6. Applications

Once all the models are implemented, they can finally be compared, evaluated and joined together to reproduce human driving experience. In this chapter the developed computational procedures are compared to other methods. Finally, a trajectory optimized with Genetic Algorithms is used as a reference to the Fuzzy Controller.

A trajectory determination method based on classic optimization and developed by [11] is compared to the Evolutionary Optimization presented in Chapter 4..

A comparison between a linear PDD Directional Controller and the Fuzzy Driver is also presented. The Stationary Kinematic Model is tested with both controllers for the same reference trajectories.

6.1. Trajectory Optimization Methods Comparison

In order to analyze how classical and GA optimization work with adjacent opposite curves, the circuit defined is an “S” chicane. Figure 6.1 shows the Center Line trajectory for this track and also its lap time and accomplished distance.

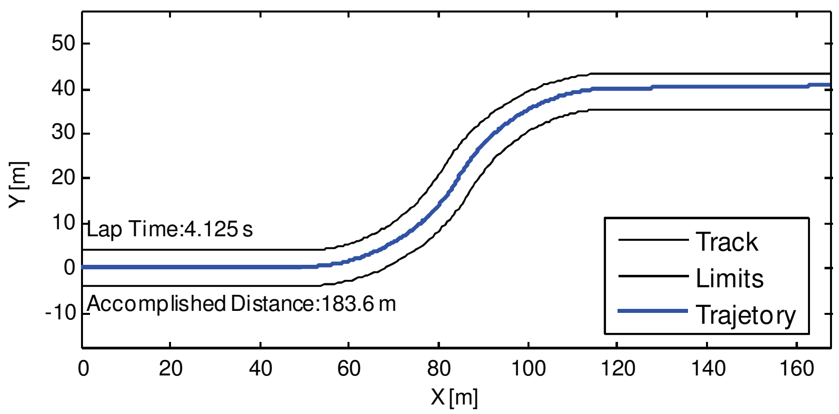

Figure 6.1 – “S” Chicane: Center Line Trajectory.

For all the simulations ahead, initial conditions for position, linear and angular velocities and accelerations are set to zero.

6.1.1. Classical Optimization

Classical optimization methods are based on the objective function's gradient. Initial tests with the traditional optimization method were presented by Hernan [11]; Figure 6.2 shows result for the same "S" chicane track shown in Figure 6.1. The trajectory is clearly better than the Center Line trajectory given as the initial step. However, from the acceleration profiles shown in Figure 6.3, it can be noticed that there was almost no effect on the Tangent Acceleration (a_T) profile.

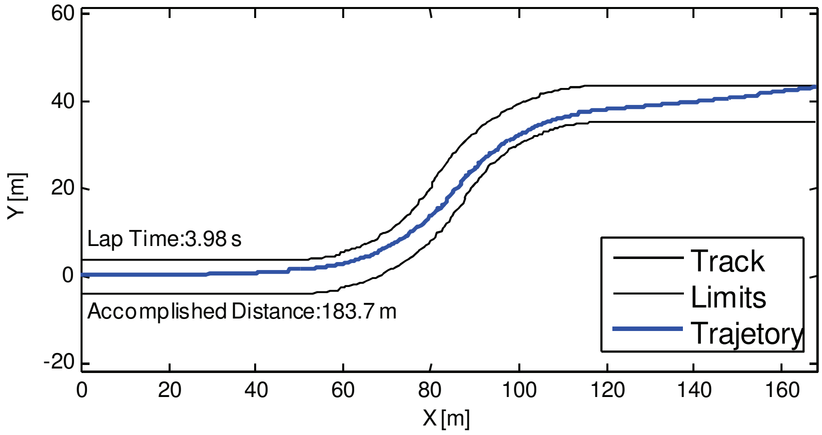

Figure 6.2 – Classical Optimization 1st Test: Obtained Trajectory.

