Fuzzy logic techniques for Cybercars: a control and decision approach.

Michelena Oriana Rojas, Joshué Pérez Rastelli, Fawzi Nashashibi

▶ **To cite this version:**

Michelena Oriana Rojas, Joshué Pérez Rastelli, Fawzi Nashashibi. Fuzzy logic techniques for Cybercars: a control and decision approach.. 22èmes rencontres francophones sur la Logique Floue et ses Applications (LFA 2013), Oct 2013, Reims, France. hal-00913106

HAL Id: hal-00913106

https://hal.inria.fr/hal-00913106

Submitted on 3 Dec 2013

Fuzzy logic techniques for Cybercars: a control and decision approach.

O. Rojas Michelena[1]   J. Pérez Rastelli[1]
F. Nashashibi[1]

[1] IMARA team at INRIA research center, Paris – ROCQUENCOURT, oriana.rojas_michelena@inria.fr, joshue.perez_rastelli@inria.fr, Fawzi.Nashashibi@inria.fr

Abstract:

In the context of Intelligent Transportation Systems (ITS) one of the aims is to reach autonomous vehicle capabilities based on human driver experiences in different situations. This problem can be treated from two points of view: by tracking a reference (curve lines -lateral control- or speed -longitudinal control-) and by the decision approach (in specific or dangerous situations). In this paper, fuzzy logic techniques have been implemented in real time control tools to translate human knowledge to driverless control processes, considering risk/warning situation. A comparison with previous works (based in classic control laws) for driving, was carried out in urban areas. Moreover, a new approach to give the driver, a reference speed when the vehicle is arriving to a traffic light intersection was developed. Some simulations show that fuzzy logic techniques are promising in the development of ITS applications.

Keywords:

Autonomous Vehicle, Fuzzy Logic Controllers, Intelligent Transportation Systems, Cybercars, Speed Control, Lateral Control tracking

1   Introduction

Autonomous driving is one of the most expanding topics in the ITS field, because it is directly beneficial to drivers. For this reason, different research centres and manufacturers around the world are working together developing solutions for driver assistant systems, intelligent infrastructure and autonomous vehicles.

Different approaches have been developed in order to control autonomous vehicles in urban and highway scenarios. The control of a dynamic system, as real vehicles, it is not a simple task because of the complexity of the system modelling and the tuning process. In this context, intelligent control techniques offer powerful methods for the control of autonomous vehicles [1][7][5].

Fuzzy logic translates human knowledge for the driverless control process, by rule bases and membership input and output functions, for tracking (control) and decision (risk/warning situation) point of view.

This approach is framed in the IMARA [13] team goals, which are focused on ITS researches, especially in autonomous driving systems (Cybercars). In this work, the focus is the control and decision stages [2]. A new fuzzy tool able to control the lateral and longitudinal actions is proposed.

The rest of this work is organized as follows. In section 2, a summary of some of previous fuzzy logic approaches, as well the proposed in this work are presented. *Lateral control tracking* and *speed reference* in traffic light intersections are described in section 3 and 4. Section 5 explains the simulations carried out with RTMaps [15] and ProSiVIC [17]. Finally, the conclusions and future works are presented in section 6.

2   Fuzzy logic approach

Fuzzy logic techniques have been widely implemented in different industrial process in the last decade [4][12]. For this reason, many libraries, mainly developed in C++, are easily found in the literature.

Conventional controllers frequently use differential equations to describe the system behavior. Sometimes, this information is incomplete because some assumptions in the modeling process. For this reason, fuzzy controllers are an interesting alternative, because they use the expert knowledge (in our case the driver), which can be represented in natural languages.

Specifically in autonomous driving applications, the experimental fuzzy coprocessor, called ORBEX, was used by the Autopia [14] team in different situations as follows: lateral and longitudinal control, overtaking with three vehicles, intersections, merging, car racing simulations, among others [6][7][1].

In 2012, this library was updated, improving its previous performance in terms of computing time, structure and adding new membership input and output functions [1].

The goal is to achieve the autonomous driving of the vehicle using simple sentences defined in a rule base. Then, it is necessary to define the input and output membership functions [8].

2.1 Real time implementation

Regarding to real time simulations, two algorithms were developed in different situations (lateral control and longitudinal speed reference), using RTMaps and Microsoft Visual Studio [16] softwares.

RTMaps is a multitasking environment that allows embedded systems and its applications to interact with multi-task processes. It has a user-friendly graphical interface where different modules can be connected for real time applications [10]. Some modules are predefined or programmed in C++ with Microsoft Visual Studio.

