Cascade Architecture for Lateral Control in Autonomous Vehicles

Joshué Pérez Rastelli, Vicente Milanés, Enrique Onieva

Cascade Architecture for Lateral Control in Autonomous Vehicles

Joshué Pérez, Vicente Milanés, and Enrique Onieva

Abstract—**Research on intelligent transport systems (ITSs) is steadily leading to safer and more comfortable control for vehicles. Systems that permit longitudinal control have already been implemented in commercial vehicles, acting on throttle and brake. Nevertheless, lateral control applications are less common in the market. Since a too-sudden turn of the steering wheel can cause an accident in a few seconds, good speed and position control of the steering wheel is essential. We present here a new cascade control architecture based on fuzzy logic controllers that emulate a human driver's behavior. The control architecture was tested on a real vehicle at different vehicle speeds. The results showed the use of a straightforward and intuitive fuzzy controller to give good performance.**

Index Terms—**Autonomous vehicles, fuzzy logic, intelligent transportation systems (ITSs), lateral control, system analysis and design.**

I. INTRODUCTION

AUTONOMOUS vehicles, once a utopia, are now poised to become a reality. As engineers, we have a broad range of possibilities of participation, since there are several fields involved in the control and automation of vehicles: control maneuvers, vehicle and infrastructure instrumentation, vehicle simulation, etc. [1].

One of the first developments in the field of advanced vehicle-control systems took place in the early 1960s by the General Motors Research Group. They developed and demonstrated the automatic control of the steering, speed, and braking of automobiles [2] on test tracks. Later, other research groups began to improve the lateral and longitudinal control of autonomous vehicles. In the late 1960s, Ohio State University and the Massachusetts Institute of Technology (MIT) began working on the application of these techniques to urban transportation problems [3]. The first broad-scale investigation of the application of automation technologies to urban transportation problems appeared in MIT's Project METRAN [4].

The authors are with the Industrial Computer Science Department, Centre of Automation and Robotics (CAR), Universidad Politécnica de Madrid, Consejo Superior de Investigaciones Científicas, (UPM-CSIC), La Poveda-Arganda del Rey, 28500 Madrid, Spain (e-mail: joshue.perez@car.upm-csic.es; vicente.milanes@car.upm-csic.es; enrique.onieva@car.upm-csic.es).

Recent years have seen significant advances in research in this area, with several applications to commercial vehicles. The leading vehicle manufacturers are investigating the development of new advanced driver-assistance systems (ADASs) [5]. The goal of ADAS is to aid drivers in critical situations rather than to replace them. The most important advances have been cruise control [6], dynamic stability control [7], antilock brakes (ABS) [8], pedestrian detection with night-vision systems [9], collision avoidance [10], and semiautonomous parking and warning signals [11], among others. These applications have mainly been implemented on the longitudinal control (actions of acceleration and braking).

Lateral control concerns the action on the steering wheel. One of the pioneers on work on lateral control was Ackermann [12]. His approach was to merge active steering with yaw rate feedback to robustly decouple the yaw and lateral motions. Another useful method for performing lateral control is based on a predefined reference trajectory [13]. Through techniques that allow quick, smooth, and high-quality control, it is possible to control such nonlinear dynamic systems such as the steering in a car. Techniques including fuzzy logic [14], linear matrix inequality optimization (in automated snowblower) [15], and yaw rate control [16] have been used to keep the vehicle following the reference trajectory.

Choi [17] developed an adaptive control law and a distance rate observer for the lateral control of autonomous vehicles using magnetic sensors in the vehicle's front wheels. In a parallel line of work, the Nissan Research Center has achieved significant advances in lateral control using error-canceling feedback control to estimate the next point on the track and to generate the possible steering outputs [18].

Another significant change in the steering wheel system has been the inclusion of electric-power-assisted steering (EPS) as a replacement for the traditional hydraulic power steering (HPS) systems in new-generation vehicles. There have been simulations of the advanced control of this kind of system [19]. Guvenc and Guvenc [20] present a two-controller structure proposal for the generic EPS system, addressing motor torque and steering motion.

Yih and Gerdes [21] of Stanford University modified a real vehicle so that it could use a steer-by-wire system and receive Global Positioning System (GPS) data. They used a bicycle model and real-time estimation for the control and the cancelation of the effects of steering system dynamics and tire disturbance forces. Furthermore, there have been several applications for automated highway systems (AHSs) targeted at heavy vehicles [22]. Those authors incorporated an inner loop controller into the nested lateral control architecture for autonomous driving.

In the control of dynamic systems, there is, however, a large gap between the design of a controller using simulations and its implementation in a real system. This gap is gradually being reduced with advances in different simulators [23], [24], but these simulators still depend on the level of complexity and the different constraints imposed by the systems. In this context, intelligent control offers powerful techniques for the control of such highly complex systems such as autonomous vehicles.

In 1985, Sugeno and Nishida published a landmark paper. They demonstrated that any industrial process whatsoever can be controlled with a simple model of human operation or human experience [14]. The problem was reduced to finding the proper control rules and fine-tuning them based on a driver's experience. Various studies have compared fuzzy control and classical control techniques in autonomous vehicles [25], with the fuzzy controllers showing good results.

The AUTOPIA program was begun in 1998 at the Institute of Industrial Automation of the Spanish National Research Council (IAI-CSIC). The experiments to be described in this paper were conducted with an electric Citroen Berlingo, implementing a new lateral control scheme for the steering wheel. It is a continuation of previous studies carried out by the group [26], [27] but considerably improving the results achieved so far. In brief, the improvements, both hardware and software, with respect to previous works are the following.

1) The power stage based on the ISA Bus card was removed for a new-generation power stage connected to an onboard PC through Ethernet connection. With the new proportional–integral–derivative (PID) controller, the delay times were reduced, and a faster time response was achieved [28]. A new motor and a gear ratio were installed [29].

2) Moreover, the advance of this new steering control permits using a new output control variable, i.e., the steering wheel's angular speed, which had never been used in previous works.

3) Taking into account this new output variable, two fuzzy inputs have been included, i.e., the actual speed and the distance to the bend, to perform a new fuzzy controller that permits smooth steering wheel control.

4) A low speed was previously used in bends (8 km/h) [30]. With the new angular speed and position controller, the corners can be taken up to 24 km/h.

5) Finally, a unique fuzzy controller capable of driving in both straight and bend segments has been developed, substituting the two previous controllers. The new output variable—the steering wheel's angular speed—permits using the same rule base for these two scenarios. Previously, two steering controllers were needed [26].

