Cooperative Throttle and Brake Fuzzy Control for ACC+Stop&Go Maneuvers

José E. Naranjo, Carlos González, *Member, IEEE*, Ricardo García, and Teresa de Pedro

***Abstract*—The goal that a car be driven autonomously is far in the future and probably unreachable, but as a first step in that direction, adaptive cruise control (ACC) and Stop&Go maneuver systems are being developed. These kind of controllers adapt the speed of a car to that of the preceding one (ACC) and get the car to stop if the lead car stops. This paper presents one such system and related experiments performed on a real road with real cars. The driving system gets its input via an RTK DGPS device and communicates its positions to one another via a wireless local area network link. It outputs signals controlling the pressure on the throttle and brake pedals. The control system is based on fuzzy logic, which is considered best to deal with processes as complex as driving. Two mass produced Citroën Berlingo electric vans have been instrumented, providing them with computer controlled actuators over the brake and the throttle to achieve human-like driving. The results of the experiments show that the behavior of the vehicles is very close to human and that they adapt to driving incidences, increasing the safety of the driving and permitting cooperation with manually driven cars.**

***Index Terms*—Fuzzy control, global positioning system, intelligent control, road vehicle control, wireless local area network (WLAN).**

I. Introduction

AUTOMATIC vehicle speed control is presently one of the hottest research topics throughout the automotive industry as a whole [1] and particularly in the intelligent transportation systems field [2]. The goal of such automation is to improve the safety of the occupants of the car by relieving the human driver of tedious tasks that could lower attention, as well as to make the traffic flow more efficient [3]. Cruise control (CC) systems, with the capability of maintaining a user preset speed, were the first step in this direction. The next step was adaptive CC (ACC) systems, which add the capability of keeping a safe distance from the preceding vehicle to CC [4]. Both systems are now on the market, and several cars come equipped with them [5]. Highways are the most common scope of applicability of such systems [6], [7].

The main limitation of conventional ACC systems is that they do not manage speeds under 30 km/h because only the throttle is used for this task and, consequently, they are inhibited in traffic jams or urban driving. Extensions of ACC with Stop&Go capability are being researched to overcome this drawback [8]. Stop&Go driving is a typical maneuver in city streets. Throttle and brake pedal automation is needed to install this feature in a vehicle. This feature makes ACC useful in urban driving and dense traffic situations, when it is all the more necessary for preventing and averting rear-end collisions and major accidents [15].

ACC research started in Europe with the PROMETHEUS project (1986–1994) that involved several European car manufacturers. This project used scanning radars, and vehicle automation was limited to throttle pedal control only. Mitsubishi was the first firm to introduce ACC in its Diamante model in 1995. In Europe, Daimler–Chrysler launched the Distronic ACC S-Class in 2000. The current focus of research into speed control is on augmenting the ACC systems with Stop&Go capabilities. This requires the collection of fine data, which can be acquired from several sensors, such as radar [9], laser, vision [10], [11], or a combination of the three [12]. Another way to input data into embedded vehicle systems is, as in our case [13], wireless communication, which provides a lot of system information (even more than could be acquired using the car's own sensors [14]).

The procedures for controlling both pedals can be based on hard mathematical models of a complete autonomous longitudinal system [16]. Also, artificial intelligence (AI) techniques can be used to emulate human driving. Neural networks [17], genetic algorithms [18], or fuzzy logic [19]–[21] are some of the available AI tools for automating driving jobs.

The use of fuzzy logic for control systems has two main features. The first of one is that these kinds of controllers do not need an exact mathematical model of the system to control. This characteristic is very important when we have to deal with difficult-to-linearize systems such as cars. Fuzzy logic avoids using approximate models that are very complex and lowly efficient if they are very realistic or lowly realistic when they are efficient. The second feature of the fuzzy control is that it does not pretend to use the mathematical representation of the systems but to emulate the behavior of the human driver and his experience, mimicking his reactions. It also permits adding to the system the subjective knowledge of the users, which is certainly a very useful characteristic for emulating human behavior [22].

Manuscript received April 14, 2004; revised August 2, 2004, January 14, 2005, May 6, 2005, May 11, 2006, July 19, 2006, and September 5, 2006. This work was supported in part by the Spanish Ministry of Education under Grant ISAAC CICYT DPI2002-04064-C05-02, by the Spanish Ministry of Public Works under Grant COPOS BOE 280 22/11/2002, and by the Res. 22778, Citroën España S.A. under Contract "Adquirir nuevos conocimientos sobre la introducción de las tecnologías de la información en el mundo del automóvil y para difundirlos en los ámbitos científicos, empresariales y comerciales (AUTOPIA)," and by Cybecars-2 Project UE-STREP 28062, 6th Framework Programme, 2006. The review of this paper was coordinated by Dr. M. S. Ahmed.

The authors are with the Industrial Computer Science Department, Instituto de Automática Industrial (CSIC), 28500 Madrid, Spain (e-mail: jnaranjo@iai.csic.es; gonzalez@iai.csic.es; ricardo@iai.csic.es; tere@iai.csic.es).

Digital Object Identifier 10.1109/TVT.2007.897632

In this paper, we present an ACC system extended with Stop&Go capabilities. It is based on fuzzy logic, installed in real vehicles, and tested on real roads. This paper deals with real experiments, and thus, it is greatly conditioned by our facilities. Currently, our longest straight-line path is about 300 m, which do not permit high speed over an extended period of time. The experiments here included show speeds of 20 km/h.

II. Cooperative Throttle and Brake Fuzzy Control

This paper is a part of AUTOPIA project of the Instituto de Automática Industrial (IAI), whose objective is to develop an automatic car. The very name of the project implies that we consider this goal to be out of our reach in the near future. The used approach is to use fuzzy logic to emulate the behavior of human drivers. Our group has been working on fuzzy logic for many years, and we consider it an appropriate tool for control applications, taking into account that people have always controlled processes whose mathematical models are not known (i.e., driving a car).