In this work, two modules based on fuzzy logic libraries were created. One of them was developed in order to compare the classic controller of a previous work with a fuzzy controller to improve the lateral control tracking presented in [2]. Moreover, another module to warn speed references at intersections with traffic lights was proposed. The idea was that the vehicle is able to know at which speed it must travel to avoid abrupt braking and save fuel.

V2I (Vehicle to Infrastructure) communications are considered in this last module. This allows the vehicle to know the position of the infrastructure and the traffic light time (time to red and time to green) up to 200 meters before arriving to the intersection. This information is used by the vehicle to achieve a better increasing, decreasing or maintaining of the reference (recommended) speed, or even stop slowly and wait for the next green light or the next *green wave* if it is necessary.

3 Lateral control tracking

For the fuzzy controller applied to lateral control, two input variables were considered, as proposed in [8], the "Heading Error" and the "Lateral Error", where:

Heading error: it is the angle between the vehicle direction and the predetermined trajectory, measured in radians (Figure 1).

Lateral error: is the deviation of the front of the vehicle from that same predefined route (Figure 1).
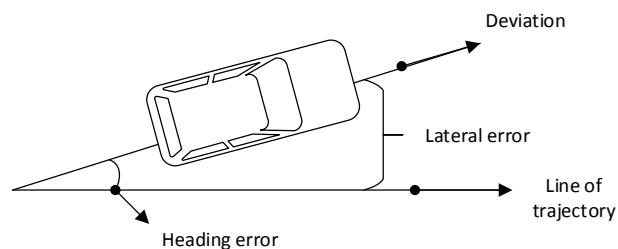

Figure 1– Input variables for the fuzzy controller.

Figure 1 shows the input variables. The surface control is showed in Figure 2. Moreover, the curvature is also considered in control law, as proposed in [2]. This variable is calculated in each segment of the path. Then, the output of the fuzzy controller is added at the end of the *defuzzification* process.

The rules used in the lateral steering control are described as follows [8]:

IF *Lat_error* Right THEN *Steer_Pos* Left
IF *Lat_error* Center THEN *Steer_Pos* Center
IF *Lat_error* Left THEN *Steer_Pos* Right
IF *Head_ error* Right THEN *Steer_Pos* Left
IF *Head_error* Center THEN *Steer_Pos* Center
IF *Head_error* Left THEN *Steer_Pos* Right

The *Steering Position* is the output variable of the controller and has three *singletons* (*Left* (0.5), *Right* (-0.5) and *Center* (0)). Each *singleton* represents the basic positions of the steering wheel.

The inference method (*center-of-area method*) uses each output variable according to each linguistic label, as proposed in [5]. The fuzzy controller allows to write the rules in an almost natural language, so if the controller read that the vehicle is coming out of the path, it orders the vehicle to steer in the opposite direction.
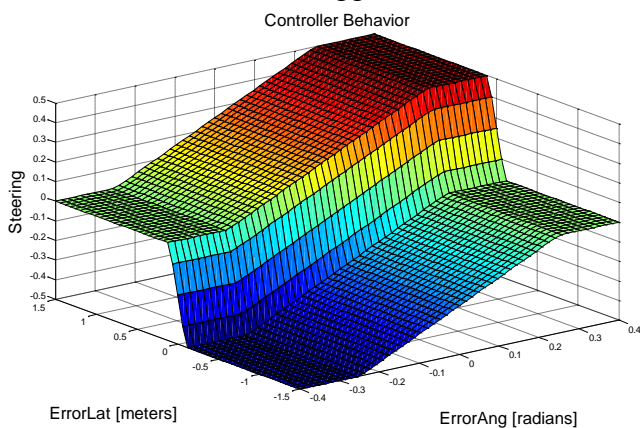

Figure 2 – Surface Controller Behavior

4 Speed reference in traffic light intersections

For this application it was necessary to use three input variables, which are the traffic light times, red light, green light and the distance to interception (DTI) (Figure 3).
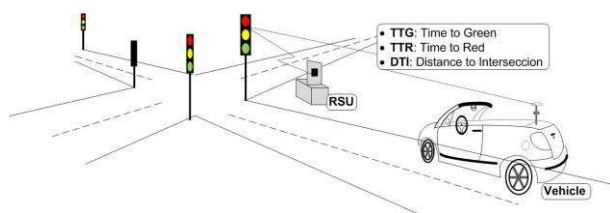