The lateral controller described in this paper considers the angular speed and position as output variables. A cascade steering control architecture performs well over different ranges of speeds and difficult urban curves (i.e., cornering). The main contribution is the use of the position and angular speed of the steering wheel controlled through fuzzy logic to carry out smooth and comfortable control aimed at emulating a human driver.
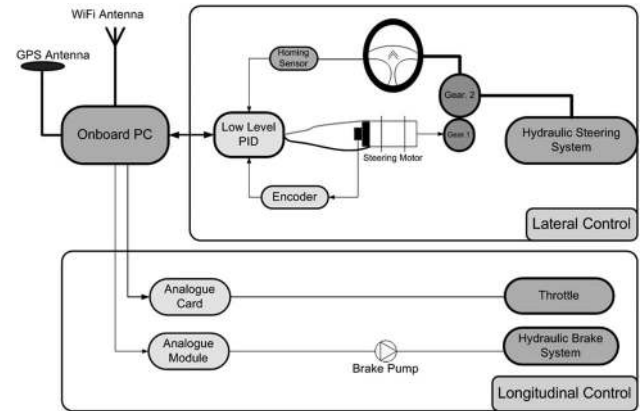

Fig. 1. Control diagram of the unmanned vehicle.

This paper is organized as follows. In Section II, we describe the system, explaining the lateral control. An overview of the control cascade paradigm is presented in Section III, where the PID and fuzzy controls are the low- and high-level controllers, respectively. Section IV describes the experiment in different situations, with various speeds and curves on a track in our own facilities. Finally, some remarks and conclusions are presented in Section V.

II. SYSTEM DESCRIPTION

The main sensor equipment is a real-time kinematic Differential GPS (DGPS), located over the rear axle of the vehicle. The DGPS sends the universal transverse mercator coordinates to the control program at a frequency of 5 Hz. With this system, an accuracy of 2 cm is achieved on our test tracks. The facilities of the Institute of Industrial Automation (IAI-CSIC) emulate an urban circuit. Even though the speed that we can use is limited, the circuit provides a good field test for maneuvering. In particular, it permits curves with different angles to be taken at different speeds.

A. Longitudinal Control

The test vehicle used in this paper is propelled by an electric motor. The longitudinal and lateral controls are separated in hardware and software so that we can independently use each system. To act on the throttle, a digital–analog card is connected between the PC and the pedal. This card sends the target reference—an analog value—from the control program installed in the PC to a trimmer throttle that emulates the desired level of pressure applied to the pedal. The accelerator control output generates a signal of 1–4 V. The computer braking procedure was implemented by adding a brake circuit connected to an electro-hydraulic pump. The real speed is read directly from the tachometer of the vehicle to close the longitudinal control loop. The connection of the longitudinal control system is shown in the lower part of Fig. 1.

B. Lateral Control

The test vehicle is equipped with an HPS system. The steering wheel is controlled by a dc motor, through a gear reduction fixed to the motor axle and the steering bar. In previous work
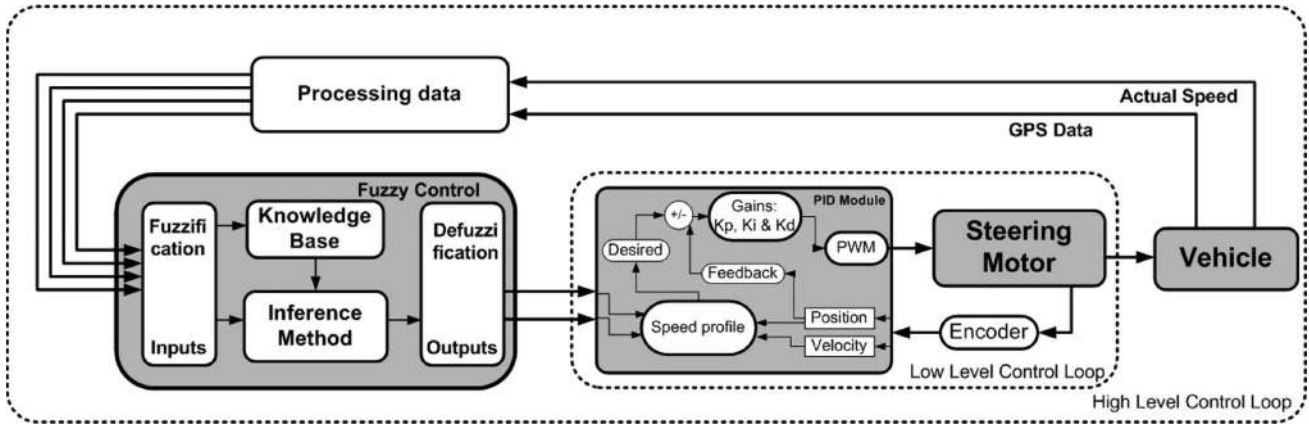
Fig. 2. Lateral control architecture.

[28], [29], the parameter estimation was carried out taking into account the control of the steering wheel position as the only output variable.

To obtain good control of the steering wheel, one needs both speed and position control of the motor coupled to the steering bar. In this paper, we perform a torque estimation that permits us to obtain a steering wheel speed range appropriate to our needs.

The dc motor has a power of 90 W. Its maximum speed, with a reduction ratio of 66 : 1 (to increase the torque and to reduce the speed of the original device), yields around 98.49 r/min, and the torque is 7.26 Nm. We used a torque wrench to experimentally estimate the torque needed to move the steering wheel when the vehicle was stopped on an asphalt surface: 5 Nm. A gear ratio of 5 : 6 was used to ensure a suitable compromise between torque and speed. The steering speed is 82.075 r/min, and the steering torque is around 8.71 Nm, in accordance with this gear ratio. A safety factor of 42.5% is obtained for the torque. This precaution is in case the friction between the wheels and the ground surface changes. A diagram of the steering connection is shown in the upper part of Fig. 1.

The capacity of our system can be better understood by considering its mechanical and physical characteristics. For the control of the steering wheel system, a cascade architecture is used, with the angular speed and position being considered inside the control loop.

III. Cascade Architecture

The lateral control was implemented by means of a two-level cascade control architecture. The low level is governed by a discrete PID controller (with no interactive algorithm), and the high level is governed through fuzzy logic. The input variables are acquired from an onboard DGPS and the tachometer of the vehicle. The low-level controller is located in an external device and receives its target from the fuzzy output, which is running on the onboard PC. It attempts to reach the desired reference as quickly and smoothly as possible.

There are other applications that use cascade control based on a double PID control loop. Two examples are air conditioning systems [31] and superheated steam [32]. Their time response is, however, too slow for the control of vehicles.

In our case, the PID controller is in the inner or secondary control loop. The sampling frequency is 100 Hz, which is 20 times faster than the DGPS frequency (5 Hz). The PID control loop is a slave of the outer or primary fuzzy control loop. Therefore, the logical sequence to tune the system is given as follows: 1) The PID controller is adjusted with the fuzzy control in open loop; 2) the loop is closed, and the fuzzy parameters are tuned, and 3) finally, an iterative process is carried out to readjust the inner parameters.