We have already developed an ACC system extended with stop and go capability [23]. First, we used the throttle only and then we included the brake [24]. The aim of this paper is to describe a global system, named Stop&Go+ACC, in which the two speed actuators (throttle and brake) act cooperatively. Besides, we explain how the inclusion of the brake affects the previous throttle-only ACC.

The usual sequence of driver actions for slowing down to match the speed of a preceding slow car is as follows.

1) Step off the throttle.
2) Use the engine brake.
3) Step on the brake when the headway is not reduced fast enough.

These actions are implemented as fuzzy rules for the Stop&Go+ACC system, achieving car control even if the preceding car brakes.

In general terms, the Stop&Go+ACC will involve joining the brake and throttle controllers, although a tuning up procedure will be necessary to synchronize and permit throttle-brake cooperation. First, we will review the existing controllers, and then, we will explain the changes needed to implement the humanlike actions in the control system.

A. Fuzzy ACC

This paper is based on a computational model of a fuzzy coprocessor, named ORBEX [25], that we had previously developed at the IAI. ORBEX can write fuzzy rules with information supplied by experts in a near natural language. For instance:

if *speed_error* **more than** null **or** *acceleration* **more than** null **then** *accelerator* up

Where the words in italics are fuzzy variables [26], [27], the words in bold are ORBEX language key words, and the words in plain script are linguistic values of the variables. The variables to the left of the term then are input variables, and the variables to the right are output variables.

The values of the fuzzy variables are taken from a set of fuzzy partitions, represented by membership functions of a variety of shapes—triangular, trapezoidal, Gaussian, ... , or even singletons [28]. The designer sets the membership function shape by experience [22]. In our system, all output membership functions are singletons, so (1), shown in the following, calculates the crisp value y_{out} of an output variable y:

$$y_{\text{out}} = \frac{\sum_i w_i y_i}{\sum_i w_i} \tag{1}$$

where w_i represents the weight of the i-rule, and y_i is the value of the output y inferred by the i-rule. The weight of a rule represents its contribution to the global control action (calculated as the minimal degree of current crisp input value membership of its respective fuzzy partitions).

B. System Architecture

Formally, a distinction can be made between four vehicle speed control layers in manual driving: a mechanical layer (the pedals and all its associated physical mechanisms), an actuation layer (the human foot that steps on the pedal), a sensorial layer (the human senses) that allows the driver to obtain information about the vehicle and the environment, and a reasoning layer, that is, the human brain.

The cars used in our research are electrical Citroën Berlingo vans, which were originally equipped with an automatic gearbox, a classical hydraulic brake system, and a electronic throttle, composed by a potentiometer attached to the pedal that sends an analog signal to the motor controller. Therefore, when the accelerator is stepped on, an electrical signal is sent to the car's internal computer, which was also original Berlingo equipment, that makes the motor move with proportional power.

Then, we can define a similar human-based four-layer hierarchical control architecture to automate the action of the throttle and the brake emulating human driving.

1) The mechanical layer. This is the same as described for the human driven car.
2) The actuation and electronic layer. As the human one, it is composed of three components: an industrial computer to host the control software, an analog output card that sends to the internal car computer a signal that emulates the throttle, and an actuator that moves the brake pedal. The brake is a common hydraulically assisted one, and it is actuated by attaching a dc motor to the pedal, powered and controlled from the PC through both control and power cards.
3) The sensorial layer. Three sensors form this layer. The first is a double frequency GPS receiver, running in carrier phase differential mode that supplies 2-cm resolution positioning at a refresh rate of 10 Hz. The data supplied by the GPS receiver is used for calculating the attitude of the car. This attitude allows determining which is the circulation lane and which other cars affect the normal

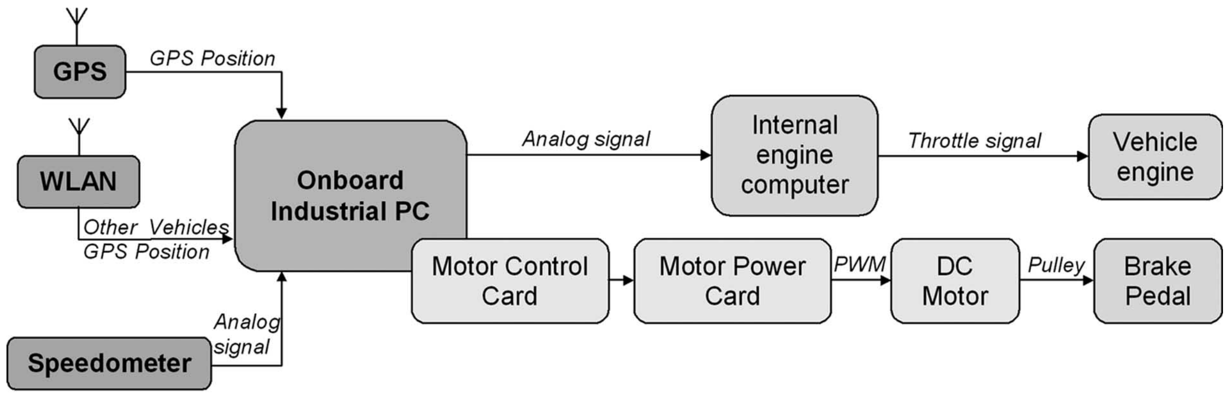

Fig. 1. Schematic description of the hardware setup.