Figure 3 – Input variables for the speed reference fuzzy controller

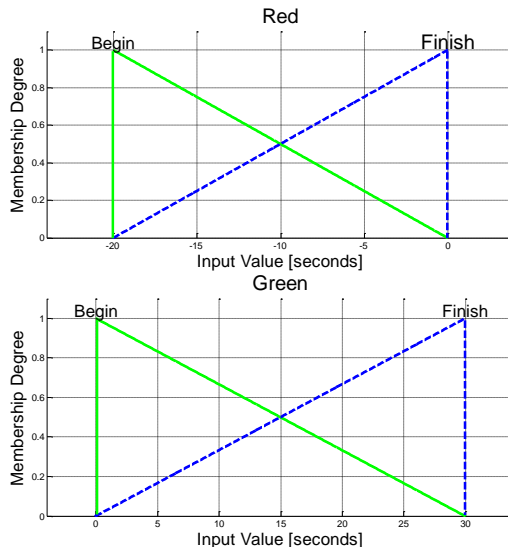Figure 4 and Figure 5 shows the membership functions for each input variable.

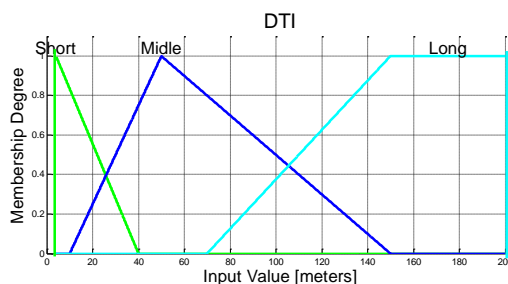Figure 4 – Membership functions for the traffic light time, red and green respectively

Figure 5 – Membership function for the distance to interception (DTI)

Two variables are used for the traffic light (Red and Green), where each has defined two membership functions completely symmetrical covering all the possible inputs. In this application the time cycle of the lights are 30 seconds for green and 20 seconds for red. The values of input membership functions were defined considering these times.

The reason to use two different variables for green light and red light is that the algorithm considers them as two principal cases; this approach is based on the scenarios described in [9], as follows:

- Scenario 1: maintain speed
- Scenario 2: accelerate and overtake
- Scenario 3: reduce speed and overtake
- Scenario 4: brake and wait for the next green

The DTI membership function (Figure 5) gives more weight to the distance when the vehicle is closer at the intersection. In this situation, the vehicle can be inside the "short" or the "middle" label, because in these cases the response has to be faster than in the case where the vehicle is in the "long" label.

The cross rule base, based on driver knowledge when is arriving to an intersection, is defined as follows:

IF *Green* Begin AND *DTI* Short THEN *Acc* MidAcc
IF *Green* Begin AND *DTI* Middle THEN *Acc* Acc
IF Green Begin AND *DTI* Long THEN *Acc* Keep
IF *Green* Finish AND *DTI* Short THEN *Acc* Brake
IF *Green* Finish AND *DTI* Middle THEN *Acc* MidBrake
IF *Green* Finish AND *DTI* Long THEN *Acc* Keep

IF *Red* Begin AND *DTI* Short THEN *Acc* Brake
IF *Red* Begin AND *DTI* Middle THEN *Acc* MidBrake
IF *Red* Begin AND *DTI* Long THEN *Acc* BrakeFew
IF *Red* Finish AND *DTI* Short THEN *Acc* Few
IF *Red* Finish AND *DTI* Middle THEN *Acc* Few
IF *Red* Finish AND *DTI* Long THEN *Acc* Keep

Here the *inference method* is the same as proposed in [5][8]. The *singletons* represent accelerations and decelerations, selected to improve the drivers comfort and avoid abrupt velocity changes [11].

To obtain the reference speed, two equations, based on the uniformly accelerated rectilinear motion, were used. The first equation is used for accelerate/maintain, and the other for decrease/break (Eq.1 and Eq. 2).

$$V = at + Vo \quad (1)$$

$$V = sqrt(2da) \quad (2)$$

"*Vo*" is the initial speed at the first iteration, and then it is the current speed; "*d*" is the DTI variable; "*t*" is the time to interception; and "*a*" is the acceleration given by the controller.

5   Experimentation and results

5.1 ProSiVIC and RTMaps

The algorithms for both applications were tested and validated in a virtual simulation environment with ProSiVIC (Connected to the RTMaps modules). This software offers a multisensory platform and provides the possibility to work with different parameters of a real vehicle, such as longitudinal and lateral speeds, steering wheel response, pitch, and roll and yaw angles, weather conditions, and friction coefficients, among others. [2]

The ProSiVIC platform allows synchronization of many variables. Among these are camera viewports, steering position, simulation time and acceleration.

5.2 Lateral Control Tracking Validation

Figure 6 and Figure 7 show the results regarding the fuzzy and classic controller. Specifically, Figure 6 shows the superposition of the lateral and heading error for each control. The heading errors graphic is similar for both controllers, because values of maximum deviation in the turning inside the roundabout are approximately 0.3 rad.

