go," *Soft Computing*, pp. 1–11, 2018.

[6] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot *et al.*, "Mastering the game of go with deep neural networks and tree search," *nature*, vol. 529, no. 7587, p. 484, 2016.

[7] P. Liskowski, W. M. Jaskowski, and K. Krawiec, "Learning to play othello with deep neural networks," *IEEE Transactions on Games*, 2018.

[8] J. Föcker, D. Cole, A. L. Beer, and D. Bavelier, "Neural bases of enhanced attentional control: Lessons from action video game players," *Brain and Behavior*, p. e01019, 2018.

[9] J. Yao and N. Azam, "Web-based medical decision support systems for three-way medical decision making with game-theoretic rough sets," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 3–15, 2015.

[10] H. Zhang, J. Zhang, G.-H. Yang, and Y. Luo, "Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 152–163, 2015.

[11] Q. Kong, H. Sun, G. Xu, and D. Hou, "The general prenucleolus of n-person cooperative fuzzy games," *Fuzzy Sets and Systems*, vol. 349, pp. 23–41, 2018.

[12] T. Verma and A. Kumar, "Ambika methods for solving matrix games with atanassov's intuitionistic fuzzy payoffs," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 1, pp. 270–283, 2018.

[13] F. Li, P. Shi, C.-C. Lim, and L. Wu, "Fault detection filtering for nonhomogeneous markovian jump systems via a fuzzy approach," *IEEE Transactions on Fuzzy Systems*, vol. 26, no. 1, pp. 131–141, 2018.

[14] L. A. Zadeh *et al.*, "Fuzzy sets," *Information and control*, vol. 8, no. 3, pp. 338–353, 1965.

[15] F. Bonanno, G. Capizzi, G. L. Sciuto, and C. Napoli, "Wavelet recurrent neural network with semi-parametric input data preprocessing for micro-wind power forecasting in integrated generation systems," in *2015*

International Conference on Clean Electrical Power (ICCEP). IEEE, 2015, pp. 602–609.

[16] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE transactions on systems, man, and cybernetics*, no. 1, pp. 116–132, 1985.

[17] G. Capizzi, G. L. Sciuto, C. Napoli, and E. Tramontana, "Advanced and adaptive dispatch for smart grids by means of predictive models," *IEEE Transactions on Smart Grid*, vol. 9, no. 6, pp. 6684–6691, 2017.