**Abstract.**

# 1 Introduction

In this work , we will try to design fuzzy based controllers to drive race cars in TORCS simulator:
AD1: Speed based fuzzy controller using the track sensors. AD2: Speed and steer based fuzzy controller using the track sensors.
AD3: Speed and steer based fuzzy controller using the turning radius.
AD5: Speed and steer based fuzzy controller using the track sensors and consideration of opponents position in the track.

# 2 The Open Racing Car Simulator "TORCS"

## 2.1 TORCS Presentation

The Open Racing Car Simulator) is a free three-dimensional racing video game. Even if it has not the graphic quality of commercial games, TORCS allows to play different races, about forty vehicles , on a wide variety of tracks (dusty roads, highways, formula 1 circuits, etc).[[**?**]]. The TORCS project was created by *Eric Espi* and *Christophe Guionneau* and currently *Bernhard Wymann*, *Christos Dimitrakakis* and other contributors continue to develop it. [[**?**]]

TORCS1 (The Open Racing Car Simulator) is one of the most popular car racing simulators. It is written in C++ and is available under GPL license front its web page. TORCS presents several advantages for academic purposes, such as:

1. It lies between an advanced simulator, like recent commercial car racing games, and a fully customizable environment, like the ones typically used by computational intelligence researchers for benchmark purposes.
2. It features a sophisticated physics engine (aerodynamics, fuel consumption, traction,...) as well as a 3D graphics engine for the visualization of the races.
3. It was not conceived as a free alternative to commercial racing games, but it was specifically devised to make it as easy as possible to develop your own controller In fact, controllers are implemented as separated software modules ,so it is easy to develop a new controller and to plug it into the game.

TORCS has several different cars with real characteristics(fuel consumption, aerodynamics, collisions or traction) (see Fig.34) and different tracks [[**?**] ]( See fig. 3 and Table1). The user can create his own track with the track editor.

**Fig. 1.** TORCS race

**Fig. 2.** TORCS race car

**Table 1.** TORCS Tracks

| Num | Name | Author | Length(m) | Width (m) |
|---|---|---|---|---|
| 1 | CG Speedway nem-ber1 | B.Wymana | 2057 | 15 |
| 2 | CG Track 2/1 | CG | 3185/2843 | 15/10 |
| 3 | Olethos Road1 | C.Dimitrakakis | 6282 | 10 |
| 4 | RuudsKongen | A.Summer | 3274 | 11 |
| 5 | Spring | E.Espie | 22129 | 12 |
| 6 | Steer1 | A.Summer | 38423 | 14 |
| 7 | Whell1 | E.Espie B.wymann | 4328 | 14 |
| 8 | Whell2 | A.Summer | 6205 | 12 |
| 9 | Alborg | E.Espie B.wymann | 2587 | 10 |
| 10 | Apline 1 | E.Espie | 6355 | 12 |
| 11 | Apline 2 | D.schellhammer | 3773 | 10 |
| 12 | Brondehach | E.Summer | 3919 | 13 |
| 13 | Crokscrew | Kilo,andrew | 3608 | 12 |
| 14 | Etrack [1-6] | E.Espie B.wymann | [3000-7000] | 12/13 |
| 15 | ERoad | E.Espie B.wymann | 3270 | 16 |
| 16 | Forza | A.Summer | 5784 | 11 |

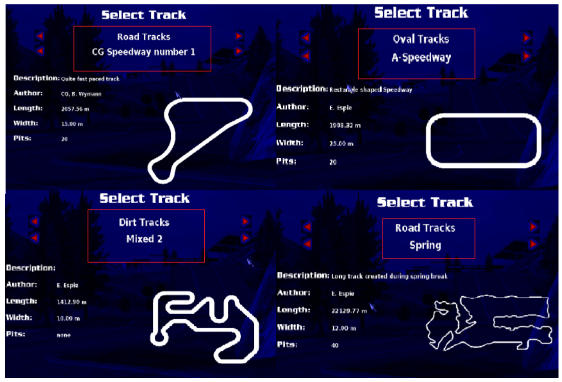


**Fig. 3.** TORCS's tracks

The drivers in TORCS can practice and test their controllers and can participate in three main kinds of races:

– *Warm-up*: The driver is able to explore the track and extract the information it contains for a limited time;
– *Qualifiers*: Time trial race where drivers try to get the fastest time.
– *Real race*:
  The best eight drivers in the qualifiers can participate in a a final race to get the winner.