The four input variables are obtained taking into account the predefined route that will be followed by the vehicle. They are the angular error, the lateral error (for the position control), the distance to the next bend, and the actual speed of the vehicle (for the angular speed control). The fuzzy control analyzes and processes these variables, and, on the basis of the knowledge base that represents the experience of a driver and an inference method, yields two outputs—the desired position of the steering wheel (between $-540°$ and $540°$) and the desired speed for the speed profile (between $0°/s$ and $220°/s$)—which are sent to the low-level controller (see Fig. 2).

A. Low-Level Controller

The low level receives the target from the high level and communicates the control signal to the steering wheel motor. This inner loop is closed by an encoder of 500 pulses per turn. This module includes a position control mode, which allows the generation of Linear Segment with Parabolic Blend (LSPB) trajectories. Before tuning the PID, a system identification process was carried out to achieve a good-quality controller.

System Estimation: An open-loop estimate of the steering wheel was made. We connected a 5-A maximum current external voltage source. Fig. 3 shows a diagram of the estimation procedure. The onboard PC receives the pulses from the encoder attached to the motor axle. This approach only considers the position of the steering wheel, taking a constant speed of around $220°/s$. This is the fastest angular speed of the profile (corresponding to 4.9 s from the rightmost to the leftmost position) because this estimate was made when the vehicle was moving. The average vehicle speed used is around 10 km/h, the idea being to get as close to the real vehicle dynamics as possible.
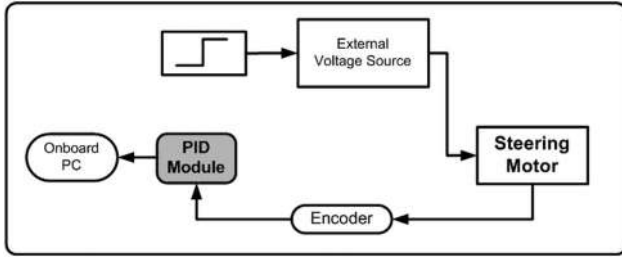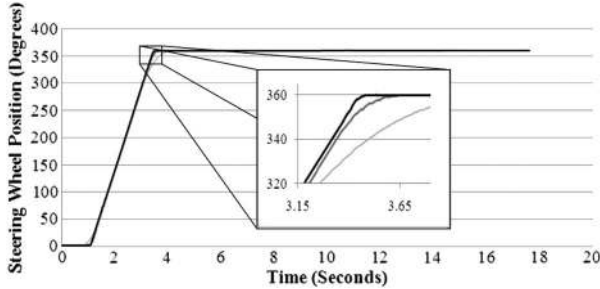
Fig. 3.   Open-loop steering wheel estimation.

Fig. 4.   PID output for different decelerations.

After the data collection, the "ident" toolbox of MATLAB was used to perform a nonparametric estimation. The transfer function obtained was

$$H(s) = \frac{0.8154 * e^{-j0.5}}{(s + 3.8913)(s + 3.9377)}. \tag{1}$$

PID Controller Estimation: Given this transfer function, the PID gains (Kp, Ki, Kd) were obtained by applying the Ziegler–Nichols method. First, the gains were determined for a continuous PID. As our module is a discrete PID, we considered the sampling period (10 ms) to determine the new gains. The values obtained were Kp = 1.1358, Ki = 0.01977, and Kd = 0.001.

The derivative gain is small because we expect a rapid response with no peaks in the output. The behavior of the low-level controller in reaching the target reference is shown in Fig. 4 for different decelerations.

Speed Profile Parameters: In robotic dynamics and control, the generation of the speed profile is one of the most important steps, calculating the trajectory in terms of the position, speed, and desired time. The generation of joint spatial trajectories has been used for robots in an environment with obstacles [33]. In our application, however, we applied linear segment with parabolic blend (LSPB) trajectories in the inner control loop. While these are generally easy to visualize and estimate, our system's inertial characteristics and the speed of the control loop do not permit a good detailed view.

Two groups of variables were taken into account to estimate the speed profile. The first group consists of dynamic variables: the actual position and speed from the encoder and the target position and the desired top speed from the high-level controller. The second group of variables is fixed by the controller: the dead zone, the maximum acceleration, and the maximum deceleration. The dead zone is complementary to the deceleration and defines a zone around the goal, i.e., the
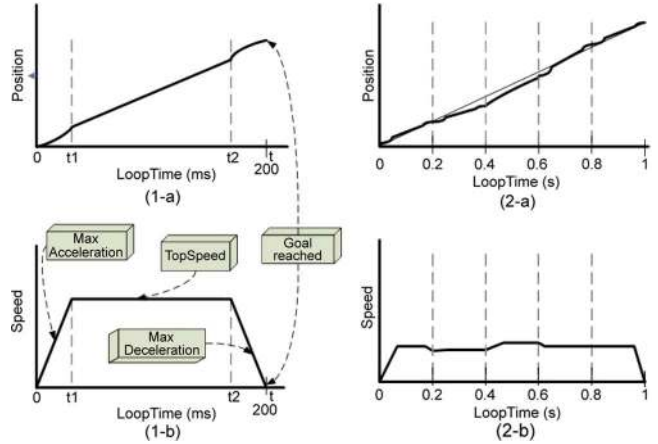

Fig. 5.   LSPB trajectories in the inner loop. Position and speed profiles in a single control loop [(1-a) and (1-b)], and in five control loops [(2-a) and (2-b)].

target position: dead zone. It is used to force the desired speed to a zero value. Fig. 5 shows the behavior of LSPB trajectories in a single control loop (see the left-hand side of Fig. 5) and in five control loops (see the right-hand side of Fig. 5). Because of the motor axle's inertia, the plot of the speed is continuous.

Fig. 5(1-b), shows the speed profile in a simple loop. The acceleration lasts from $t = 0$ to $t = t_1$. The deceleration ramp lasts from t_2 to the end. The values of t_1 and t_2 are set by the high-level controller. Thus considering the position equation (2) below and differentiating twice (3), one obtains the value of the acceleration. An analogous method is used for the deceleration estimation, i.e.,

$$P_{(t)} = P_i + \frac{Vmax}{2 * t_1} * t^2 \tag{2}$$

$$\ddot{P}_{(t)} = \frac{Vmax}{t_1}. \tag{3}$$

The end goal is continuous control, increasing or decreasing the steering wheel's angular speed (depending on the case), until the desired steering position is reached. The idea is to reach the desired position (given by the high-level controller) at the corresponding desired angular speed. With the proposed cascade architecture, we eliminated possible oscillations in the steering wheel control.