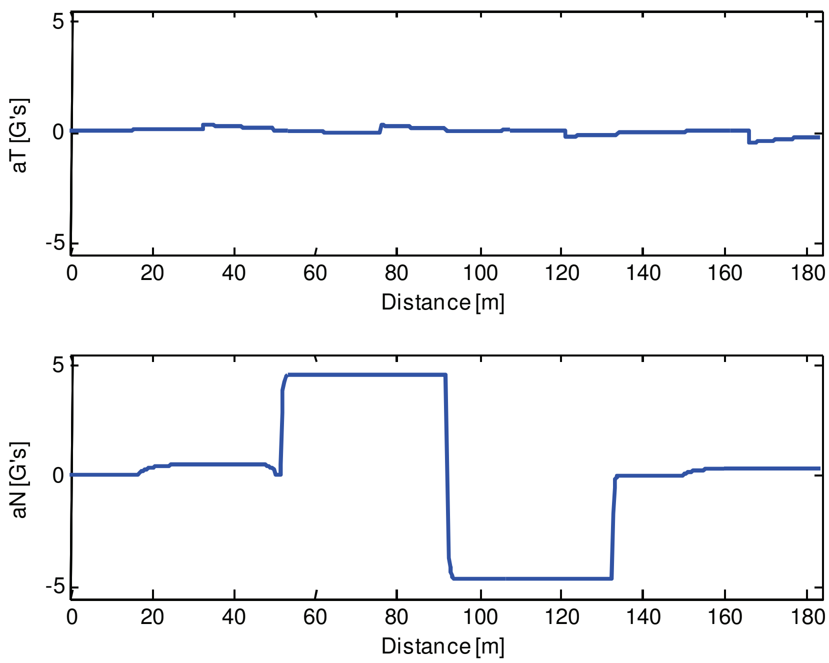

Figure 6.3 – Classical Optimization 1st Test: Acceleration Profiles.

6.1.2. Genetic Optimization

Optimization by Genetic Algorithms gave origin to the results shown in Figure 6.4.

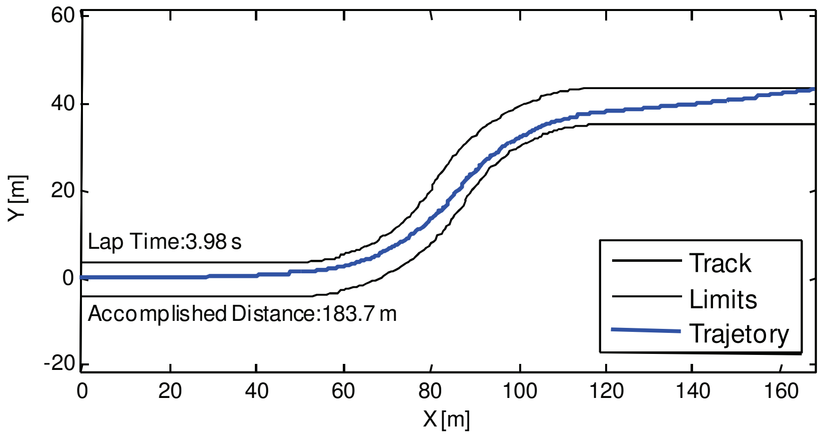

Figure 6.4 – Genetic Optimization 1st Test: Obtained Trajectory.

As shown Figure 6.5, acceleration profiles present a sensible change in the longitudinal acceleration. Therefore, as shown in Figure 6.6, the speed increases and the lap time is 3.98s.

