TECHNICAL UNIVERSITY OF CLUJ-NAPOCA

ACTA TECHNICA NAPOCENSIS

Series: Applied Mathematics, Mechanics, and Engineering
Vol. 66, Issue II, June, 2023

A COMBINED NEURO-FUZZY SYSTEM TO COMMAND AN AUTONOMOUS AUTOMOBILE FROM VIRTUAL REALITY

Anca-Elena IORDAN, Florin COVACIU

Abstract: *The aim of this article consists in implementation of a combined neuro-fuzzy system used to control an autonomous automobile that need to ride in a labyrinth. The three proximity sensors attached at the vehicle are input for the artificial intelligence system and, as outputs, the intelligent strategy is characterized by angle of the front wheels of the autonomous automobile are turned, and automobile speed. Used methodologies were StarUML, C# language programming and opensource framework AForce.NET. The use of the combined neuro-fuzzy system compared to a fuzzy system is clearly superior in the control of the autonomous vehicle.*

Key words: *fuzzy logic, artificial neural network, automobile driving, virtual reality.*

1. INTRODUCTION

The goal of a driverless automobile rolling around on the streets looks inconceivable, but however we can be near to seeing similar automobiles on way round the world, due to artificial intelligence [1]. The automobile and technology industries have done important steps in artificial intelligence algorithms [2] usage over the past few years, replacing human beings with computers. Autonomous driving has the principal benefit in excess to supplant the humane chauffeur characterized by physical and emotive boundedness, the automobile capability to render predictions, but also commune with roads and other automobiles.

With the scope of control improving of the virtual autonomous automobile presented in paper [3], the purpose of this work is to present a combined system between fuzzy logic [4] and artificial neural network [5]. The paper structure is the following:

- Section 2 renders the related works.
- Section 3 presents the methodologies used to control the autonomous vehicle.
- The obtained results for automobile command by the instrumentality of artificial intelligence in section 4.
- Finally, the conclusions and references.

2. RELATED WORKS

One of the most significant provocations in realization of a control system for autonomous driving is the complicated interaction from the car and the environment. Some manners using artificial intelligence [6] have been used for the design of similar advanced system of vehicle control. Diverse strategies have been suggested as fuzzy logic [7], artificial neural networks, Kalman filter, and Petri nets with the scope to resolve this subject.

In [8] a methodology has been used to confirm the user experience in autonomous automobiles based on continuous information, objectives collected from physiological signals, while the user is immersed in a driving simulation based on virtual reality (VR). The automobile simulation is based on the GENIVI opensource platform. For the virtual driving experiment healthy people with a valid driving license have been recruited. Each person in the experiment wears HTC Vive headset, support for VR has been facilitated by the fact that GENIVI is based on the Unity 3D game engine, what naturally enables the building of VR applications for HTC Vive.

Camara et. al [9] started a study in which the authors applique game theory to command the

Received: 09.01.23; **Similarities**: 29.05.23: **Reviewed**: 22.05./17.05.23: **Accepted**:21.06.23.

interactions between autonomous automobiles and pedestrians, how they surpass for space on the way, while hoping to escape collisions. This game theory pattern was detailed just in virtual laboratory environments, and to be as real as possible, this investigation empirically reviews the comportment of pedestrians while passing the road in the attendance of autonomous automobiles approaching in multiple practical VR circumstances. Participants in this study were not knowledgeable regarding to behaviour of the virtual automobile, thereby they did not understand that it is an autonomous automobile and, so it has a theoretical comportment of the game.

The results in this study showed that the most participants did not expect autonomous automobile to turn off in some scripts and there was no exchange in their passing comportment. For the theoretical parameters of the game these outcomes supply some incipient estimates required for autonomous automobile in their passer interactions, and showing in general how these parameters can be deduced out of VR.

Wen et. al [10] suggest a script engendering pipeline concentrated on generating scripts in a distinctive zone immediately to an autonomous automobile. With this strategy, a script plane is engendered to ascertain the script simulation zone. To assess whether the chosen agents can engender a practical script, a scenario agent selector based on a convolutional neural network (CNN) is introduced, and to trigger an accident event, a collision event detector is introduced that manages the collision message. The command software has been implemented in Python and the VR environment was developed in Unity 3D, and the connection among the virtual environment and the command system was reached by transmission control protocol (TCP).