The control speed we proposed was aimed at mimicking human behavior. In previous work [26], steering control around bends was done with the dc motor working at top speed throughout the maneuver. This can involve excessive effort for the motor, thus reducing its life cycle. The most important feature now is not merely reaching the desired steering wheel position but doing so at the right speed, which will be based on the vehicle's speed and the distance to the next bend. These decisions are made by the high-level controller.

B. High-Level Controller

The aim of the outer control loop is to smoothly and effi-ciently move the steering wheel to the target position in accor-dance with the desired angular speed. The high-level control is based on a fuzzy logic controller. It sends the steering wheel's target position and angular speed to the PID control loop. The
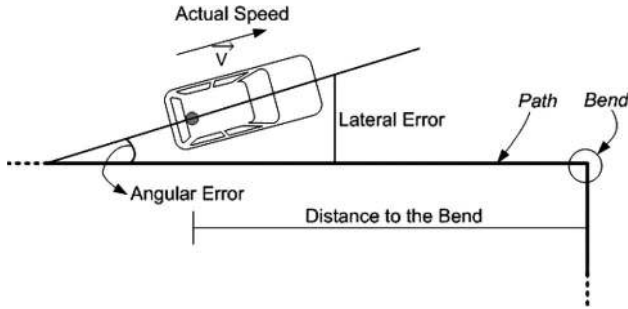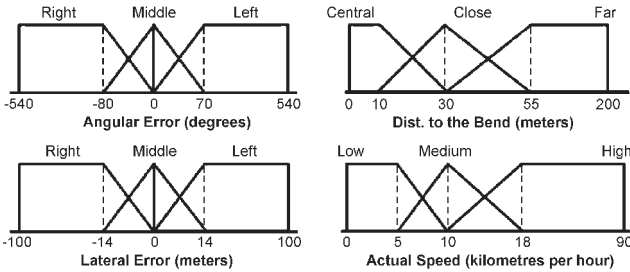
Fig. 6. Fuzzy input variables.

Fig. 7. Membership functions for the steering wheel control.

fuzzy controller uses four input variables taken from the DGPS, the digital cartography, and the actual vehicle speed. These variables are the angular error, the lateral error, the distance to the bend, and the actual speed. The frequency of the high-level control loop is 5 Hz. The fuzzy control is divided into three stages: fuzzification, inference, and defuzzification (see the left-hand side of Fig. 2).

In previous work, our autonomous vehicles used two driving modes: straight mode and curve mode [26]. The greatest difficulty was encountered in the curve mode, because only the position of the steering wheel was controlled while the angular speed was kept fixed. In the straight mode, the position outputs were limited to maintaining the route, avoiding large adjustments. In this paper, we consider a single controller that uses the angular speed to adapt the movement of the steering wheel to the route.

1) Input Variables: Four input variables govern the high-level controller: the angular error and lateral error for the position loop and the distance to the bend and actual longitudinal speed for the angular speed loop. The angular error is the angle between the vehicle's axis and the predefined path, measured in degrees. The lateral error is the deviation, in meters, of the front of the vehicle from that same predefined route. The distance to the bend, in meters, is estimated using the vehicle's position on the reference map. Finally, the actual longitudinal speed comes from the tachometer and is measured in kilometers per hour. Fig. 6 shows these input variables.

These input values are transformed and interpreted as fuzzy data in the fuzzification stage processes. Each variable is defined by a membership function involving its corresponding linguistic labels, which is represented in the rule base. The membership functions have triangular or trapezoidal shapes, as shown in Fig. 7. The lateral and angular errors are calculated from two consecutive DGPS points, using three labels in each case: left, middle, and right, as shown on the left-hand side of Fig. 7.

The other input variable that has been introduced is the distance to the bend. The value of this variable is computed by examining the distance to two vertices of the trajectory—the preceding one and the next one. When the vehicle is in the first half of the segment, the value is the distance to the preceding vertex with a negative sign. In the second half of the segment, it is the distance to the next vertex with a positive sign. A positive value of the variable contributes to turning the steering wheel while entering the bend, and a negative value contributes to centering it again after the bend. Five linguistic labels are defined to distinguish whether the vehicle is far from, close to, or in the middle of the bend. The membership functions are symmetric, so that the steering wheel must return with the same angular speed as the one it used when entering the bend. The upper right graph of Fig. 7 shows the corresponding membership functions.

The actual speed is measured in kilometers per hour. In our application, we considered a speed range from 8 to 24 km/h, in curves of 90°. Three labels are defined for the vehicle's speed, as shown in the lower right graph of Fig. 7.

2) Output Variables: The defuzzification operation uses the center-of-area method [see (4) below], where W_i are the weights of the linguistic label i for each membership function, O_i are the assigned values of the singleton output for the linguistic label i, and X_i is the crisp value of each rule i condition. This formula is implemented for each output variable depending on each linguistic label. The Mandani inference method was used to calculate each crisp steering signal, i.e.,

$$X_i = \sum \frac{W_i O_i}{W_i}. \qquad (4)$$

Our controller is a typical fuzzy PD controller, i.e., it uses two inputs and produces one output. The parameters are normalized and denormalized at each fuzzy stage (fuzzification, inference, and defuzzification). The output variables are singletons with possible weights: Left (value −1), Right (value 1), and Nothing (value 0) for the position control loop, and Low (value 0.4), Medium (value 0.6), Med_high (value 0.8), and High (value 1) for the angular speed loop.

3) Rule Base: The rule base interprets the input variables in a process termed inference. It provides the target output of position and angular speed for the steering wheel. Our fuzzy controller, which is known as ORBEX (experimental fuzzy coprocessor) was developed in the IAI and allows fuzzy rules to be written as sentences in an almost natural language [30]. The sentences used are of the following form: IF. . . THEN. . .. ORBEX allows the user to define variables and their fuzzy rules and to join them with other variables in rules to yield crisp signals for the controllers.

The rules—using the ORBEX syntax—used for the position control are the following:

IF *Lat_Error* Left THEN *Steering_Pos* right
IF *Lat_Error* Middle THEN *Steering_Pos* nothing
IF *Lat_Error* Right THEN *Steering_Pos* left.
IF *Ang_Error* Left THEN *Steering_Pos* right
IF *Ang_Error* Middle THEN *Steering_Pos* nothing
IF *Ang_Error* Right THEN *Steering_Pos* left.

TABLE I
FUZZY RULE FOR THE STEERING WHEEL SPEED CONTROL

Dist_Bend	Actual Speed		
	Low	Medium	High
Central	*Medium*	*Med_high*	*High*
Close	*Medium*	*Med_high*	*Med_high*
Far	*Medium*	*Medium*	*Low*

Steering_Pos is the singleton output for the steering wheel position. The rule bases for the angular speed output are symmetric with respect to the negative or positive values of the distance to the bend. Table I shows the rule base.