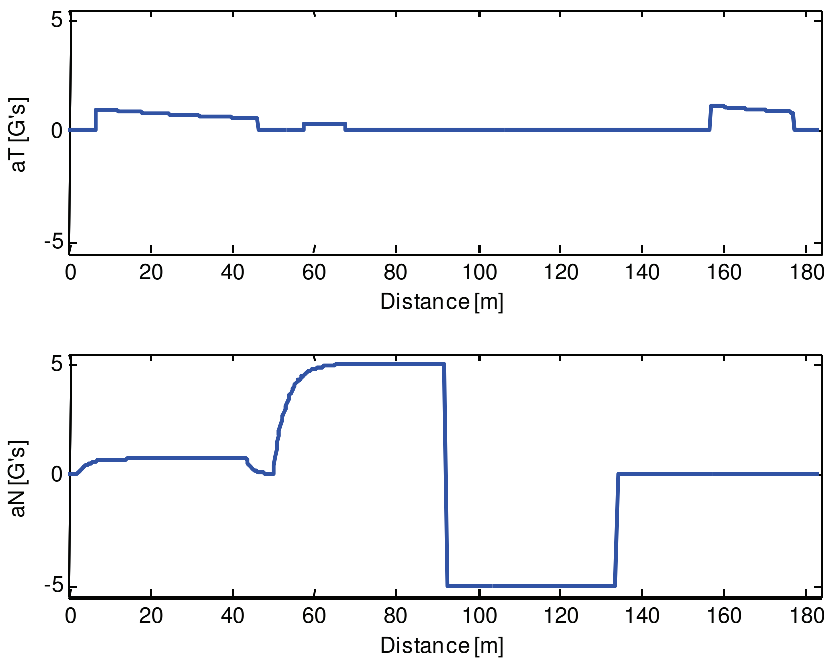

Figure 6.5 – Genetic Optimization 1st Test: Acceleration Profiles.

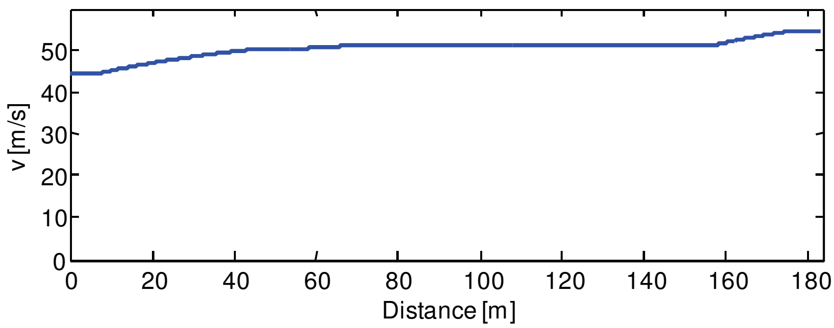

Figure 6.6 – Genetic Optimization 1st Test: Longitudinal Speed.

6.2. Trajectory Error Calculation Comparison

A procedure for the calculation of the trajectory error was developed in Chapter 3, with the purpose of modeling human perception of the car's position related to the desired trajectory. Two different models were implemented and tested and both are compared here.

6.2.1. Present-based Trajectory Error

The first simulation covers the Fuzzy Controller block with the Present-based Trajectory Error, as shown in Figure 6.7. The driver, emulated by the Fuzzy Controller, only notices a change in the trajectory when this actually happens. The test was made at *25 m/s* in a track model of the Catalunya circuit in Barcelona, as seen in Figure 6.8.

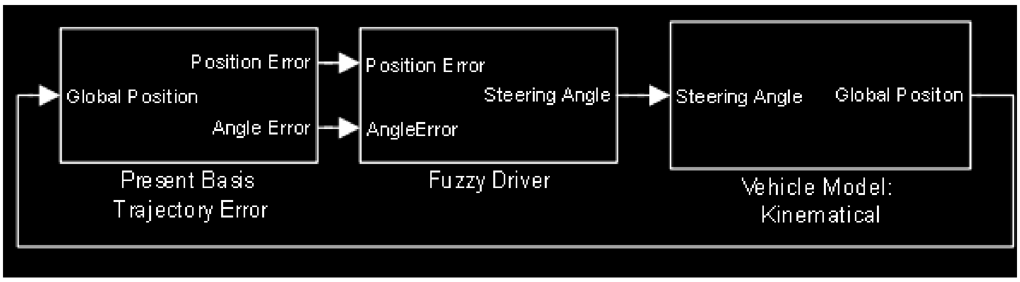

Figure 6.7 – Present-based Trajectory Error and Fuzzy Controller: Block Diagram.