During the simulation no user intervention is required because the used script engendering pipeline may engender scripts including pedestrians, automobiles, and animals. The results show that the centred action dispatcher generated a real visual scenario, letting the agents around the event generate related actions, and the CNN-based script agent selector chose agents that insured practical scripts with 92.67% accuracy.

3. USED METHODOLOGIES

The design of the neuro-fuzzy software was made through AForce.NET opensource framework development in C# language programming [11]. This framework is composed of more libraries as: AForge.Fuzzy (fuzzy logic), AForge.Imaging (images processing), AForge.Genetic (genetic algorithms), AForge.Neuro (neural networks), AForge.MachineLearning (machine learning), and AForge.Robotics (robotics). In this project were used two libraries: AForge.Fuzzy [12] and AForge.Neuro [13].

AForge.Fuzzy library contains five interfaces, and fifteen classes as can be seen in UML package diagram shown in Figure 1.

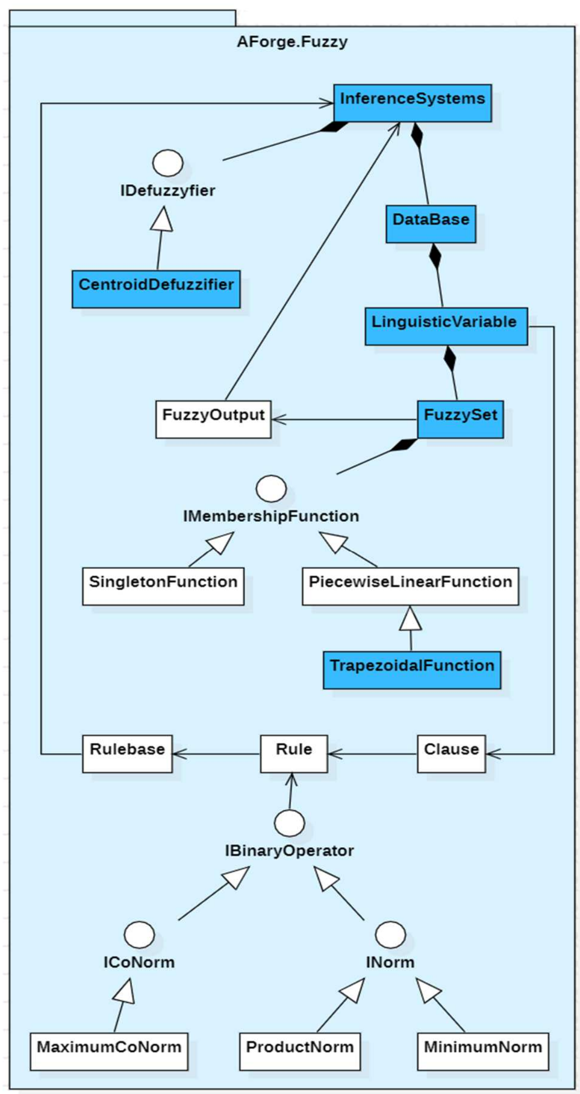

Fig. 1. UML package diagram: AForge.Fuzzy

This UML [14] package diagram was made using StarUML tool. From these 15 classes, the next 6 classes were used in this project:

- The class "InferenceSystem" constitutes a complete fuzzy inference system, with a data base, a rule base, a fuzzy output, and a defuzzification method to determine the numeric output.
- The class "Database" contains a collection containing software linguistic variables.
- The class "LinguisticVariable" represents the variables used in fuzzy systems that may keep a linguistic value (fuzzy set).
- The class "FuzzySet" is the foundation of all fuzzy hypothesis, representing a suite wherein parts are characterized by a degree, habitually between 0 and 1.
- The class "TrapezoidalFunction" is a membership function of the fuzzy sets with a trapezoid's shape.
- The class "CentroidDefuzzifier" is used in defuzzification process to calculate the centroid of the fuzzy output.

