Virtual Environment for Control Design and Evaluation with Real Driver Assessment

M.A. Naya*, D. Dopico, M. Gonzalez, J. Cuadrado

***Escuela Politecnica Superior, Universidad de La Coruña**
Mendizabal s/n, 15403 Ferrol, Spain
Phone: +34-981337400; E-mail: minaya@cdf.udc.es

Keywords: driving simulator, real-time dynamics, fuzzy logic control, human-in-the-loop, virtual reality.

1. Introduction

This paper reports on the development of a low-cost and efficient computational framework for the design of automobile motion controllers [1]. The starting point of the work has been an existing prototype car, whose virtual counterpart has been implemented in Fortran language through an authors' dynamic formulation. Virtual reality capabilities have been given to the program by means of a realistic graphical output (either on screen or HMD) and game-type driving peripherals (steering wheel and pedals), in order to enable human participation both in the controller design and evaluation processes. Fuzzy logic has been chosen for the design of the control algorithms, which have been implemented on the Matlab environment. Figure 1 illustrates the general structure of the computational tool developed.

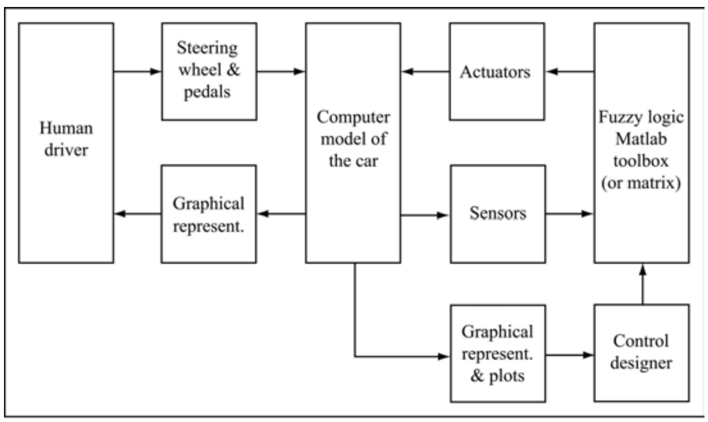

Figure 1. Computational framework for controller design.

2. The computational model of the car

The mathematical description of the car, illustrated in Fig. 2 along with the physical prototype, has been carried out in natural coordinates. As seen in the figure, the coordinates have been used to model the chassis and the steering and suspension systems.

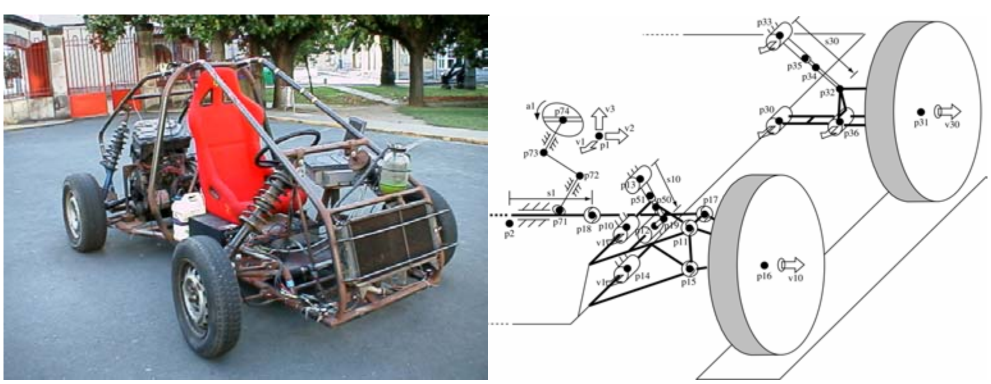

Figure 2. The prototype and its model.

The steering wheel is kinematically guided. The chassis inertial parameters have been obtained from a CAD model, while those of the engine have been experimentally estimated. Suspension forces have been considered through linear springs and dampers. Tire forces have been introduced through the *Magic Formula*. Power transmission forces are derived from the torque-speed engine relationships and the gear ratios, including engine braking torque at closed throttle position. The braking torque has been estimated from disk geometry.