**TORCS controller definition** A robot is a program run from TORCS that drives a car. It gets as input information about the current state of the car and its situation on the track. These collected data are used to compute actions to do in the next simulation tick; like steer, gear changes, acceleration or brake and clunch. A client may request a restart of the race by sending a special action on the server: Resatart or shutdown. [[**?**]]

### 2.2 TROCS Architecture

The Open Racing Car Simulator (TORCS) is a standalone application where robots are built as separate modules loaded into the main memory when a race takes place. [[**?**] ] This simulator includes the original architecture of TORCS in three ways: [[**?**]]

1. TORCS is a client-server applications, robots (bots) are run in external processes connected to the server running across by the UDP connections.

2. The simulation is achieved in real time and game ticks are approximately 20 ms of simulated time, the server sends the current sensors values to each robot and waits 10 ms (real time) to receive an action from the bot. If no action is happened, the simulation continues and the last action will be used.

3. The "TORCS" software make a physical separation between the driver code and running server, it builds an abstract layer that gives complete freedom of choice of programming language that will be used for robots and limit access only to information defined by the designer (the data encapsulation) See Figure 4.

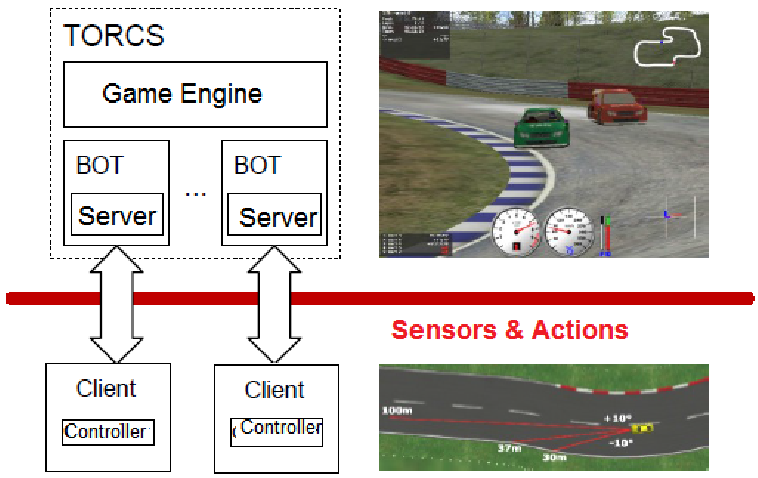


**Fig. 4.** TORCS Architecture

### 2.3 Sensors and Actuators

The TORCS software creates a physical separation between the game engine and pilotes.Ainsi to develop a robot it is not necessary to have any knowledge a about the engine TORCS or internal data structure.[[**?**]]

In the competition, The robot perceives the racing environment through a number of sensors that provide information about the state of the car (the current speed, fuel level, ...), the state of the race (current lap, the elapsed distance,...), opponents positions and track borders, ... these data enable the designer to form a base of solutions to achieve a typical driving. [[**?**]] [[**?**]]
Tables 2 present a complete list of available sensors with a description of each one.[[**?**]]

| Sensor | Name | Range (unit) | Data type |
|---|---|---|---|
| 1 | Angle | [$-\pi$,$+\pi$ ] | Double |
| 2 | curLapTime | [0,$+\infty$) (s) | Double |
| 3 | damage | [0,$+\infty$)(point) | Double |
| 4 | distFromStart | [0,$+\infty$) (m) | Double |
| 5 | distRaced | [0,$+\infty$) (m) | Double |
| 6 | focus | [0,200] (m) | Double |
| 7 | fuel | [0,$+\infty$) (l) | Double |
| 8 | gear | {-1,0,1,.. 6}g | Integer |
| 9 | lastLapTime | [0,+1] (s) | Double |
| 10 | opponents | [0,200] (m) | Double |
| 10 | racePos | {1,2,...,N} | Double |
| 11 | rpm | [0,$+\infty$) (rpm) | Double |
| 13 | speedX | ($-\infty$,$+\infty$) (km/h) | Double |
| 14 | speedY | ($-\infty$,$+\infty$) (km/h) | Double |
| 15 | speedZ | ($-\infty$,$+\infty$) (km/h) | Double |
| 16 | track | [0,200 ] | Double |
| 17 | trackPos | ($-\infty$,$+\infty$) | Double |
| 18 | wheelSpinVel | [0,$+\infty$) (rad/s) | Double |
| 19 | z | ($-\infty$,$+\infty$) (m) | Double |

**Table 2.** Available TORCS sensors Description

**Sensors Description:**

1. **Angle:**Angle between the car direction and the direction of the track axis. see(fig 6 and fig 5 )

