Soft Computing and Geometrical Control for Computer Aided Driving

Javier Alonso Ruiz, Teresa de Pedro, Carlos González and Ricardo García

Industrial Automation Institute, CSIC, La Poveda, Arganda del Rey.
28500 Madrid, Spain
{jalonso, tere, gonzalez, ricardo}@iai.csic.es

Abstract. After having designed control systems for real autonomous cars in an urban environment using straight lines as reference [2], we are now trying to build a fuzzy control system based on clothoids [3], the curve used in roads and train tracks. This paper proposes a method based on soft computing and upgraded using genetic algorithms. Both sets of simulations are shown and compared.

1. Introduction

This paper is based on our previous research into real autonomous car controllers in urban environments. These control systems uses straight lines as reference lines [2]. Now, we propose a fuzzy control system based on clothoid curves [3] to navigate the bend sections of the road. Our goal is to improve the car's behaviour on bends by reducing steering wheel efforts. This improvement will reduce electrical consumption. And a decrease in excessive steering wheel movements will increase the comfort feeling.

The system designs a course map as a succession of straight lines and bends. Each bend is planned as a soft curve with a continuous and smooth curvature progression. The fuzzy controller uses this soft curvature progression to navigate on bends. To be more precise this differential equation system is shown in [4] relating the state and control variables and its integration equations for admissible paths:

$$\begin{pmatrix} x' & y' & \theta' & \kappa' \end{pmatrix} = v \cdot \begin{pmatrix} \cos\theta & \sin\theta & \kappa & 0 \end{pmatrix} + \sigma \cdot \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix} \qquad \textbf{(1)}$$

where (x y θ κ) stand for attitude and curvature, and σ is $\sigma = \kappa' = \Phi/\cos^2\Phi$ (Φ is the wheel angle). Our proposal is to express the position of the car depending on curvature, f(x,y,κ,v,t), related to the steering angle, and to control the car using its usual inputs, acceleration and curvature (v', κ=1/r). This curvature-based fuzzy logic control method has been optimized using genetic algorithms.

2. Trajectory Design

The car's course is considered as a succession of straight-line and bend sections. For each bend section, there are three reference points, the starting point, the ending point and one point from the middle of the curve. Therefore, the straight-line sections start at the end of a bend section and end at the beginning of the next one. As the fuzzy control system has been tested extensively on straight-line sections [2], there will be no further references.

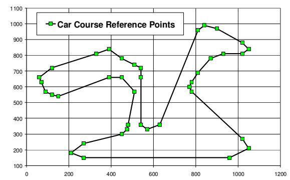

Fig. 1. Jerez circuit car course reference points.

The trajectory on the bend sections is designed in the same way as bends in roads are and should be as smooth as possible [6]. Therefore, the trajectory changes its curvature as it advances in the bend. At the beginning of the bend the curvature is zero. It then starts to increase until it reaches a certain value. The curvature then remains constant until, in the last part, it falls to zero. The clothoid spiral is used to ensure that this transition to the circular curve is smooth. The clothoid, also known as Cornu's spiral or Euler's spiral, makes a perfect transition spiral, as its curvature increases linearly with distance along the spiral. The curvature of a clothoid spiral is linearly related to its arc length. And, when the path of the bend is followed at a uniform velocity, the speed of rotation is linear. The curve with the reverse relation is called the anti-clothoid and is used to return from the circular curve to the next straight-line section.

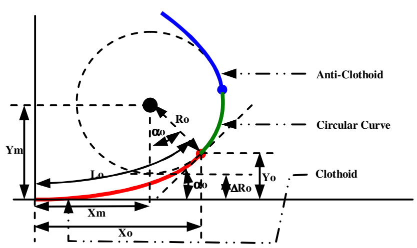

Fig. 2. Transition from the approaching clothoid to the circular curve, and the clothoid parameters.

The parameters shown are the circumference radius (R_o), the approaching transition clothoid total length (L_o), the circular curve minimum distance to the initial straight line (ΔR_o), the coordinates of the tangency point between the clothoid and the circumference (X_o,Y_o), the coordinates of the circumference center (X_m,Y_m), and the angle between the initial straight-line segment and the tangent line at the tie point of the two curves (the clothoid and the circular curve) (α_o).

Each bend has its own curvature radius, but if this parameter is not available, the circular curve radius is obtained from the distance between the starting and the ending point of the bend. A clothoid curve is used to reach the circular curve. Its intrinsic equation is:

$$R \cdot L = A^2 \qquad (2)$$

where R is the curvature radius at one point, L is the curve length between its inflexion point (where R is infinite) and the point of radius R, and A is the clothoid characteristic parameter. As soon as the maximum speed on the curve is fixed, the value of the transition clothoid characteristic parameter A (in meters), and the clothoid minimum length L can be inferred [6].