Fuzzy control allows human behavior to be emulated in following a predefined path. The fusion of the two controllers—one for the position and the other for the speed control of the steering wheel—make the steering behavior more comfortable.

Fig. 8 shows the control surfaces for the steering position and the steering angular speed. These are representations of the mapping of the inputs and outputs based on the fuzzy rules. The smoothness of the surfaces is indicative of the reasonableness of the controller rules and tuning. The angular speed controller surface (left) is highest when the vehicle is in the middle of the bend and when the actual speed is high but is lower once the vehicle has passed the bend and is beginning a straight segment of the route.

IV. EXPERIMENTS AND RESULTS

The experiments to be described show the behavior of the steering wheel cascade architecture implemented in our electrical vehicle. They illustrate the performance at both low and high vehicle speeds. The predefined path selected for this purpose was based on the tracks of the IAI facility. Both paved sections of the urban circuit and unpaved parts of the circuit's terrain were included. The reason was that the urban (paved) circuit only has curves of around 90°, and we wanted to include other angles to test the system over a reasonably wide range. A secondary consequence was that we also increased the external perturbations acting on our system.

The upper left picture in Fig. 9 shows the path used. The angles at the curves varied from 21° to 105°. The experiments consisted of negotiating these curves at different longitudinal speeds. The route has ten straight sections of different lengths. The sections that begin with curves $\theta1$, $\theta2$, $\theta3$, and $\theta6$ are on unpaved terrain. The total route is 500 m in length. The speeds used in the experiments were between 8 and 24 km/h.

The rest of Fig. 9 shows the actual behavior of the vehicles following this predefined path. Four speeds were used to actually test our controller: 8, 12, 16, and 20 km/h. These speeds were taken from human experience, since greater speed is unnecessary in an urban circuit. These experiments showed good performance of the steering wheel angular speed and position controller. Also shown is an extra experiment at a greater speed (24 km/h) to test the robustness of the controller. In this test, although the vehicle keeps to the road, one observes that the error is greater as it follows the path, particularly around the sharpest curve ($\theta8$; see the lower right graph of Fig. 9). Indeed, this was the most difficult curve, because the vehicle must turn 105°.

When the vehicle enters and leaves the unpaved sections of road, it tracks the route with no problem.

The variables of our controller are mainly calculated on the basis of a high-precision DGPS, which uses a reference position of a central station in our facilities. Sometimes, this precision is lost due to communication problems over the wireless network. In the middle left graph of Fig. 10 (around curve 5), with the vehicle travelling at 12 km/h, one observes such a loss of GPS precision—from fixed (up 2 cm) to float (50 cm) mode—which lasted for 400 ms. However, our cascade control system is able to filter this signal, avoiding sudden jumps of the steering wheel and maintaining the vehicle on the road. If this error persists for a second—five control loops—the vehicle would make an emergency stop. However, this was not the case in this experiment since the precision losses were for less than a second, and the vehicle continued on its route.

The goal of smooth comfortable autonomous driving with this steering wheel controller has been achieved. Fig. 10 shows the two outputs of the controller: the position and the angular speed of the steering wheel. The graphs show the results of experiments at the five different speeds. The time scale (the horizontal axis) of each experiment varies from 240 to 80 s, in accordance with the longitudinal speed of the vehicle.

The graphs on the left-hand side of the figure show the performance with respect to the position of the steering wheel. The curves correspond to the position output (in degrees) of the fuzzy control and the actual position of the steering wheel (read from the motor encoder). The low-level control loop, which is a PID controller, permits the response generated by the high-level controller to be smoothed out. It thus avoids undesired oscillations of the steering wheel that could particularly be dangerous at a high speed. In the second graph on the left-hand side of Fig. 10, at around 70 s, the fuzzy controller shows a large spike in the position, which was due to the loss of quality of the DGPS. However, the steering wheel speed control permits no sudden movements when the vehicle is on a straight stretch of road far from a curve; therefore, in practice, the system continued to function despite this brief loss of input quality, and the vehicle was able to automatically continue on its route.

The fuzzy controller's position output has large fluctuations at the beginning of each route. This is because the two points of the DGPS are still very close to each other. As the vehicle starts to move, however, the fuzzy controller's output improves. This output also oscillates when the vehicle takes bends of just a few degrees at low speeds, but with the angular speed control (applied to the steering motor) and the action of the PID, the actual position of the steering wheel smoothly changes.

The graphs on the right-hand side of Fig. 10 show the output of the angular speed controller. This controller was designed for vehicle speeds in the range of 5 to 18 km/h, as the latter is the top speed at which a human driver will negotiate a corner of 90° or greater to avoid accidents. In the experiments at 20 and 24 km/h, therefore, the controller's output is identical because the membership function of the "actual speed" is saturated at the linguistic value "high" for speeds of 18 km/h or greater (see Fig. 7).