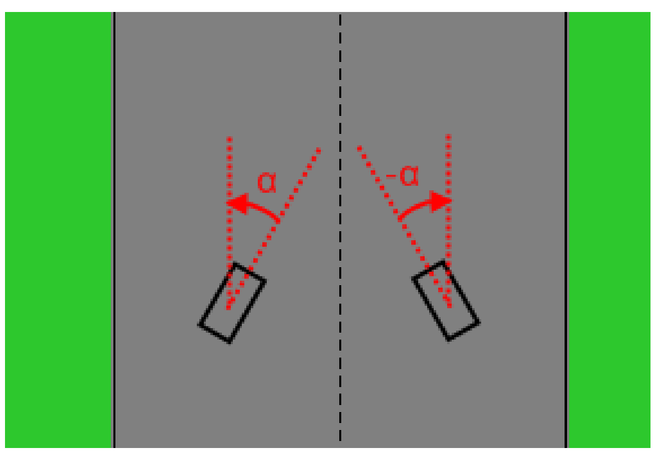


**Fig. 5.** Angle between Car and track axis.

2. **curLapTime:** Time elapsed during current lap.

3. **damage:** Current damage of the car (the higher is the value the higher is the damage).
4. **distFromStart:** Distance of the car from the start line along the track line.
5. **distRaced:** Distance covered by the car from the beginning of the race.
6. **focus:** Denote the normalized deviation and the lateral deviation of the car with the track axis. -focus = 0, when the car is in the middle of the track.
   -focus ¡0 when the car is deviated to the left.
   -focus ¡-1 or focus ¿ 1 when the car is out of the track.(See fig 5)

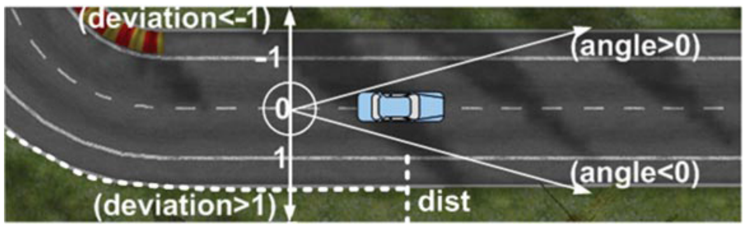


**Fig. 6.** TORCS angle sensor.

7. **fuel:** Current fuel level.
8. **gear:** Current gear: -1 is reverse, 0 is neutral and the gear from 1 to 6 .
9. **lastLapTime:** Time to complete the last lap
10. **opponents:** Vector of 36 opponent sensors: each sensor covers a span of 10 degrees within a range of 200 meters and returns the distance of the closest opponent in the covered area. The 36 sensors cover all the space around the car, spanning clockwise from -180 degrees up to +180 degrees with respect to the car axis.(see fig.10)

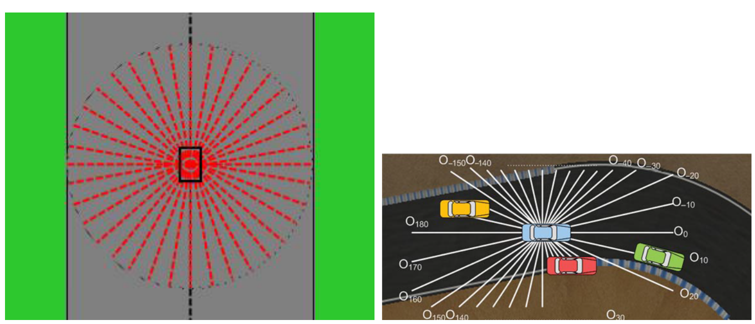

**Fig. 7.** Opponents sensors.

11. **racePos:** Position in the race with respect to other cars.(see fig.11)
12. **rpm:** Number of rotation per minute of the car engine.