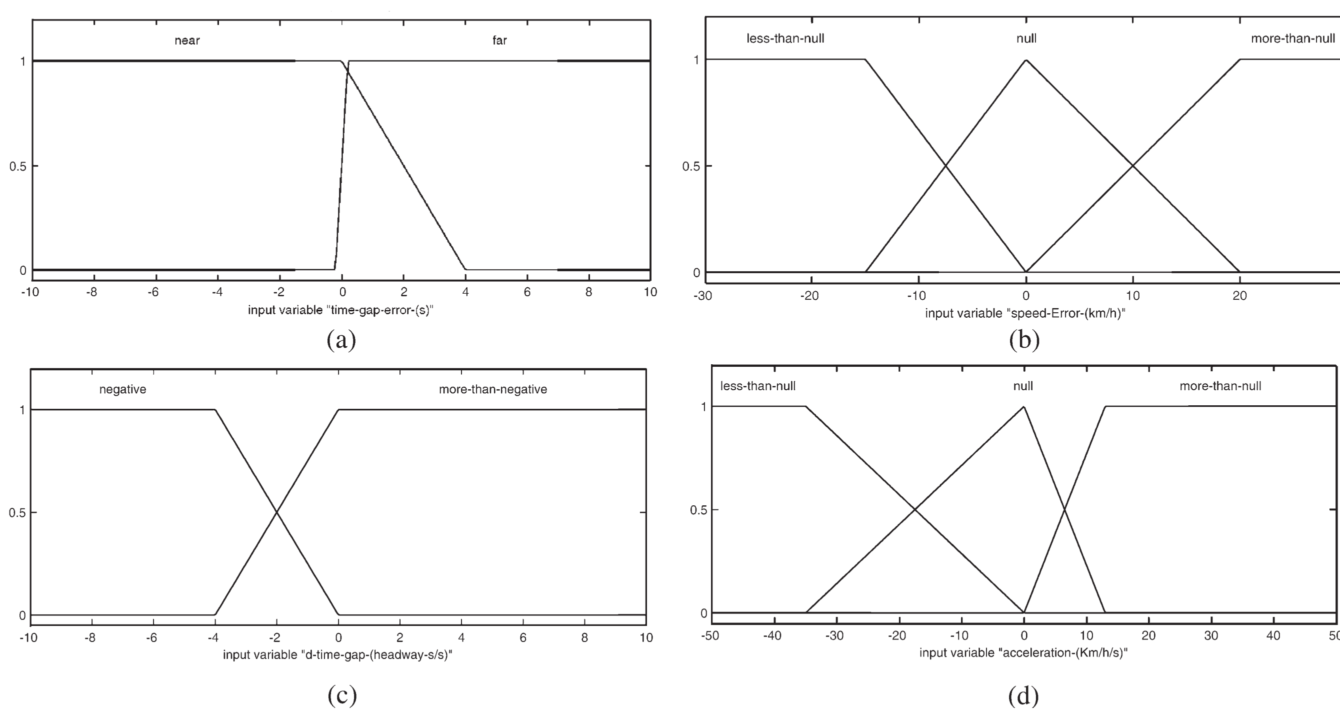

Fig. 2. ACC fuzzy controller input membership functions. (a) *time_gap_error*. (b) *speed_error*. (c) *d_time_gap*. (d) *acceleration*.

circulation. The second sensor we use is a wireless local area network (WLAN) system (IEEE 802.11), which provides information about the position of the other car that circulates in the driving zone. In our case, the second car used for performing the experiments are also equipped with a GPS receiver and a WLAN system, and therefore, it sends its position continuously to the ACC equipped one. This is compatible with the information provided by sensors such as artificial vision [29] or laser scanner. The last sensor is an analog input card that acquires the information from the speedometer of the vehicle.

4) The control layer. This is the longitudinal car control system, based on fuzzy logic. It maintains the speed (CC) and adapts to lane speed (ACC).

A schematic description of the hardware setup is shown in Fig. 1.

C. Throttle Controller

The newly developed controller is based on the existing ACC controller, although this does not use the brake [8]. Briefly, it has four input variables, one output variable, and five rules.

1) Speed error: This is the difference between the current speed and the user-preset speed. It is given by (2). It is represented by the linguistic variable *speed_error*, whose membership function shapes are illustrated in Fig. 2(b).

$$\text{Speed Error} = \text{Current\_Speed} - \text{Preset\_Speed} \qquad (2)$$

2) Acceleration: This is the derivative of the speed at instant t. It is given by (3). It is represented by the linguistic

variable *acceleration*, whose the membership functions are shown in Fig. 2(d).

$$\text{Acceleration}_t = \frac{\text{Current\_Speed}_t - \text{Current\_Speed}_{t-1}}{\Delta t} \quad (3)$$

A digital Fourier filter is applied to this variable with a sampling rate of 10 Hz, a filtering cutting rate of 1 Hz, and four coefficients.

3) Time_gap_error: This is given by (4) below. It has a related linguistic variable named *time_gap_error* whose membership function shapes are shown in Fig. 2(a).

$$\text{Time Gap Error} = \text{Time\_Gap}_{\text{current}} - \text{Time\_Gap}_{\text{target}}. \quad (4)$$

Current time_gap or time headway: This is the time it would take to catch up with the preceding vehicle at the current speed. It is expressed mathematically as

$$\text{Time Gap}_{\text{current}} = \frac{x_{\text{Pursued}} - x_{\text{Pursuer}}}{\nu_{\text{Pursuer}}} \quad (5)$$

where x_{Pursued} and x_{Pursuer} are the GPS coordinates of the lead car and the controlled tail car along the reference trajectory, respectively, and ν_{Pursuer} is the speed of the controlled rear car.

Target time gap: This is the time-headway that the ACC should keep from the preceding vehicle. It should be between 1 and 2 s in commercial ACCs.

4) Derivative of time_gap: This is the variation of the current time gap with time (6). Its related linguistic variable is named *d_time_gap,* and its membership functions are shown in Fig. 2(c).

$$\text{Derivative of Time Gap}_i = \frac{\text{Time\_Gap}_i - \text{Time\_Gap}_{i-1}}{\Delta t}. \quad (6)$$

As this variable is very unstable, it has to be stabilized, in this case, with another Fourier filter, which is very appropriate for our purposes.