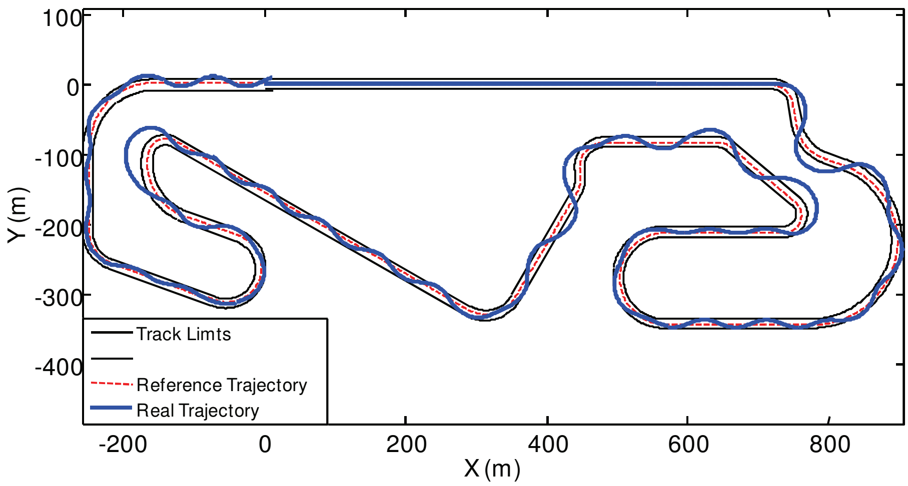

Figure 6.8 – Present-based Trajectory Error and Fuzzy Controller: Obtained Trajectory.

The bad performance is due to the absence of future information. At lower speeds, the controller receives the trajectory information from the error procedure with a higher frequency, presenting a minor oscillatory behavior. However, at higher speeds, if a track change is not expected by the controller, it is unable to take a proper action.

6.2.2. Future-based Trajectory Error

The second test considers the Future-based Trajectory Error block and the Fuzzy Controller block, shown in Figure 6.9. The resulting trajectory is plotted in Figure 6.10.

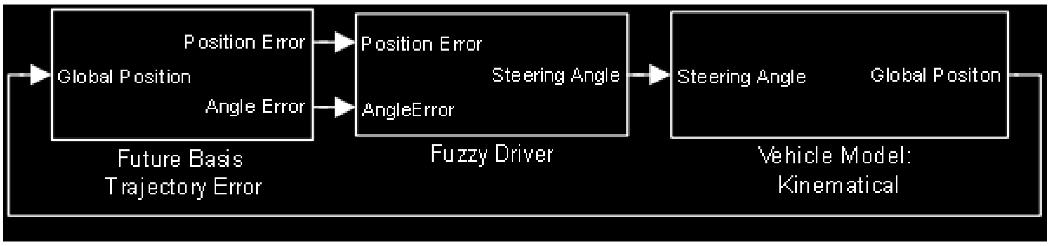

Figure 6.9 – Future-based Trajectory Error and Fuzzy Controller: Block Diagram.

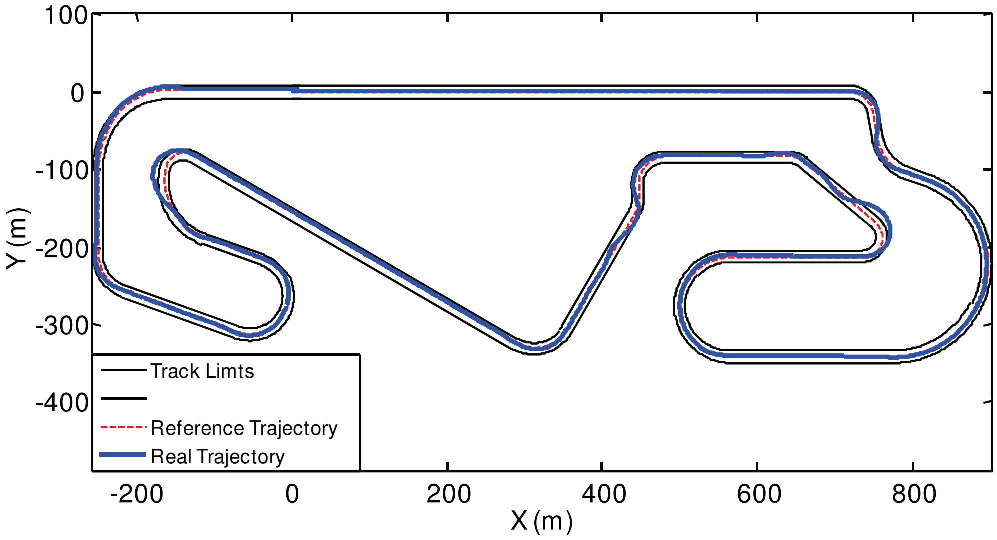

Figure 6.10 – Future-based Trajectory Error and Fuzzy Controller: Obtained Trajectory.

This model is capable of analyzing future points of a trajectory and takes into account their deviation from the car's tendency of movement. The result is a smoother trajectory, with less deviations and a shorter lap time.