The progressive increase that one observes in the angular speed output is because all the tests were made on the same
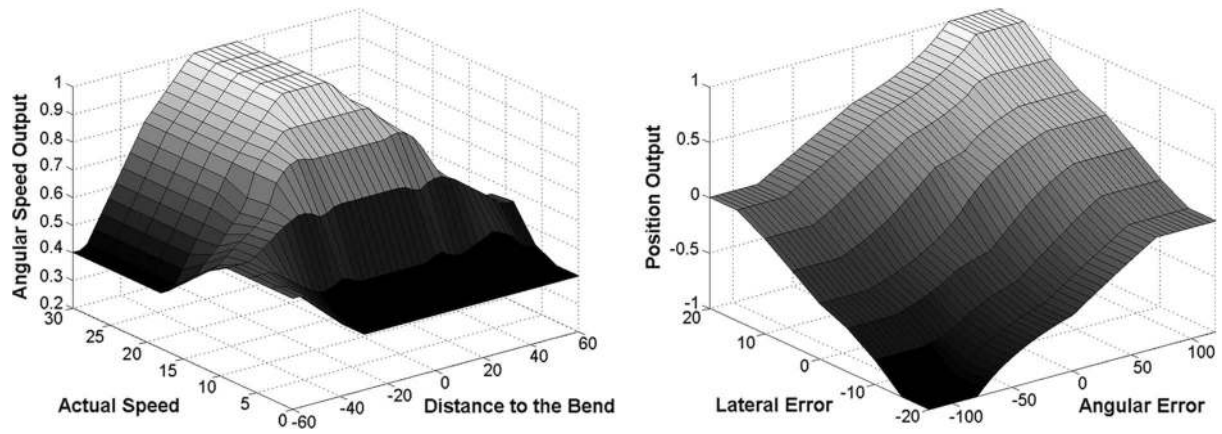
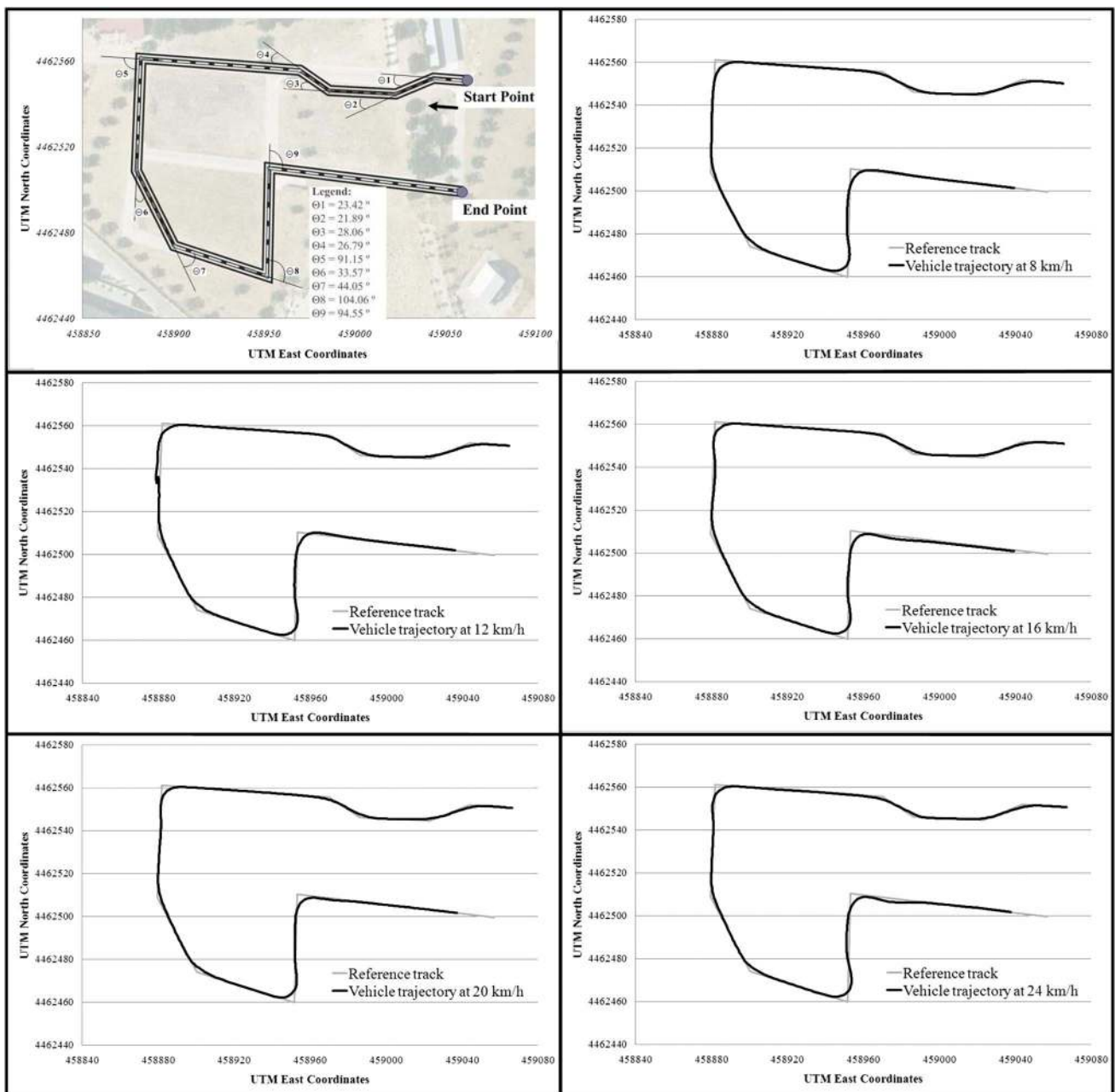Fig. 8.   Control surface of the steering wheel.

Fig. 9.   Path used including bends of different angles and the vehicle's trajectory in experiments at different speeds.
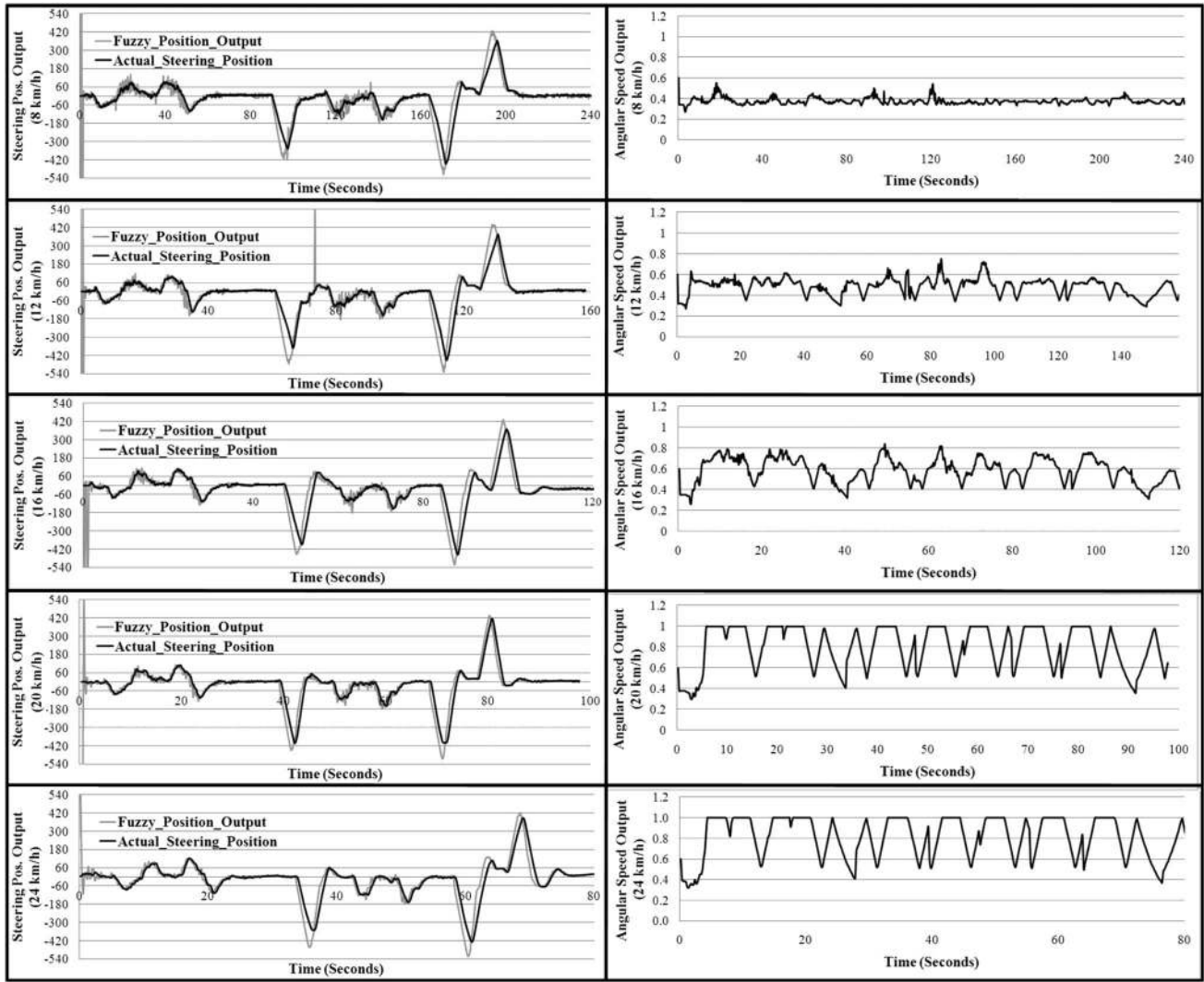
Fig. 10.   Fuzzy control output variables: position of the steering wheel before and after the low-level control (left) and the steering wheel angular speed output (right) at different speeds: 8, 12, 16, 20, and 24 km/h.
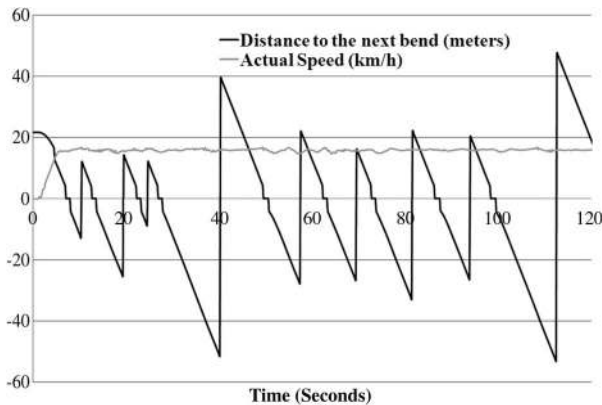