The nucleus of the fuzzy control system is made up of fuzzy rules. In our case, they are

$\mathbf{R}_1$ **if** *speed_error* **more than** null **then** *throttle* up;

$\mathbf{R}_2$ **if** *speed_error* **less than** null **and** *time_gap_error* **more than** near **then** *throttle* down;

$\mathbf{R}_3$ **if** *acceleration* **more than** null **then** *throttle* up;

$\mathbf{R}_4$ **if** *acceleration* **less than** null **and** *time_gap_error* far **then** *throttle* down;

$\mathbf{R}_5$ **if** *time_gap_error* near **and** *d_time_gap* negative **then** *throttle* up.

These rules are indicative of some ORBEX engine features.

The control designer can assign the same linguistic values to different variables. In our case, *speed_error* and *acceleration* have the same linguistic value "null," but its meaning is different for each variable, as shown in Fig. 2(b) and (c).

The values "up" and "down" of the output *throttle* are defined by singletons $[-1]$ and $[+1]$, respectively. ORBEX infers a linguistic value for *throttle* values from every rule and blends them via the defuzzification procedure, yielding a crisp value used to activate the analog output card connected to the throttle pedal. This output physical value represents increments to the voltage to be sent to the engine internal controller. Its voltage range is 0–5 V.

Additionally, we should make a couple of appointments concerning the features of the controller.

A parallelism exists between the fuzzy controller and a classic proportional-derivative: We could say that the rules involving the *speed_error* behave like proportional component of the control and that the rules involving the *acceleration* behave like a derivative component. This means that, when the speed of the car is not at the desired value, the *speed_error* rules adjust the throttle pressure, and the *acceleration* rules smooth out the actuation of this command, just like the damping effect of a D-control term.

As we can see, the rule set is a subset of the complete fuzzy inference matrix. The use of an incomplete inference matrix is only feasible when there are situations when some of the elements of the complete set of rules reflect situations that are impossible or have no interest for our control purposes. In our case, because we know the system to model behavior (a human driver) and the working requirements (traffic laws), we have enough knowledge to deal with this rule selection, generating good practical results, as shown in the experiment section.

D. Brake Controller

In order to control the brake pedal, as a human driver would do, we have extended the above ACC fuzzy controller. First, we extended only the CC part with the brake control. This extended CC controller includes the rules for the throttle plus new rules for the brake. There is a duality between the throttle and brake rules. In fact, the brake rules are derived from the throttle rules by substituting the action "*throttle* up" for "*brake* down" and "*throttle* down" for "*brake* up." The joint controller has to coordinate the actions of throttle and brake, namely, to avoid simultaneous actions. This is achieved by defining the membership functions of the "nullb" values involved in brake control (Fig. 3) according to the respective functions of the "null" values involved in throttle control (Fig. 2).

For the *speed_error* variable in the throttle controller, the null membership function is a triangle defined by the parameters -15, 0, and 20. The equivalent definition of the "nullb" membership function in the brake controller is a trapezoid defined by the parameters -14, 0, 3, and 25. In the joint controller, these definitions assure the following facts: 1) The brake is released before the throttle is stepped on when the car is traveling at a speed lower than the target, and 2) the throttle is released before the brake is stepped on when the car is traveling at a speed higher than the target. Additionally, the slopes of the "nullb" function are smoother than the slopes of the "null" function, this assures that 3) the throttle is fully released before the brake starts to act.

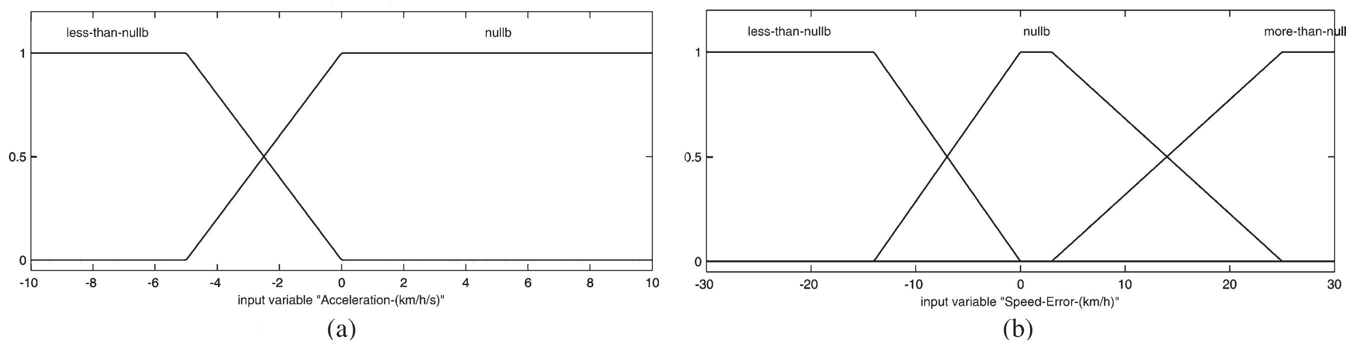

Fig. 3. Brake CC fuzzy controller input membership functions. (a) *accelaration*. (b) *speed_error*.