After parameters calibration – such as the step size, δ_i, and the number of forward steps considered, n_{FS} – the following experiments consider the Fuzzy Controller and the Future-based Trajectory Error.

6.3. Vehicle Controllers Comparison

Considering the vehicle control problem, several different approaches were previously adopted in the Vehicle's System Group. Hey [12] proposed and simulated a Classical Controller applied to a Kinematic Vehicle Model; those results are used here for comparison.

6.3.1. PDD Controller

Figure 6.11 shows the reference and the trajectory obtained from a simulation that assembles the stationary kinematic vehicle model and a PDD controller presented [12]. As this model operates with constant speed and this study is aimed at evaluating racing cars, the chosen velocity for analysis is 50m/s, (180km/h).

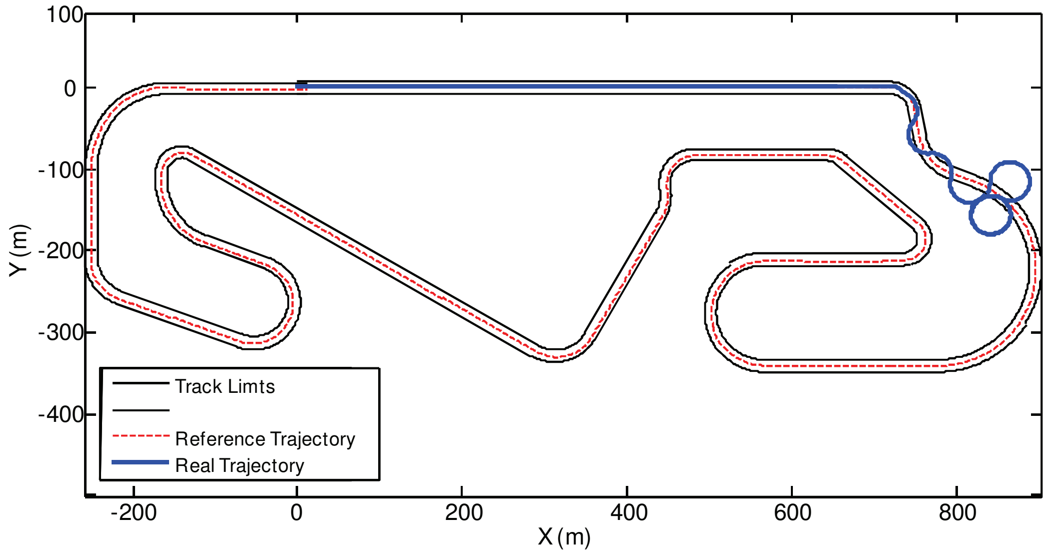

Figure 6.11 – PDD Controller High Speed Test: Trajectory.

The system becomes unstable at high speeds, and the more it tries to correct the trajectory, the larger the oscillations are. As shown in Figure 6.11, the vehicle diverges from the reference trajectory at a curve, and starts to execute an 8-shaped trajectory.

6.3.2. Fuzzy Driver

Figure 6.12 shows the Fuzzy Driver's trajectory at the same speed and applied to the same vehicle model. The fuzzy controller manages to follow adequately the whole reference trajectory and, despite some oscillatory behavior due to high speed, it leaves the track limits only a few times.