AForge.Neuro library contains one interface, twelve classes and a subpackage named AForge.Neuro.Learning as can be seen in the UML package diagram [15] shown in Figure 2. AForge.Neuro.Learning package contains two interfaces and five classes. From these seventeen classes, the next five classes were used in this project:

- The class "ActivationNeuron" computes weighted sum of the inputs, adds threshold value, and then applies activation function.
- The class "ActivationLayer" represents a layer of which is usually used in multi-layer neural networks.
- The class "ActivationNetwork" is a foundation for multi-layer neural network with activation functions.
- The class "BackPropagationLearning" implements backpropagation learning algorithm, which is widely used for training multi-layer neural networks with continuous activation functions.
- The class "SigmoidFunction" represents the sigmoid activation function with the expression (1). The parameter α determines steepness of the function. Increasing value of this property changes sigmoid to look more like a threshold function. Decreasing value of this property makes sigmoid to be very smooth.

$$f(x) = \frac{1}{1 + e^{-\alpha x}} \tag{1}$$

4. RESULTS

With the purpose the vehicle to be capable to rotate the front wheels to left or right with a particular angle of rotation, a combined neuro-fuzzy strategy was designed. The neuro-fuzzy system is characterized by three inputs since on vehicle front are bound three sensors: a front and two lateral sensors.

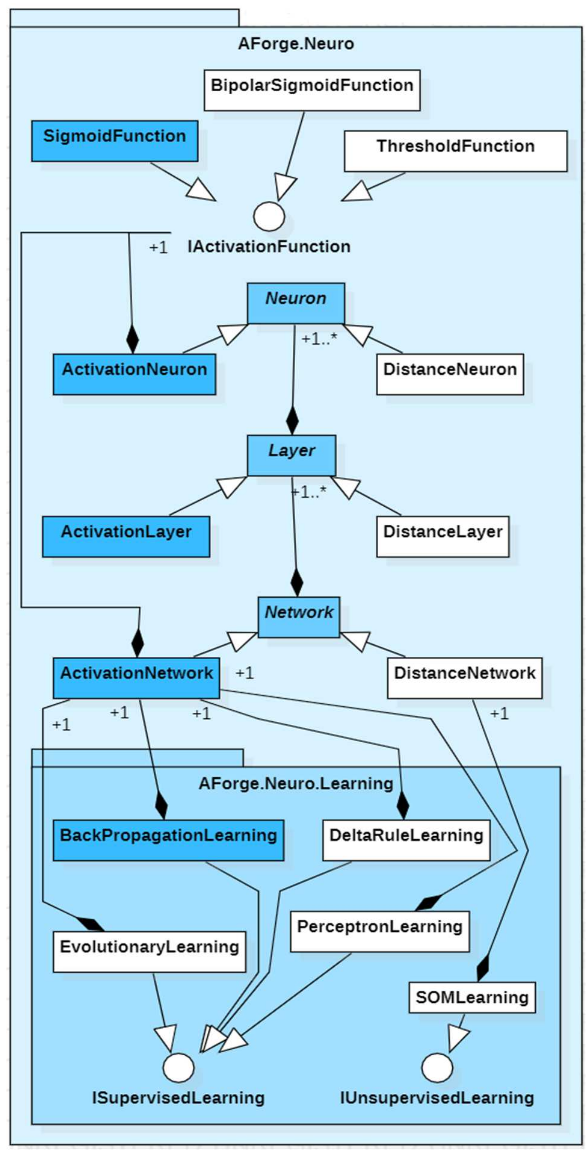

Fig. 2. UML package diagram: AForge.Neuro

Adequate to the lateral distances, in implementation phase of control system, three variables of “TrapezoidalFunction” type are instantiated, corresponding to the three fuzzy variables (Figure 3). Starting from these three instances of type “TrapezoidalFunction”, three variables of type “FuzzySet” are instantiated by using the composition relation. Then two "LinguisticVariable" objects are instantiated starting from the three “FuzzySet” objects. All the existing relations between these eight objects created with the scope to instantiate two objects of “LinguisticVariable” type that will represent two of the input data for the inference system that aims to obtain the rotation angle of the vehicle tires.

Corresponding to the front distance, in the implementation phase of the combined neuro-fuzzy system, three variables of “TrapezoidalFunction” type are instantiated corresponding to the three fuzzy variables as can be seen in object diagram represented in Figure 4. The three instances of “TrapezoidalFunction” type correspond to three variables of “FuzzySet” type instantiated by using the composition relationship. Then, corresponding to the front distance, an object of the "LinguisticVariable type is instantiated starting from the objects of "FuzzySet" type.