The dynamic formalism [2] consists of an index-3 augmented Lagrangian formulation with projections of velocities and accelerations, which is combined with the trapezoidal rule as numerical integrator, and features sparse matrix technology. A code that calculates the dynamics of the described model according to the indicated formalism has been implemented in Fortran language, due to its high efficiency.

3. The virtual reality interface

The described car model has been used to create a driving simulator, shown in Fig. 3, by combining the already mentioned vehicle dynamics code, with a realistic graphical output (perceived by the user either through the computer monitor or through a HMD) and game-type driving peripherals (steering wheel and pedals).

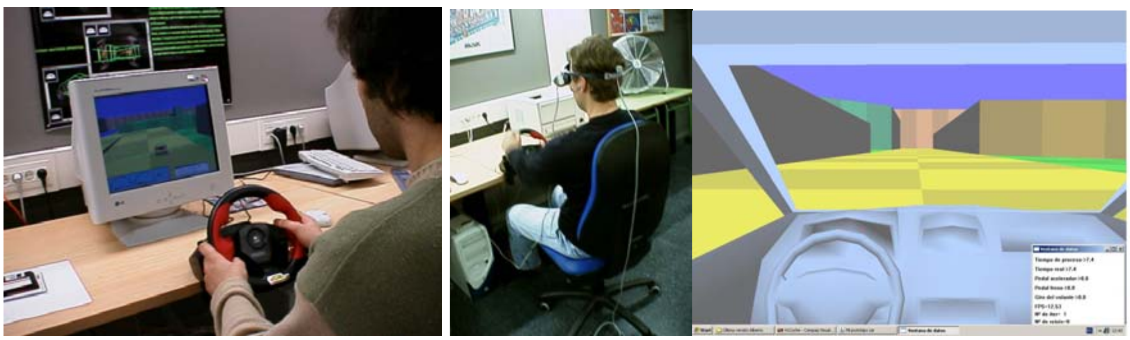

Figure 3.Car driving simulator: monitor (left) and HMD (right).

Such a virtual reality interface serves to several purposes. Prior to the control design stage, a human driver is expected to carry out the targeted maneuver on the simulator, in order to provide an initial assessment on maneuver feasibility and realistic values of the involved parameters: time required to perform the maneuver, maximum speed for stability, more challenging parts of the maneuver, etc. Moreover, the performance achieved by the human driver may be used as a reference to measure the quality of the successive controllers obtained during the design stage, until the desired behavior is attained. Additionally, when the aim is to partially control the motion of the car (as opposed to cases in which the total automatic control of the car is pursued), both the human driver and the controller must simultaneously operate (see Fig. 1): this is also enabled by the virtual reality interface. Furthermore, the graphical representation of the car motion along with the plots provided by the program, are essential elements which allow the designer to tune the controller parameters until a satisfying result is achieved, as illustrated in Fig. 1.

4. The control module

The controllers, based on fuzzy logic, are implemented by using the corresponding toolbox of Matlab. They receive information from virtual sensors defined in the computational model of the car, and act upon actuators, also defined in the virtual model, as illustrated in Fig. 1. Fuzzy logic controllers have been developed by applying Mamdani-type inference and its typical defuzzification process [2], finding the centroid of a two-dimensional function to determine the value of the output variable from its membership to the output fuzzy set.

To illustrate the use of the proposed tool, controllers have been designed for two maneuvers of the car. The first maneuver, a 20 m straight path, is a very simple one, only requiring throttle-brake control. The second maneuver may be referred to as obstacle avoidance, and needs both throttle-brake and steering control.

5. References

[1] P. Antos, J. Ambrosio, "A control strategy for vehicle trajectory tracking using multibody models", *Multibody System Dynamics*, Vol. (11), 365-394, (2004).

[2] J. Cuadrado, R. Gutiérrez, M.A. Naya, M. González, "Experimental validation of a flexible MBS dynamic formulation through comparison between measured and calculated stresses on a prototype car", *Multibody System Dynamics*, Vol. (11), 147-166, (2004).