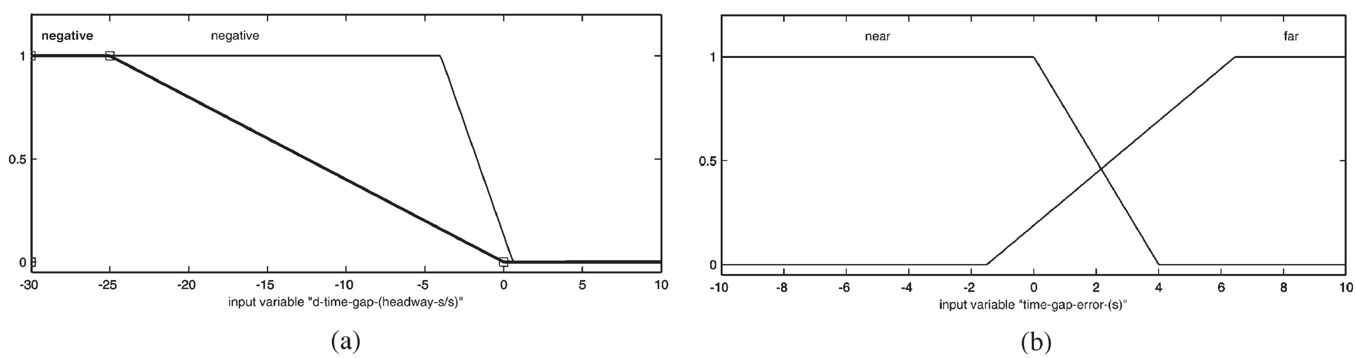

Fig. 4. ACC extended with brake pedal input membership functions. (a) *d_time_gap*. (b) *time_gap_error*.

For the *acceleration* variable, the shape of the "nullb" membership function assures that the brake pedal is released when deceleration is satisfactory for control purposes.

Finally, the brake rules added to the ACC throttle controller are (using brake only for the CC part of the ACC):

$\mathbf{R}_6$ **if** *speed_error* **more than** nullb **then** *brake* down;
$\mathbf{R}_7$ **if** *speed_error* **less than** nullb **then** *brake* up;
$\mathbf{R}_8$ **if** *acceleration* **less than** nullb **then** *brake* up.

The first rule $(\mathbf{R}_6)$ acts when the current speed is higher than the preset CC speed and works cooperatively with the first throttle rule $(\mathbf{R}_1)$. The second rule $(\mathbf{R}_7)$ is the complementary rule and interacts with the second throttle rule $(\mathbf{R}_2)$. The last rule $(\mathbf{R}_8)$ forms the derivative part of the control system, smoothing the speed adaptation maneuvers and actuating cooperatively with the fourth throttle rule $(\mathbf{R}_4)$. Finally, note that the definition of the brake controller does not fully mimic the throttle controller: The dual rule of $\mathbf{R}_3$ $(\mathbf{R}_9)$ has been suppressed because it is not necessary.

The output of the brake controller, named *brake* is also defined as singletons in a similar way than the throttle one and has two linguistic labels "up" and "down" defined at -1 and $+1$, respectively. After defuzzification, this controller generates a crisp value that indicates the position command for the dc motor that controls the brake pedal. Then, its physical interpretation is the increment for the angular position (degrees) of the motor attached to the brake pedal with a pulley. Its range is 0°–240°.

E. Extension of ACC With Brake Pedal Actuation Fuzzy Controller

Now, we are going to explain how to add the full braking capability for the ACC. The desired performance for this controller is expressed in the following four points.

1) ACC will automatically manage the throttle and the brake pedals.
2) The brake pedal will act on the ACC only when the speed reduction produced by fully releasing the throttle (engine braking) is insufficient.
3) Stop&Go maneuvers will use the throttle and the brake pedals.
4) The ACC works like a classical CC when there is no preceding car in the lane.

This controller extends the union of the throttle-only ACC and the throttle plus brake CC. The extension includes the headway variable in the braking rules. Therefore, a new rule has been added and two rules have been modified:

$\mathbf{R}_{10}$ **if** *time_gap_error* near **and** *d_time_gap* negative **then** *brake* down;
$\mathbf{R}_{11}$ **if** *speed_error* **more than** nullb **then** *brake* down;
$\mathbf{R}_{12}$ **if** *speed_error* **less than** nullb **and** *time_gap_error* **more than** near **then** *brake* up;
$\mathbf{R}_{13}$ **if** acceleration **less than** nullb **and** *time_gap_error* far **then** *brake* up.

The first rule $\mathbf{R}_{10}$ represents the need for braking when the distance between the controlled car and the leading car is "near"