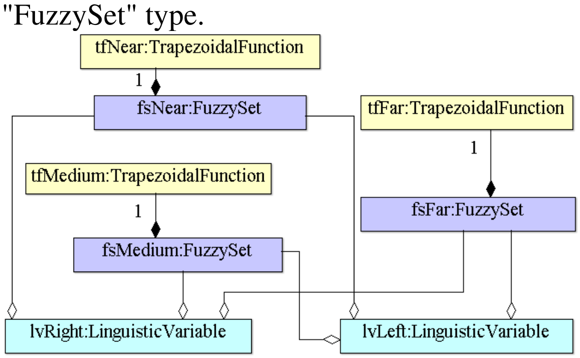

Fig. 3. Object diagram – Fuzzy sets for lateral distances

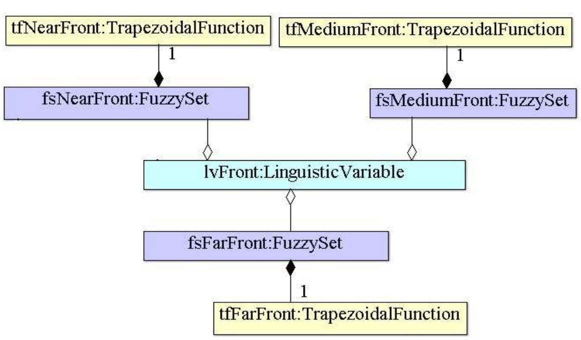

Fig. 4. Object diagram – Fuzzy sets for frontal distances

All existing relations between these seven objects created with the scope to instantiate an object of “LinguisticVariable” type that will represent the third input date for the inference system that aims to obtain the rotation angle of the vehicle tires. After instantiating the three linguistic variables that represent the input data of the inference system, rules are defined for determining the rotation angle of the vehicle wheels. Since we have three linguistic variables, and each linguistic variable is characterized by three fuzzy sets, 27 rules will be defined for the inference engine.

In the case of the result of neuro-fuzzy system associate to steering angle, seven variables of “TrapezoidalFunction” type are instantiated, corresponding to the seven fuzzy variables as can be seen in object diagram represented in Figure 5. Then, corresponding to the steering angle, an object of the "LinguisticVariable type is instantiated starting from the seven objects of "FuzzySet" type. After creating the three objects corresponding to the input data (lvFront, lvRight, lvLeft) and the object in which the output date will be stored (lvAngle), an object is created with the database that will be associated with the inference system used in determining of the angle. The object diagram which is showing the relationships between the seven objects used in the inference process is shown in Figure 6.

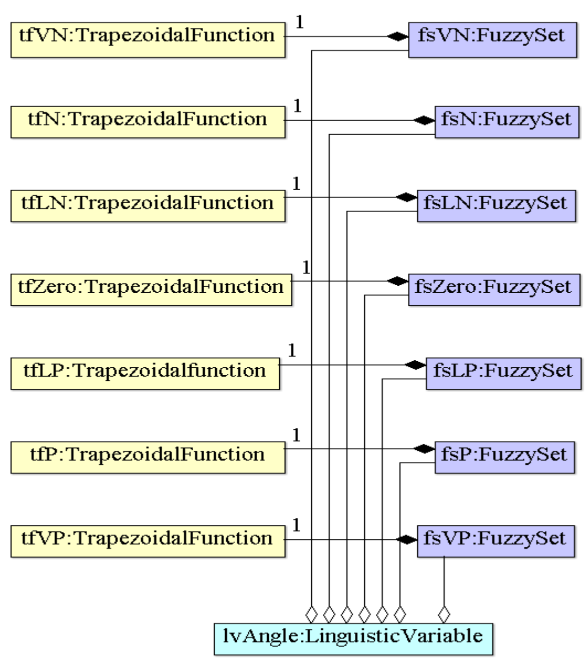

Fig. 5. Object diagram – Fuzzy sets for steering angle

To eliminate the incertitude whensoever frontal or lateral distance to the wall is is not clear, the fuzzy system was combined to a backpropagation learning method what is used for training on a large scale multilayer neural networks characterized by continuous activation functions. The UML diagram that shows the interactions between the objects used in this learning algorithm is presented in Figure 7. The classes corresponding to the instances in this diagram are part of the AForge.Neuro and AForge.Neuro.Learning packages. The inputs for this neural network are two objects of "LinguisticVariable" type and the output will be an object of the same type.