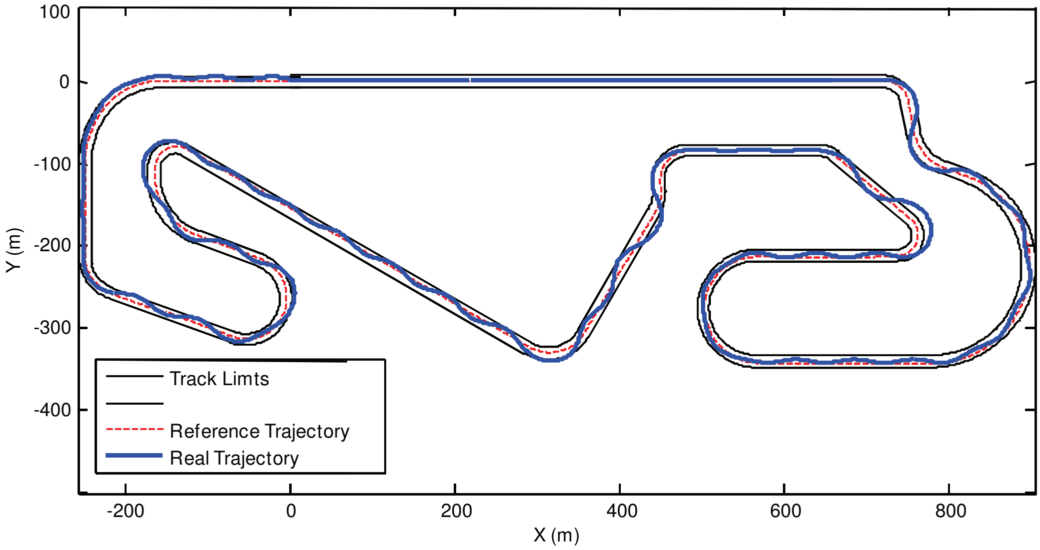

Figure 6.12 – Fuzzy Driver High Speed Test: Trajectory.

6.4. Fuzzy Driver Applied To Genetic Optimized Trajectory

The abovementioned solutions were tested and evaluated in comparison to previously used techniques. Genetic optimization and the Fuzzy Controller can easily be assembled with the purpose of defining the optimal trajectory and also control the vehicle.

The array of position coordinates obtained from the optimization algorithm is the reference trajectory for the controller. Figure 6.13 shows the center line initial trajectory to be optimized by the genetic algorithm.

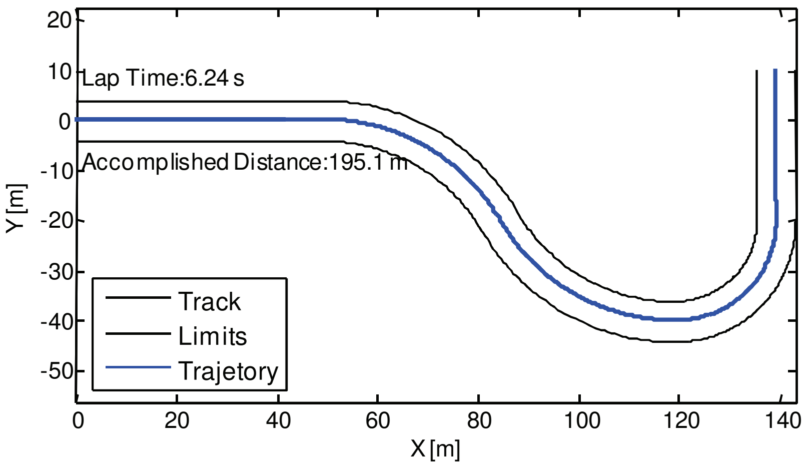

Figure 6.13 – Intelligent Applications Test: Center Line Trajectory.

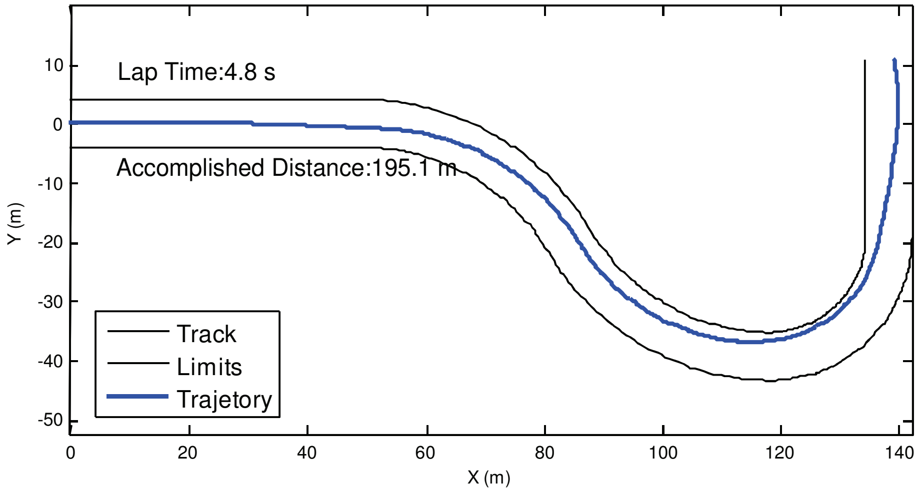

Figure 6.14 – Intelligent Applications Test: Evolved Trajectory.