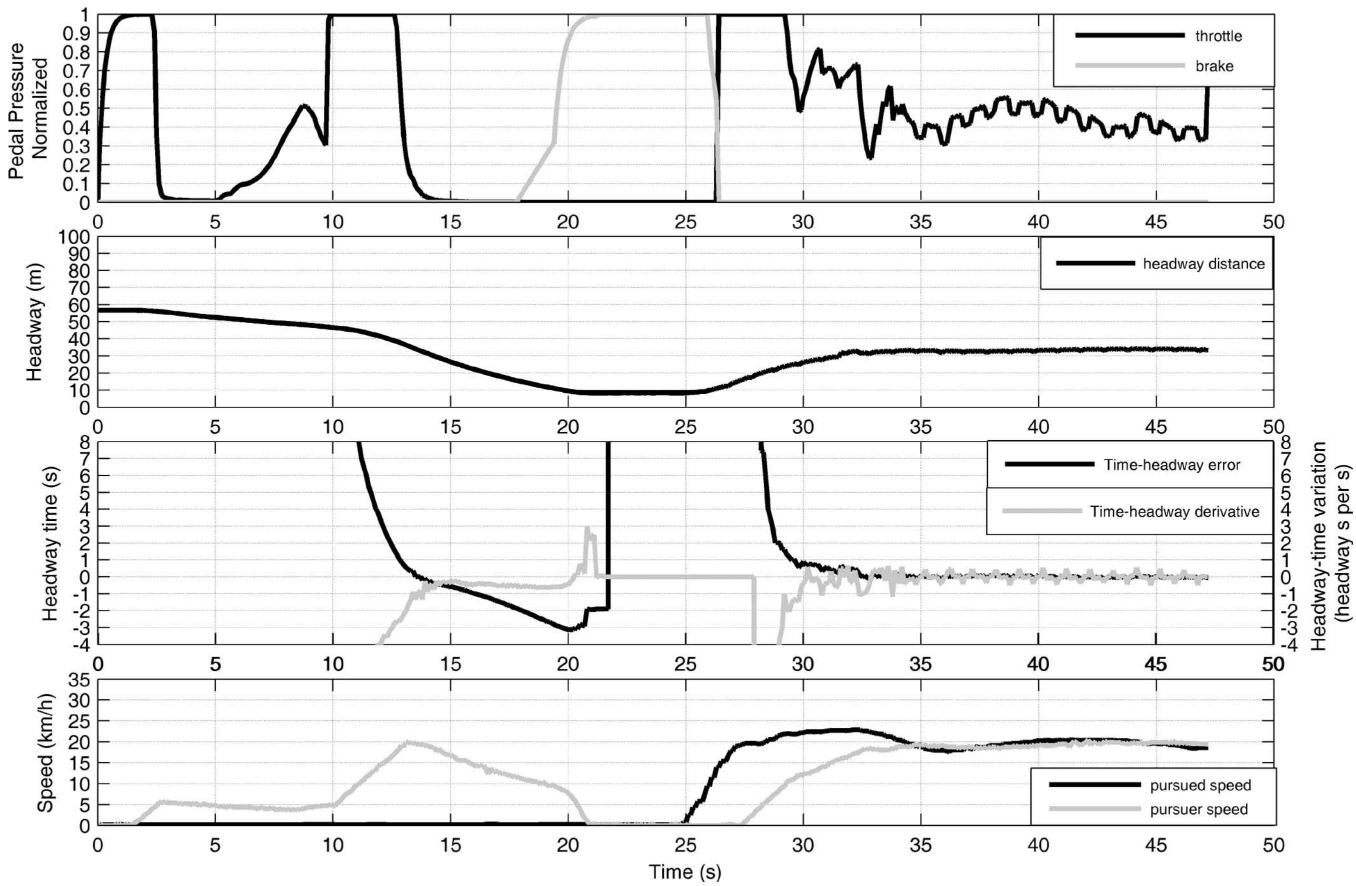

Fig. 5. ACC and Stop&Go maneuver. First experiment.

enough. This distance is reduced as the "negative" membership function of the *d_time_gap* variable indicates [Fig. 4(a)]. Rules $\mathbf{R}_{12}$ and $\mathbf{R}_{13}$ release the brake pedal, depending on the *time_gap_error*. They keep the brake pedal down as long as necessary. The definition of the membership functions for the input variables (Fig. 4) has been refitted to fine tune the cooperation between the brake and throttle controllers for ACC and Stop&Go maneuvers.

Finally, a minimum headway distance has been defined at high level in order to stop the car when the precedent car is stopped, for example, in a traffic jam. This minimum distance has been set at 10 m, measured from the ACC car GPS antenna to the GPS antenna of the precedent one. The real distance between the front of the car and the rear of the other is about 4 m. The reason for including this stopping distance on the system is that, when the car is approximating the stopped fore one, the speed is reducing (it tends to 0), in order to maintain the selected safe time gap. If only the time distance would be considered (if speed is 0 time headway tends to infinite), the car never stops, and it will collide with the precedent one at low speed. Therefore, this is the reason why it is necessary to add a shortcut in the approximation distance that will make the car stop when it is near enough in distance headway. This allows us to perform a pure stop and go operation. This distance is also used for safety reasons in order to minimize the effect of the GPS positioning delay.

To summarize, the full ACC control using the throttle and the brake is made up of the five rules set out in Section II-C and the last four rules.

III. Related Results

In this section, we present an experiment that shows the behavior of the AUTOPIA Stop&Go+ACC system. Two vehicles are involved in it: Babieca, always manually driven, and Rocinante, always automatically driven. In the tests, Babieca is tracked by Rocinante. The differences between them are the initial conditions and the unpredictability of manual driving. Note that both vehicles can be moved automatically, should it be necessary, but we have preferred to drive one manually to verify that our controller can interact with manually driven cars.

An analysis of the experimental graphs (Fig. 5) shows how the components of the Stop&Go+ACC controller are conveniently chained to run the cruise according to established targets, changing conditions, and increasing, namely, unpredictable human actions.

In the graphs shown in Fig. 5, the x-axis represents the time in seconds. The top graph shows the pressure on the throttle and brake throughout the trip. This pressure has been normalized between 0 (pedal fully released) to 1 (pedal fully depressed). The second and third graphs show the time headway in seconds and the gap headway, in meters, between the cars, respectively.