Once the clothoid parameters have been obtained, the center of the circumference is calculated. A distance d is defined as the sum of the circumference radius and the circular curve minimum distance to the initial straight line ($d = R_o + \Delta R_o$). The crossing point (c0) of the parallel lines at distance d to the entrance and exit straight lines is obtained. This point will be the center of the circumference that contains the circumference arc. Once the center has been found, the perpendicular lines to the entering and exiting straight lines that go through this circumference center determine the points (n1, n2), and the starting points of the clothoids (m1, m2).

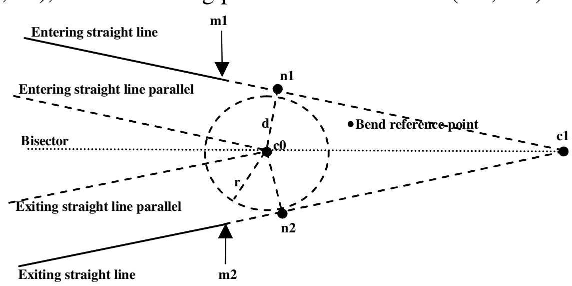

Fig. 3. Initial circumference layout.

To guarantee that the designed curve goes through a given bend reference point, the center of the circumference will be moved throughout the bisector of the entering and exiting straight lines just before the calculation of m, and n points.

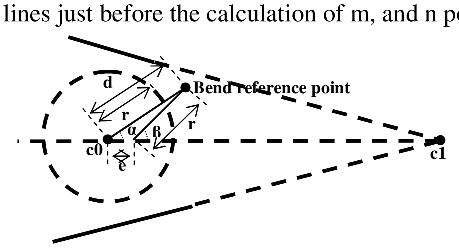

Fig. 4. Center of the circumference adjustment.

The distance moved (e) depends on the distance between the circumference center(c0) and the bend reference point (d), the circumference radius (r) and the angles between the bisector of the entering and exiting straight lines, and the lines that go to the bend reference point (α and β).

$$d \cdot \sin\alpha = r \cdot \sin\beta \qquad d \cdot \cos\alpha = e + r \cdot \cos\beta \tag{3}$$

This is solved:

$$\beta = \arcsin\left(\frac{d}{r}\sin\alpha\right) \qquad e = d\cdot\cos\alpha - r\cdot\cos\beta \tag{4}$$

Moving a distance e to the center of the circumference, the designed curve goes through the intermediate point of the curves that has been provided.

Once all the necessary points have been found, the length (in degrees) of the circumference arc is calculated. This measure is calculated by reducing the angle formed by the two straight-line sections from 180º (U-turn) and, twice, the angle turned in the approach curves [6] (clothoids). Thus the vehicle will follow the entering straight line (with null curvature) as far as the point m1 at which the clothoid begins. Its curvature will increase until the circumference (to 1/radius curvature) is reached. Next, it will turn the previously calculated number of degrees, its radius of curvature remaining constant. And then it will reduce that curvature progressively until it reaches point m2.

3. Fuzzy Controller

We have used a fuzzy logic control system to get a solution that is as smooth as possible [1]. The state of the system is described with different sets of variables depending on the car's desired behavior. This article focuses on bend sections control only, because the straight-line sections control has been already presented in other papers [1], [2] and [5]. So, if the car is tracking a bend, the variables are based on the curvature. This control system must maintain the curvature goal that depends on the point of the curve where the car is. The control variables used are the curvature error and the error accumulated on that bend (a fuzzy PI fuzzy control). The car's curvature is calculated from the steering wheel position and the distance between the front and rear axes of car. These variables are "*Curvature error",* the difference between the car's desired and real curvature, and "*Curvature error summation",* the accumulated curvature error.

The input fuzzy linguistic variables will have only one fuzzy partition, but we will use the fuzzy modifiers LESS THAN and MORE THAN for operations, as we have three fuzzy partitions. Its fuzzy membership functions will be very simple to reduce the computational cost in a real-time car controller.

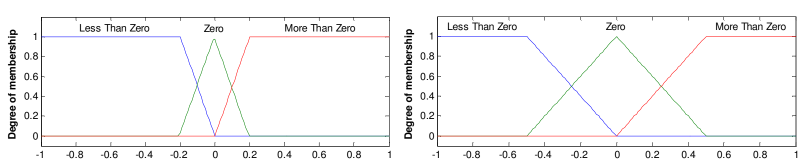

Fig. 5. The membership functions of "Curvature error" and "Curvature error summation" linguistic variables

The fuzzy rules for the bend sections are:

Table 1. Bend section fuzzy rule base.