Figure 6.14 shows the evolved trajectory that optimizes the lap time. The tangent profile appears on the transition between the second and third curves and, in order to maintain high speed, the final curve is performed with larger radius.

Finally, this trajectory is the reference signal for the Fuzzy Driver. The result is shown in Figure 6.15.

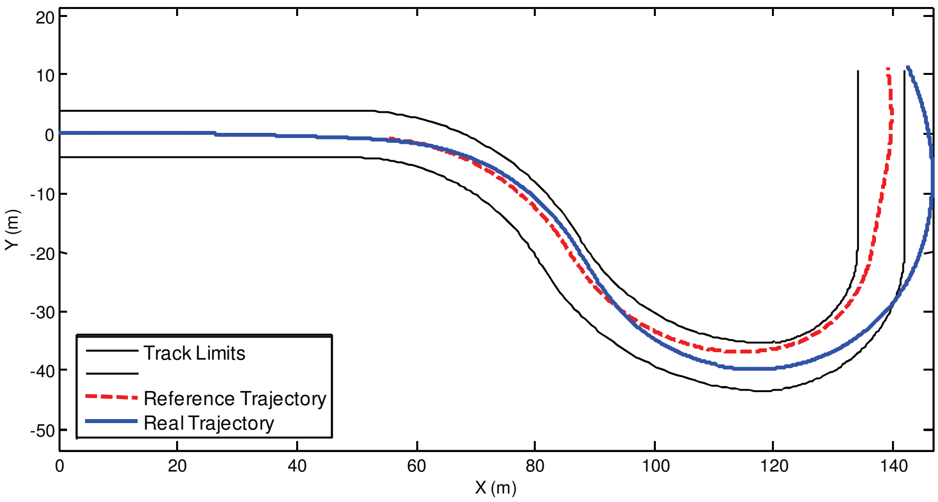

Figure 6.15 – Fuzzy Driver High Speed Test: Trajectory.

This controller only acts on lateral kinematics and the vehicle model used does not consider variation in speed. Therefore, the presented deviations from the reference trajectory were expected. The evolved trajectory demands variation in the longitudinal velocity, as shown in the acceleration profiles on Figure 6.16.

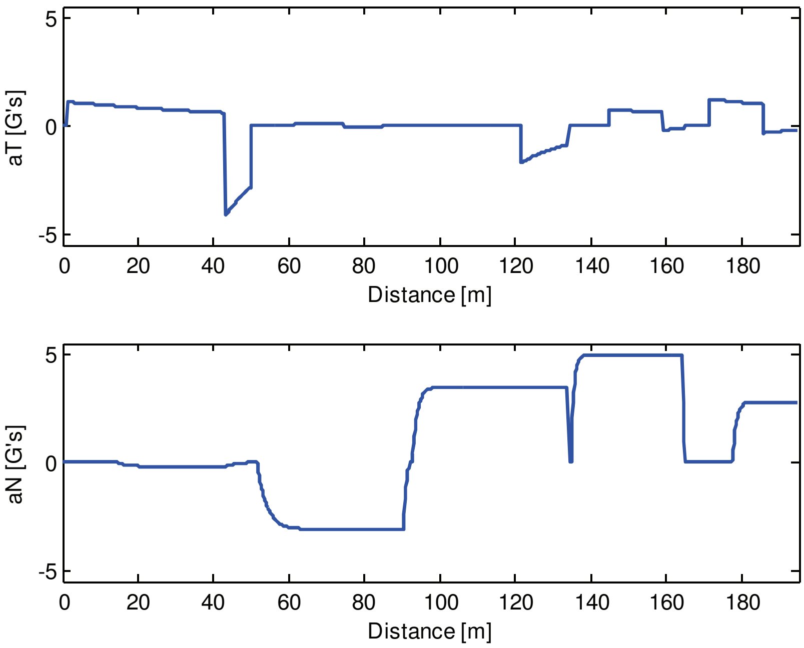

Figure 6.16 – Intelligent Applications Test: Evolved Acceleration Profiles.

Another fuzzy controller developed by Hey[12] that acts on longitudinal accelerations, inferring when to throttle, brake and change gear, shifts the vehicle in order to control the vehicle's speed profile. The next step would be to link both fuzzy controllers, creating some rules to correlate positions errors and longitudinal velocity, enabling a more accurate trajectory control.