However the most important behaviour in this graphic is the lateral error result, where the difference is actually remarkable, since the maximum lateral error introduced with the fuzzy controller is only the 10% of the error obtained with the classic control.

It should be noted that this test was made considering the same scenario and conditions, with low speed (no more than 30 km/h) and simulated in a roundabout.
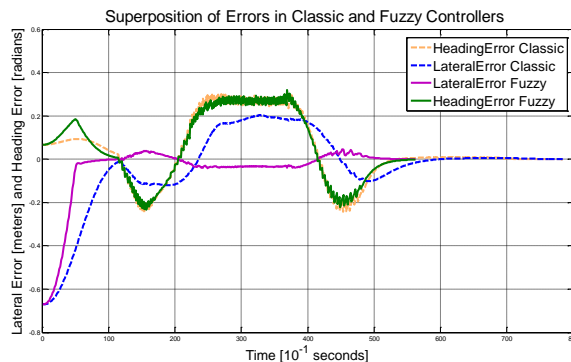

Figure 6 – Lateral and heading errors for classic and fuzzy controllers

In the Figure 7 the steering output result for both controllers are shown. It can be appreciated that the decision in both

controllers are similar. The difference is at the beginning, where the fuzzy control is more drastic, but in the roundabout the fuzzy response looks more stable than the "Steering with the classic controller" response. In general the peak values are the same, but this was an expected behavior, after all the difference between lateral errors is only 0.18 m.
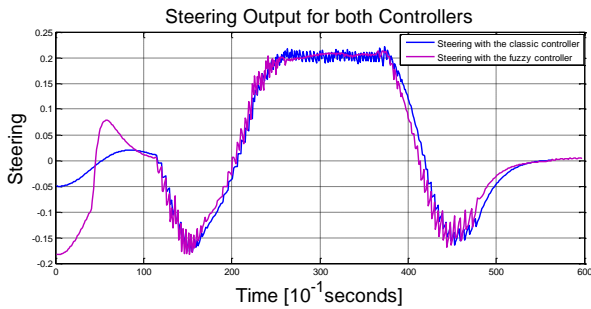

Figure 7 – Steering output of controllers, the classic and the fuzzy control

5.3 Speed Reference Tool Validation

For this experiment, several parameters have to be shown, e.g.: distance to interception, traffic light time, time to interception and current speed.

Here two simulations were performed; one for long distance (126m) and one for short distance (22m), in order to evaluate the fuzzy controller speed reaction.

Figure 8 shows how the parameters move when the vehicle is approaching the intersection. The distance to interception decreases, and the time to intersection is directly proportional to the current speed. In Figure 9, the controllers response with the traffic lights (dotted lines) can be appreciated. In this figure the real speed represents the decision taken by the driver. The purple and blue lines are the recommended speed given by the tool with the fuzzy and classic controller, respectively.
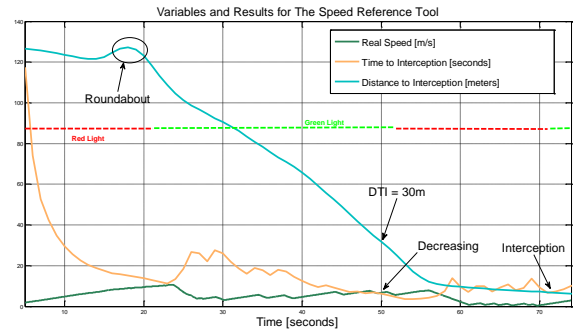

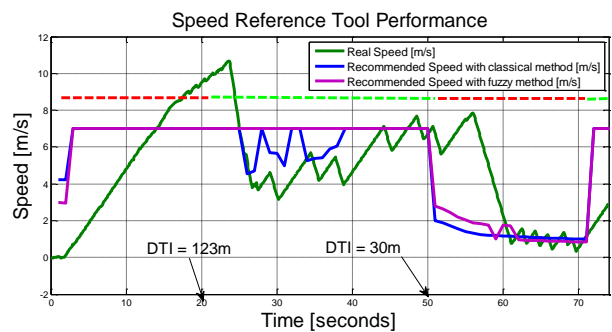Figure 8 – Speed reference tool variable behaviors for the intersection problem in long distance