In order to highlight the performances of the combined neuro-fuzzy system in relation to the variant of the classic fuzzy system, over 100 crossing of the labyrinth were tested for three distinguished speeds: low, medium, and high. Number of collisions at medium speed is represented in Figure 8, and the number of collisions at high speed is represented in Figure 9. In case of medium speed, for the classic fuzzy system there were situations when the car hit the walls of the maze several times, but also situations when it never hit the walls of the maze. At medium speed the maximum number of collisions is 5.

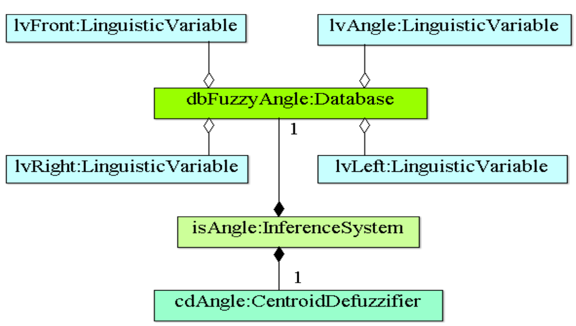

Fig. 6. Object diagram – Fuzzy inference system for steering angle

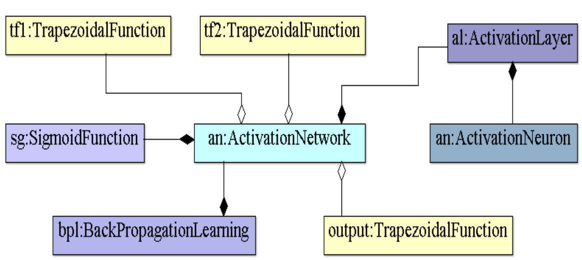

Fig. 7. Object diagram – Backpropagation Learning Algorithm

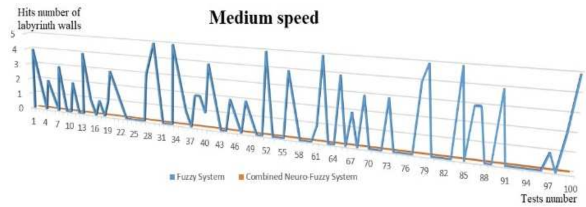

Fig. 8. Performances – medium speed

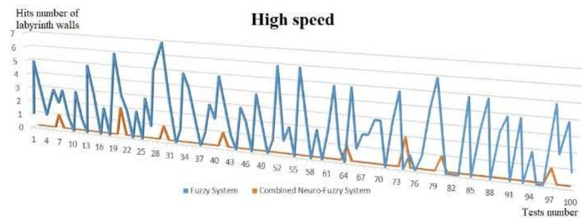

Fig. 9. Performances – high speed

Regarding the results corresponding to the combined neuro-fuzzy system was obtained that the car never hit the walls of the maze in case of low speed and medium speed. At high speed the maximum number of collisions is 7, and the collision rate increases when using the classic fuzzy algorithm. At the same high speed, in the case of combined neuro-fuzzy system, out of the 100 traverses of the labyrinth, only in 8 situations collisions appeared.

5. CONCLUSION

This study presents a combined neuro-fuzzy system to command an autonomous automobile using virtual reality. To escape the eventual collisions with the maze walls, the vehicle has three attached sensors to establish distances to walls. Distances are calculated by an intelligent module implemented using AForge.NET framework. The results of using this combined neuro-fuzzy system in cases of uncertainty are superior to the classic fuzzy system because the values determined for speed and angle lead to an optimal vehicle trajectory.

6. REFERENCES

[1] Panoiu, M., Panoiu, C., Iordan, A., Ghiormez, L., *Artificial neural networks in predicting current in electric arc furnaces*, IOP Conference Series: Materials Science and Engineering 2014, Vol. 57(1), 012011.

[2] Muscalagiu, I., Iordan, A., Muscalagiu, D.M., Panoiu, M., *Implementation and*

evaluation model with synchronization for the asynchronous search techniques, Recent Advances in Computer Engineering 2009, No. 13. pp. 211-216.