Fig. 11.   Fuzzy control input variables. Lateral and angular errors, distance to the bend, and actual longitudinal speed.

route so that the variation of one of the input variables ("distance to the bend") has the same shape at different speeds. Fig. 11 shows the distances to the nearest bend and the vehicle's actual speed for the experiment with a reference speed of

16 km/h. The negative values of the distance to the bend mean that the vehicle has passed that curve, and these values instantaneously change to positive again when the vehicle reaches the midpoint of the straight stretch and a new reference point is taken.

In Fig. 11, one observes how, when the vehicle is less than 5 m on either side from the center point of the curve (an interval taken to allow for the length of the vehicle and the width of the road), the "distance to the bend" is set to zero. This is because, in this interval, the vehicle is turning at its maximum for that curve, and the readings of the GPS are rapidly varying; therefore, with a trapezoidal instead of triangular shape for the "Central" label of the membership function (see Fig. 7), we avoid any unnecessary sudden changes due to this situation when the vehicle reads the next straight segment.

Finally, Table II presents the root-mean-square error (RMSE) with respect to the reference line, where the curve segments are generated using quadratic Beziers curves [34]. It shows the RMSE for the whole route, as well as for the straight stretches and bends. One can appreciate that the greatest error–in bends–is lower than 1 m.

TABLE II
RMSE WITH RESPECT TO THE REFERENCE LINE

Stretch	Speed (km/h)				
	8	12	16	20	24
Straight	0.370	0.515	0.584	0.239	0.513
Bend	0.963	0.974	0.834	0.780	0.996
Total	0.706	0.774	0.716	0.569	0.791

V. CONCLUSION

We have described a study of the implementation of a cascade control architecture of the steering wheel for an autonomous vehicle, using intelligent and classical controls. The original hydraulic power steering system of a Citroen Berlingo was equipped with a dc motor and controlled by an external motion controller with a discrete PID. The control scheme was closed with an external fuzzy control loop that drives the steering wheel position and its angular speed using a DGPS as the main input sensor.

This control architecture gave good results for different vehicle speeds and curves. The PID controller implemented in the inner control loop smoothes out any sudden changes in the fuzzy control output signals, which is particularly important at low speeds. In future work, other kinds of input sensor may be added to the control architecture.

The angular speed controller showed good performance. Indeed, the results allow one to draw two important conclusions. While with the previous controller the steering wheel was always turned at the maximum speed, now, it is turned only slowly when the vehicle itself is moving at low speeds. Furthermore, the movement of the steering wheel has more degrees of freedom when the vehicle is close to or in a curve or is negotiating a curve at a greater speed, allowing it to adapt to the requirements of the situation, and less freedom when the vehicle is traveling on a straight segment.

The speeds used in the present experiments exceeded those used in previous work for curved segments. The reason that this was possible was the inclusion of the angular speed, as well as the position, as a control output parameter. The use of the four input variables, i.e., the angular error, the lateral error, the distance to the bend, and the actual speed, in the fuzzy control loop avoids having to implement different driving modes (the previous application involved two fuzzy controllers, i.e., one for curves and one for straight sections). With this system, we now have a single fuzzy controller for the steering wheel whose angular speed control allows us to drive fast along straight stretches while guaranteeing only smooth adjusting movements of the wheel and to safely negotiate curves with rapid variations if necessary. The RMSE values show the good behavior of the system.

REFERENCES

[1] T. Vaa, M. Penttinen, and I. Spyropoulou, "Intelligent transport systems and effects on road traffic accidents: State of the art," *IET Intell. Transp. Syst.*, vol. 1, no. 2, pp. 81–88, Jun. 2007.

[2] K. Gardels, "Automatic car controls for electronic highways," General Motors Res. Lab., Warren, MI, GMR276, 1960.

[3] D. Barrick, "Automatic steering techniques," *IRE Int. Conv. Rec.*, vol. 10, pt. 2, pp. 166–178, 1962.

[4] M. Hanson, *Projest Metran: An Integrated, Evolutionary Transportation System for Urban Areas*. Cambridge, MA: MIT Press, 1966.

[5] H. Cheng, N. Zheng, X. Zhang, J. Qin, and H. Van de Wetering, "Interactive road situation analysis for driver assistance and safety warning systems: Framework and algorithms," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 157–167, Mar. 2007.

[6] B. van Arem, C. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

[7] J. Yoon, W. Cho, B. Koo, and K. Yi, "Unified chassis control for rollover prevention and lateral stability," *IEEE Trans. Veh. Technol.*, vol. 58, no. 2, pp. 596–609, Feb. 2009.

[8] J.-S. Lin and W.-E. Ting, "Nonlinear control design of anti-lock braking systems with assistance of active suspension," *IET Control Theory Appl.*, vol. 1, no. 1, pp. 343–348, Jan. 2007.