Figure 9 – Speed reference tool performance for long distance (126m)
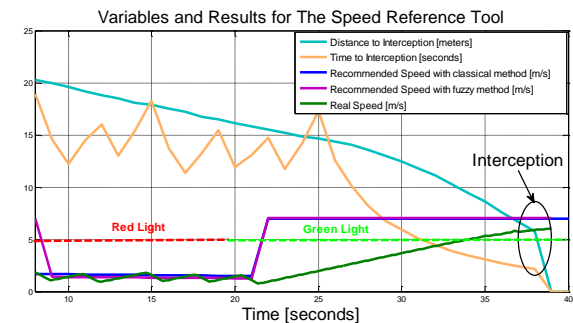

Figure 10 – Speed reference tool performance for the intersection problem in short distance

Figure 10 and Figure 11 the same experiment was done, but with a shorter distance. The results for both controllers were similar. The difference was the simplicity in the tuning process of the fuzzy controller.
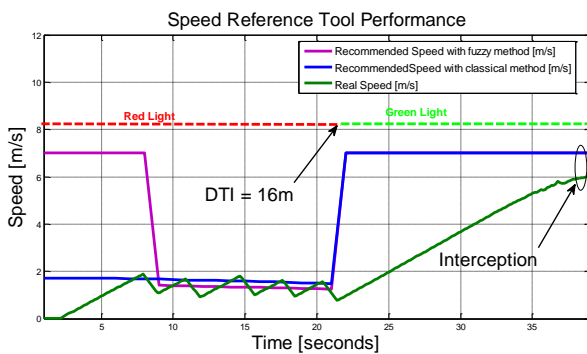
Figure 11 – Speed reference tool performance
for short distance (22m)

Figure 11 shows the current and recommended speed. The first one doesn't follow exactly the recommended by the module; this is due to the lack of capacity in the graphic card for the simulation software (ProSiVIC) in our computer, thereby altering the different variables. However, this behavior is useful, because the controller does responds to inaccurate speed inputs, analogous to a real vehicle driver behavior.

6   Conclusions and future works

This work describes two ways to use fuzzy logic techniques to improve vehicle maneuvers. The behavior of a human driver was emulated, both cases: steering wheel and a reference longitudinal speed in traffic lights intersections.

These controllers worked very well, even exceeded expectations, improving previous works, and giving an easy way to translate human knowledge in the driving process. For future works other algorithms based on neuro-fuzzy systems will be considered, which are able to learn from a human driver.

References

[1] E. Onieva, J. Godoy, J. Villagrá, V. Milanés, J. Pérez, *On-line learning of a fuzzy controller for a precise vehicle cruise control system*, Expert Systems with Applications, Volume 40, Issue 4, March 2013.

[2] Gonzalez, D. y Pérez, J. *Control architecture for Cybernetic Transportation Systems in urban enviroments*, 2013.

[3] Lopez, H. (2012). *Tool for Design of Fuzzy Controllers for Automatic Driving Vehicles.* Degree Thesis. Universidad Simon Bolivar, Sartenejas, Venezuela.

[4] M. Sugeno and M. Nishida, "*Fuzzy control of model car*", Fuzzy Sets Syst., vol. 16, no. 2, pp. 103–113, Jul. 1985.

[5] Mamdani, E. *Application of fuzzy algorithms for control of a simple dynamic plant. Proc. Inst. Elect. Eng.*, vol. 121(12), páginas 1585_1588, 1974.

[6] Milanés, V., Pérez, J., Onieva, E., González, C., & de edro, T. (2010). *Lateral power controller for unmanned vehicles*. Przeglad Elektrotechniczny, Electrical Review, 86, 207–211.

[7] Naranjo, J. E., Gonzalez, C., Garcia, R., & de Pedro, T. (2008). *Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver*. IEEE Transactions on Intelligent Transportation Systems, 9, 438–450.

[8] Pérez, J., Milanés, V. y Onieva, E. *Cascade architecture for lateral control in autonomous vehicle. Intelligent Transportation Systems*, IEEE Transactions on, vol. 12, NO. 1(6), páginas 73-82, 2011.

[9] Rakha, H., Kamalanathsharma, R. and Ahn, K. *Eco-Vehicle Speed Control at signalized intersections using I2V communication*. 2012

[10] RTMaps, version 4.x, *User's manual*, Intempora S.A, 2011.

[11] Solea, R. and Nunes U. *Trajectory Planning and Sliding-Mode Control Based Trajectory for Cybercars*. 2007.

[12] T. H. S. Li and S.-J. Chang, "*Autonomous fuzzy parking control of a carlike mobile robot*," IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 33, no. 4, pp. 451–465, Jul. 2003.

[13] http://imara.inria.fr/

[14] http://www.car.upm-csic.es/autopia/equipo.php

[15] www.intempora.com

[16] www.microsoft.com/visualstudio

[17] www.civitec.com