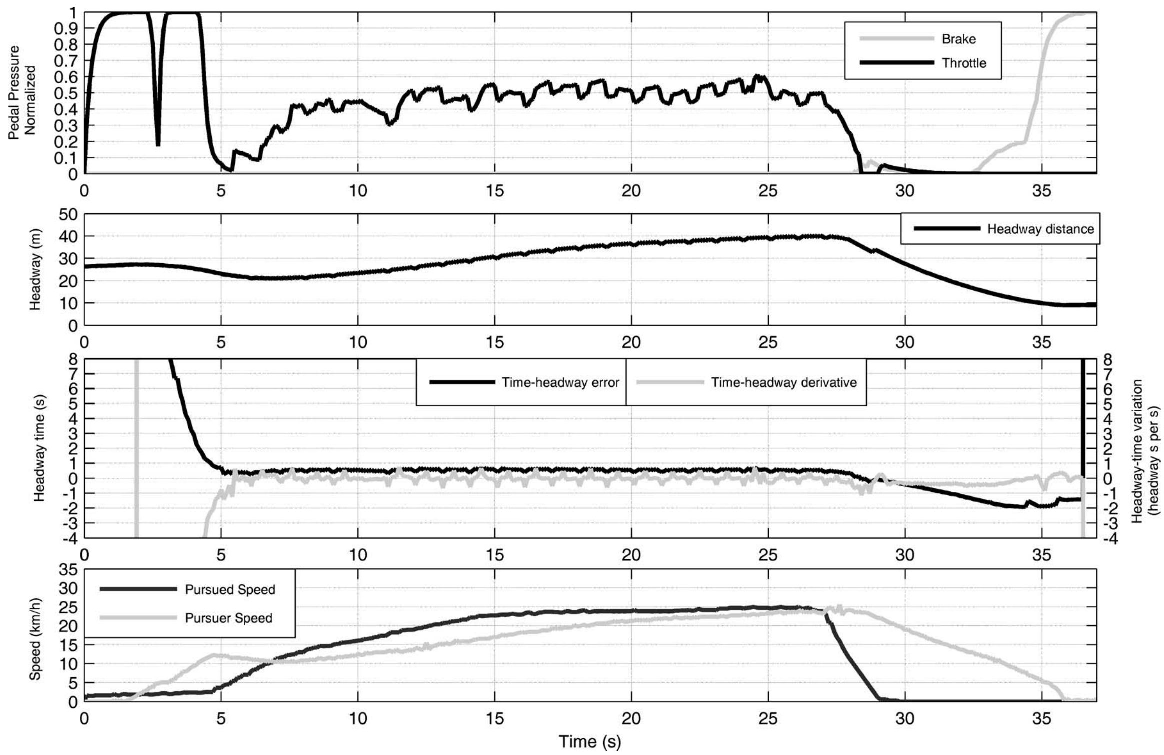

Fig. 6. Stop caused by a sudden braking action of the fore car.

The bottom graph represents the speed of each car in kilometers per hour.

The experiment shown in Fig. 5 represents a classical situation of a vehicle in free circulation (Rocinante) that comes near a traffic jam (Babieca). The first vehicle adapts its speed to the traffic situation, stopping if necessary. The initial conditions of the experiment are the following: Both cars are placed in the same lane and direction, stationary and 67-m apart. The targets are the following: Speed is 30 km/h, time headway is 4 s, and minimum gap headway is 10 m. Rocinante starts driving along its lane, and the initial conditions allow CC control. As Rocinante accelerates, the distance from Babieca, which is still stationary, decreases, and the control switches to ACC, adjusting the speed to keep a safe distance. As the time headway decreases, the pressure on the throttle decreases too. Around second 10, headway is dropping fast, and the extended ACC is activated to slow the vehicle down using the brake. Rocinante continues reducing speed until it is 10 m behind Babieca. At this point, the experiment is reproducing a traffic jam situation. The second graph shows that the *headway error* can become negative (-3 s). This means that Rocinante is only 3 s away from Babieca, which is less than the target headway. This error is not meaningful at this point, because, as the speed is extremely low, the control has switched from ACC to Stop&Go, for which the relevant parameter is the gap in meters. The third graph shows that Stop&Go keeps Rocinante stationary 10 m behind Babieca from second 20 to second 35, more or less. Babieca then starts moving, followed by Rocinante. This automatic control behavior is very similar to human driving: The driver accelerates until an obstacle appears in its path, then releases the pressure on the throttle to slightly reduce speed, and, if this reduction is not enough, applies the brake until the car stops without crashing into the vehicle in front. These results demonstrate the safety of this system.

Fig. 6 shows the effect of a sudden braking when ACC is engaged.

IV. Conclusion

The automation of the throttle and brake pedal of a car allows testing a wide set of automatic control operations. ACC systems are presently commonly installed in vehicles, adding the functionality of maintaining speed and adapting it to the precedent vehicle. However, these systems have an evident limitation: Only use the motor braking for reducing speed; the brake pedal is not automated, and, consequently, it is not capable of managing the speed in stop and go situations. In this paper, we have presented an extension of the ACC functionality where both speed actuators are controlled from a fuzzy system, obtaining management very similar to human behavior and speed maintenance and reduction, and the capability to drive the car in traffic jam situations. Some experimental tests with real vehicles have been performed in order to show the performance of the system.

Acknowledgment

The authors would like to thank the Spanish Ministry of Public Works COPOS project and the Spanish Ministry of Science and Technology Integration of Sensors to Active Aided Conduction project. The authors would also like to thank Citroën España SA, without whose collaboration this research would not have been possible.

REFERENCES

[1] W. D. Jones, "Keeping cars from crashing," *IEEE Spectr.*, vol. 38, no. 9, pp. 40–45, Sep. 2001.

[2] M. Bursa, "Big names invest heavily in advanced ITS technology," *ISATA Mag.*, no. 8, pp. 24–30, Dec./Jan. 2000.

[3] J. Crosse, "Tomorrow's world," *Automot. World*, pp. 46–48, Jan./Feb. 2001.

[4] *Transport Information and Control Systems—Adaptive Cruise Control Systems—Performance Requirements and Test Procedures,* Int. Std. ISO/FDIS 15622, 2002.

[5] R. Holve, P. Protzel, J. Bernasch, and K. Naab, "Adaptive fuzzy control for driver assistance in car-following," in *Proc. 3rd EUFIT*, Aachen, Germany, Aug. 1995, pp. 1149–1153.

[6] J. Koziol, J. Hitz, W. G. Najm, S. Chen, A. Lam, V. Inman, M. Carter, M. Penic, M. Jenson, M. Baker, M. Robinson, and C. Goodspeed, "Evaluation of the intelligent cruise control system. Volume I—Study results," in *Res. Special Programs Admin., Volpe Nat. Transp. Syst. Center*, Cambridge, MA, Oct. 1999.

[7] P. A. Ioannou and C. C. Chien, "Autonomous intelligent cruise control," *IEEE Trans. Veh. Technol.*, vol. 42, no. 4, pp. 657–672, Nov. 1993.

[8] J. E. Naranjo, C. González, J. Reviejo, R. García, and T. de Pedro, "Adaptive fuzzy control for inter-vehicle gap keeping," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 3, pp. 132–142, Sep. 2003. Special Issue on ACC.