[9] L. Bi, O. Tsimhoni, and Y. Liu, "Using image-based metrics to model pedestrian detection performance with night-vision systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 1, pp. 155–164, Mar. 2009.

[10] S. Gehrig and F. Stein, "Collision avoidance for vehicle-following systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 233–244, Jun. 2007.

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

[12] T. B. J. Ackermann and D. Odenthal, *Advantages of Active Steering for Vehicle Dynamics Control*. Cologne, Germany: German Aerospace Center, 1999, 99ME013.

[13] C. Unsal, J. S. Bay, and P. Kachroo, "Intelligent control of vehicles: Preliminary results on the application of learning automata techniques to automated highway system," in *Proc. 7th Int. Conf. Tools Artif. Intell.*, Nov. 5–8, 1995, pp. 216–223.

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

[15] H.-S. Tan, F. Bu, and B. Bougler, "A real-world application of lane-guidance technologies—Automated snowblower," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 538–548, Sep. 2007.

[16] R. White and M. Tomizuka, "Autonomous following lateral control of heavy vehicles using laser scanning radar," in *Proc. Amer. Control Conf.*, Jun. 25–27, 2001, vol. 3, pp. 2333–2338.

[17] S. B. Choi, "The design of a look-down feedback adaptive controller for the lateral control of front-wheel-steering autonomous highway vehicles," *IEEE Trans. Veh. Technol.*, vol. 49, no. 6, pp. 2257–2269, Nov. 2000.

[18] Y. Hayakawa, R. White, T. Kimura, and G. Naito, "Driver-compatible steering system for wide speed-range path following," *IEEE/ASME Trans. Mechatronics*, vol. 9, no. 3, pp. 544–552, Sep. 2004.

[19] X. Chen, T. Yang, X. Chen, and K. Zhou, "A generic model-based advanced control of electric power-assisted steering systems," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1289–1300, Nov. 2008.

[20] B. A. Guvenc and L. Guvenc, "Robust two degree-of-freedom add-on controller design for automatic steering," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 1, pp. 137–148, Jan. 2002.

[21] P. Yih and J. C. Gerdes, "Modification of vehicle handling characteristics via steer-by-wire," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 6, pp. 965–976, Nov. 2005.

[22] M. Tai, P. Hingwe, and M. Tomizuka, "Modeling and control of steering system of heavy vehicles for automated highway systems," *IEEE/ASME Trans. Mechatronics*, vol. 9, no. 4, pp. 609–618, Dec. 2004.

[23] T. Bauml, H. Giuliani, D. Simic, and F. Pirker, "An advanced simulation tool based on physical modelling of electric drives in automotive applications," in *Proc. IEEE VPPC*, Sep. 9–12, 2007, pp. 736–741.

[24] L. Cardamone, D. Loiacono, and P. L. Lanzi, "On-line neuroevolution applied to the open racing car simulator," in *Proc. IEEE CEC*, May 18–21, 2009, pp. 2622–2629.

[25] S. Chaib, M. S. Netto, and S. Mammar, " H_{infty} adaptive, PID and fuzzy control: A comparison of controllers for vehicle lane keeping," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 14–17, 2004, pp. 139–144.

[26] J. E. Naranjo, C. Gonzalez, R. Garcia, T. de Pedro, and R. E. Haber, "Power-steering control architecture for automatic driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 4, pp. 406–415, Dec. 2005.

[27] J. E. Naranjo, C. Gonzalez, R. Garcia, and T. de Pedro, "Lane-change fuzzy control in autonomous vehicles for the overtaking maneuver," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 3, pp. 438–450, Sep. 2008.

[28] J. Perez, C. Gonzalez, V. Milanes, E. Onieva, J. Godoy, and T. de Pedro, "Modularity, adaptability and evolution in the AUTOPIA architecture for control of autonomous vehicles," in *Proc. IEEE ICM*, Apr. 14–17, 2009, pp. 1–5.

[29] V. Milanés, J. Pérez, E. Onieva, C. González, and T. de Pedro, "Lateral power controller for unmanned vehicles," *Elect. Rev.*, vol. 86, no. 1, pp. 207–211, Jan. 2010.

[30] R. García, T. de Pedro, J. Naranjo, J. Reviejo, and C. Gonzalez, "Frontal and lateral control for unmanned vehicles in urban tracks," in *Proc. IEEE IV*, Versailles, France, 2002, pp. 583–588.

[31] K. Xie, W. Hao, and J. Xie, "Superheated steam temperature cascade control system based on fuzzy-immune PID," in *Proc. 4th Int. Conf. FSKD*, Aug. 24–27, 2007, vol. 2, pp. 624–628.

[32] J. Wang, Y. Jing, and C. Zhang, "Robust cascade control system design for central airconditioning system," in *Proc. 7th WCICA*, Jun. 25–27, 2008, pp. 1506–1511.

[33] Z. Y. Guo and T. C. Hsia, "Joint trajectory generation for redundant robots in an environment with obstacles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 13–18, 1990, pp. 157–162.

[34] I. Skog and P. Händel, "In-car positioning and navigation technologies—A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 3, pp. 4–21, Sep. 2009.

Vicente Milanés was born in Badajoz, Spain, in 1980. He received the B.E. and M.E. degrees in electronic engineering from the Extremadura University, Caceres, Spain, in 2002 and 2006, respectively, and the Ph.D. degree in electronic engineering from the Alcala University, Madrid, Spain, in 2010.

Since 2006, he has been with the Industrial Computer Science Department (currently the Centre of Automation and Robotics), Instituto de Automática Industrial, Consejo Superior de Investigaciones Científicas, Madrid. His research interests include autonomous vehicles, fuzzy-logic control, and intelligent transportation systems.

Joshué Pérez was born in Coro, Venezuela, in 1984. He received the B.E. degree in electronic engineering from the Simón Bolívar University, Caracas, Venezuela, in 2007 and the M.E. degree in system engineering and automatic control from the University Complutense of Madrid, Madrid, Spain, in 2009. He is currently working toward the Ph.D. degree with the Centre of Automation and Robotics, Consejo Superior de Investigaciones Científicas, Madrid.

His research interest includes fuzzy logic, modeling, control, and cooperative maneuvers among autonomous vehicles.

Enrique Onieva was born in Priego de Córdoba, Spain, in 1983. He received the B.E. degree in computer science engineering and the M.E. degree in soft computing and intelligent systems from the University of Granada, Granada, Spain, in 2006, and 2008, respectively.

Since 2007, he has been with the Industrial Computer Science Department (currently the Centre of Automation and Robotics), Instituto de Automática Industrial, Consejo Superior de Investigaciones Científicas, Madrid, Spain. His research interests include autonomous vehicles, fuzzy-logic control, and intelligent transportation systems.