Rule	IF	Precedent	THEN	Consequent
7	**IF**	Curvature_error MORE_THAN (zero)	**THEN**	Steering negative
8	**IF**	Curvature_error zero	**THEN**	Steering unchanged
9	**IF**	Curvature_error LESS_THAN (zero)	**THEN**	Steering positive
10	**IF**	Sum_Curvature_error MORE_THAN (zero)	**THEN**	Steering negative
11	**IF**	Sum_Curvature_error zero	**THEN**	Steering unchanged
12	**IF**	Sum_Curvature_error LESS_THAN (zero)	**THEN**	Steering positive

The system actuation is defined by a goal position for the steering wheel and another goal position for the accelerator-brake set [5]. The output variable that controls the steering wheel has three normalized singleton values: negative (-1), unchanged (0), and positive (+1). And its surface control obtained is:

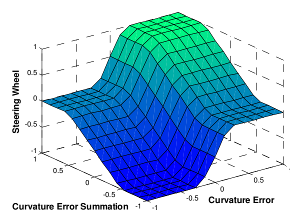

Fig. 6. Steering wheel control surface.

4. GA Optimization

The control system performance is improved by modifying the membership function parameters. The objectives of the genetic algorithm fitness function are to get a control as smooth as possible (minimize steering wheel movements). Thus, the effort is defined as the difference between consecutive steering wheel goals. To ensure that the car trajectory still matches the car's designed path, the maximum error on bend sections is added to the fitness function. Both objectives are combined as follows:

$$\textit{Fitness} = k1 * \textit{effort} + k2 * \textit{Bends\_maximum\_error} \quad (5)$$

We have obtained different optimized set values depending on the nature of the circuit curves (motorways with small curvature or race circuits).

5. Experimental Results and Conclusions

The control system performance improvement can be seen from the simulator results. The simulator uses detailed maps of competition circuits, primarily Jerez, keeping its real proportions. And it takes into account the mechanical characteristics of our CITROËN cars. Fig. 7 left shows the simulated steering wheel effort results without optimization (top line) and after the application of the genetic algorithm (bottom line). In Fig. 7 right, the car course in the simulations goes through all the reference points and is adjusted to the circuit dimensions. Therefore, the car remains on the road.

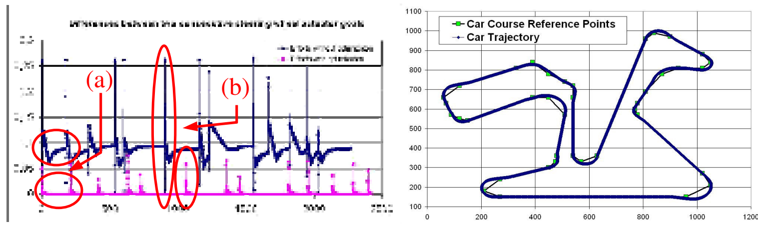

Fig. 7. On left: Differences between two consecutive steering wheel actuator goals. **On right:** Plotting a course on the Jerez circuit map

The effort reduction along each bend (Fig. 7a) ensures that electrical consumption will be reduced and the comfort feeling will increase. And the effort reduction at the starting and ending point of each bend (Fig. 7b) ensures that the expected lifetime for the steering wheel motor will be lengthened considerably. This also contributes to a reduction in electrical consumption, and an increase in the comfort feeling.

6. Acknowledgments and References

This work was supported by the Spanish Ministry of Education and Science ISAAC project, DPI2002-04064-C05-02, and the Spanish Ministry of Development COPOS project, BOE 280, November 22, Res. 22778.

[1] García R and de Pedro T, "First Application of the ORBEXoprocessor : Control of Unmanned Vehicles" *Mathware and Soft Computing*, no. 7, vol 2-3, pp 265-273, 2000
[2] García R, de Pedro T, Earanjo J E, Reviejo J and González C, "Frontal and Lateral Control for Unmanned Vehicles in Urban Tracks" *IEEE Intelligent Vehicle Symposium (IV2002)*
[3] Jiménez Shaw J, "Resolución numérica completa de la Ecuación de la Clotoide", web page and Clothos constants calculation program. http://javier.jimenezshaw.com/
[4] Laugier C and Fraichar T, "Decisional Architectures for Motion Autonomy" chapter 11 in *Intelligent Vehicle Technologies*, editors: Vlacit, ParenT, Harashima, SAE International, ISBN 0 7680 0780 I
[5] Naranjo J E, Reviejo J, Gonzalez C, García R, de Pedro T, "A Throttle & Brake Fuzzy Controler: Towards the Automatic Car" *Lecture Notes in Computer Science: Computer Aided Systems Theory – Eurocast 2003*, LNCS 2809 pp 291-301
[6] Web page www.carreteros.org, Norm 3.1-IC "Trazado"