[3] Covaciu, F., *Development of a Virtual Reality Simulator for an Autonomous Vehicle*, Acta Technica Napocensis – Applied Mathematics, Mechanics, and Engineering 2022, Vol. 65(2), pp. 155-160.

[4] Cuntan, C., Baciu, I., *Fuzzy Algorithm for Steel Level's Control in the Mould at Continuous Casting*, Proceedings of the International Conference on Accomplishments in Electrical, Mechanical and Informatics Engineering, Banjaluka, Bosnia and Herzegovina, 2007, pp. 385-390.

[5] Covaciu, F., Iordan, A.E., *Control of a Drone in Virtual Reality using MEMS Sensor Technology and Machine Learning*, Micromachines 2022, Vol. 13(4), pp. 1-19.

[6] Iordan, A.E., *Usage of Stacked Long Short-Term Memory for Recognition of 3D Analytic Geometry Elements*, Proceedings of the 14th International Conference on Agents and Artificial Intelligence 2022, Vol. 3, pp. 745-752.

[7] Panoiu, C., Panoiu, M., Rat, C., Rob, R., *The Computer Assisted Parking of a Bluetooth Controlled Car using Fuzzy Logic*, Advances in Intelligent Systems and Computing 2016, Vol. 356, pp. 51-66.

[8] Morra, L., Lamberti, F., Prattico, F.G., Rosa, S.L., Montuschi, P., *Building Trust in Autonomous Vehicles: Roles of Virtual Reality Driving Simulators in HMI Design*, IEEE Transactions on Vehicular Technology 2019, Vol. 68(10).

[9] Camara, F., Dickinson, P., Fox, C., *Evaluating Pedestrian Interaction Preferences with a Game Theoretic Autonomous Vehicle in Virtual Reality*, Transportation Research 2021, 78.

[10] Wen, M., Park, J., Cho, K., *A Scenario Generation Pipeline for Autonomous Vehicle Simulators*, Human-centric Computing and Information Sciences 2020, Vol. 10(24).

[11] Iordan, A.E., *Optimal Solution of the Guarini Puzzle Extension using Tripartite Graphs*, IOP Conference Series: Materials Science and Engineering 2019, Vol. 477(1).

[12] Gong, C., He, H., *Research of AForge.NET in Motion Video Detection*, Applied Mechanics and Materials 2014, Vol. 508, pp. 496-500.

[13] Verma, S., Pillai, P., Hu, Y., *Development of an Eye-Tracking Control System using AForge.Net Framework*, International Journal of Intelligent Systems Technologies 2012, Vol. 11, pp. 286-303.

[14] Iordan, A.E., Covaciu, F., *Improving Design of a Triangle Geometry Computer Application using a Creational Pattern*, Acta Technica Napocensis - Applied Mathematics, Mechanics, and Engineering 2020, Vol. 63(1), pp. 73-78.

[15] Iordan, A.E., *Development of Interactive Software for Teaching Three-Dimensional Analytic Geometry*, Mathematics and Computers in Science and Engineering 2009, Vol. 9, pp. 23-28.

Sistem combinat neuro-fuzzy pentru controlul unui vehicul autonom in Realitate Virtuala

Rezumat: Scopul acestui articol constă în implementarea unui sistem combinat neuro-fuzzy utilizat pentru a controla un vehicul autonom care trebuie să treacă printr-un labirint. Cei trei senzori de proximitate atașați vehiculului sunt de intrare pentru sistemul de inteligență artificială și, ca ieșiri, strategia inteligentă are unghiul la care sunt rotite roțile din față a vehiculului autonom și viteza vehiculului. Metodologiile utilizate au fost: StarUML, limbajul de programare C# language și framework-ul AForce.NET. Utilizarea sistemului combinat neuro-fuzzy comparativ cu un sistem fuzzy este net superioară în controlul vehiculului autonom.

Anca-Elena IORDAN, PhD, Lecturer, Technical University of Cluj-Napoca, Computer Science Department, anca.iordan@cs.utcluj.ro, Barițiu street no. 28, Cluj-Napoca 400027, ROMANIA.

Florin COVACIU, PhD Eng., Associate professor, Technical University of Cluj-Napoca, Department of Design Engineering and Robotics, florin.covaciu@muri.utcluj.ro, Muncii Blvd. no. 103-105, Cluj-Napoca 400641, ROMANIA.