[9] A. Hoess *et al.*, "RadarNet," in *Proc. Conf. eMobility*, Gothenburg, Sweden, May 2001. CD-ROM.

[10] J. Langheim *et al.*, "CARSENSE—Sensor fusion for DAS," in *Proc. ITSWC*, Chicago, IL, Oct. 2002. CD-ROM.

[11] A. Broggi *et al.*, *Automatic Vehicle Guidance: The Experience of the ARGO Autonomous Vehicle*. Singapore: World Scientific, 1999.

[12] R. Labayrade, C. Royere, D. Gruyer, and D. Aubert, "Cooperative fusion for multi-obstacles detection with use of stereovision and laser scanner," in *Proc. 11th ICAR*, Coimbra, Portugal, Jun. 30–Jul. 3, 2003, pp. 1538–1543.

[13] J. E. Naranjo, "Sistemas de Comunicaciones Inalámbricos Aplicados a la Conducción Automática de Vehículos," Ph.D. dissertation, Univ. Politécnica de Madrid, Madrid, Spain, Sep. 2001.

[14] S. Kato, S. Tsugawa, K. Tokuda, T. Matsui, and H. Fujiri, "Vehicle control algorithms for cooperative driving with automated vehicles and intervehicle communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 3, pp. 155–161, Sep. 2002.

[15] "Scenarios and Evaluation Framework for City Case Studies," *European Commission Fifth Framework Programme Energy, Environment and Sustainable Development Programme Key Action 4: City of Tomorrow and Cultural Heritage, Deliverable 2, 3*, 2002.

[16] J. P. Laumond, Ed., "Robot motion planning and control," in *Lecture Notes in Control and Information Sciences 229*, New York: Springer-Verlag, 1998.

[17] D. A. Pomerleau, "ALVINN: An autonomous land vehicle in a neural network," in *Advances in Neural Information Processing Systems 1*. San Mateo, CA: Morgan Kaufmann, 1989.

[18] A. Stafylopatis and K. Blekas, "Autonomous vehicle navigation using evolutionary reinforcement learning," *Eur. J. Oper. Res.*, vol. 108, no. 2, pp. 306–318, Jul. 1998.

[19] D. Driankov and A. Saffioti, Eds., *Fuzzy Logic Techniques for Autonomous Vehicle Navigation*. Berlin, Germany: Springer-Verlag, 2001.

[20] M. Sugeno *et al.*, "Fuzzy algorithmic control of a model car by oral instructions," in *Proc. IFSA Special Issue Fuzzy Control,* K. Hirota and T. Yamakawa, Eds., Oct. 1987, pp. 817–818.

[21] M. Sugeno, I. Hirano, S. Nakamura, and S. Kotsu, "Development of an intelligent unmanned helicopter," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 1995, vol. 5, pp. 33–34.

[22] J. M. Mendel, "Fuzzy logic systems for engineering: A tutorial," *Proc. IEEE*, vol. 83, no. 3, pp. 345–377, Mar. 1995.

[23] R. García, T. De Pedro, J. E. Naranjo, J. Reviejo, and C. González, "Frontal and lateral control for unmanned vehicles in urban tracks," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2002, pp. 583–588.

[24] J. E. Naranjo, J. Reviejo, C. González, R. García, and T. de Pedro, "A Throttle and Brake Fuzzy Controller: Towards the Full Automatic Car," in *Lecture Notes on Computer Science 2809*. Berlin, Germany: Springer-Verlag, Jul. 2003.

[25] R. García and T. De Pedro, "First application of the ORBEX coprocessor: Control of unmanned vehicles," in *Proc. EUSFLAT-ESTYLF Joint Conf.*, 2000, vol. 2–3, pp. 265–273. n. 7.

[26] L. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Inf. Sci.*, vol. 8, no. 3, pp. 199–249, 1975.

[27] L. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Inf. Sci.*, vol. 9, no. 1, pp. 43–80, 1975.

[28] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 1, pp. 116–132, Jan./Feb. 1985.

[29] M. A. Sotelo, F. J. Rodriguez, and L. Magdalena, "VIRTUOUS: Vision-based road transportation for unmanned operation on urban-like scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 5, no. 2, pp. 69–83, Jun. 2004.

José E. Naranjo was born in Orense, Spain, in 1975. He received the B.E. and M.E. degrees in computer science engineering and the Ph.D. degree in computer science from Polytechnic University of Madrid, Madrid, Spain, in 1998, 2001, and 2005, respectively.

Since 1999, he has been with the Industrial Computer Science Department, Instituto de Automática Industrial (CSIC), Madrid. His research interest include fuzzy logic control and intelligent transportation systems.

Carlos González (M'80) was born in Torrelavega, Cantabria, Spain, in 1947. He received the degree in physics from Madrid University, Madrid, Spain, in 1969, but he has always worked in computer science, the Master's degree in computer science from the University of California, Los Angeles, in 1974, and the Ph.D. degree in physics from Madrid University in 1978.

He has been with the staff of the Instituto de Automática Industrial, Madrid, since 1971, where he has been a Software Specialist in automation projects.

Ricardo García received the Ph.D. degree in physics from Bilbao University, Bilbao, Spain.

He was a Founder of Instituto de Automatica Industrial, Spanish Research Council, Madrid, Spain, where he has worked in intelligent robotics.

Dr. García received the "Barreiros" Research on Automotive Field Prize for his AUTOPIA project on ITS in 2002.

Teresa de Pedro received the Ph.D. degree in physics from Universidad Complutense of Madrid, Madrid, Spain, in 1976.

Since 1971, she has been working on artificial intelligence applied to automation with the Instituto de Automática Industrial, Spanish Research Council, Madrid. Currently, she is interested in fuzzy models for unmanned vehicles, and she is the Head of a Spanish team involved in the Integration of Sensors to Active Aided Conduction project.