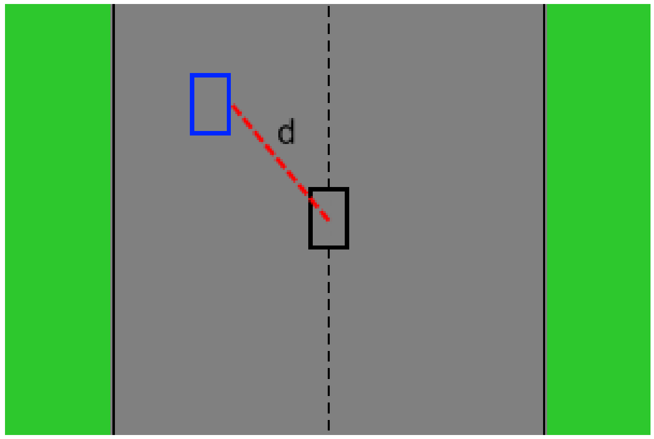


**Fig. 8.** Trackpos sensor value.

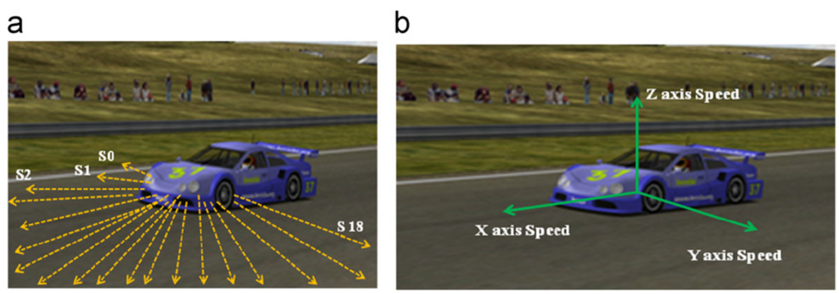


**Fig. 9.** (a) Position sensor (b) Speed sensors.

13. **speedX:** (km/h) Speed of the car along the longitudinal axis of the car.
14. **speedY:** (km/h) Speed of the car along the transverse axis of the car.
15. **speedZ:** (km/h) Speed of the car along the Z axis of the car (See fig.9).
16. **track:**
    Vector of 19 sensors each returns the distance between the edge of the track and the car within 100 meters. By default, the sensors sample the space in front of the car every 10 degrees, spanning clockwise from -90 degrees to +90 degrees to the axis of the car. However, the configuration of the track sensors can be set by the client once during initialization, before the start of each race. When the car is outside the track (less than -1 or greater than 1), the returned values are not reliable (usually -1 is returned).
    ( See fig 10 and fig.9)

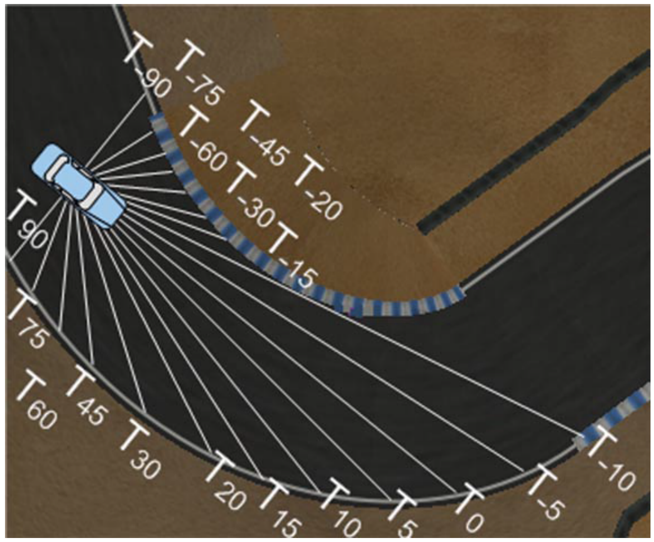


**Fig. 10.** Track borders sensors.

17. **trackPos:** Distance between the car and the axis of the track. The value is normalized to the width of the track: it is 0 when the car is on the axis, -1 when the car is on the right edge of the track and 1 when it's on the left. Values greater than 1 or less than -1 means that the car is outside the track[[**?**]](Voir fig11)

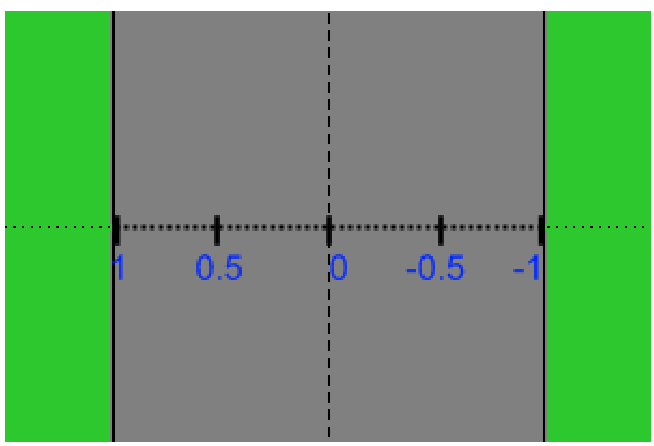


**Fig. 11.** Car position in the race.

18. **wheelSpinVel:** Vector of 4 sensors representing the rotation speed of wheels.

19. **Z:** Distance of the car mass center from the surface of the track along the Z axis.

**Actuators** The driver bot is controlled in the game TORCS through a typical set of actuators: the steering wheel "Steer", the accelerator "accel" the brake pedal and the gearbox. In addition, a meta-action is available to request a restart of the race to the server. Table 3 details the available actions and their representation. [[**?**] ]

| **Action** | **Range** (unit) | **Data type** |
|---|---|---|
| Acceleration | [0,+1] | Double |
| Brake | [0,+1] | Double |
| Gear | -1..0..+6 | Double |
| Steer | [-1,+1] | Double |
| Clunch | [-1,+1] | Double |

**Table 3.** TORCS Effectors

## 3 TORCS Controller Architecture

The basic architecture of a controller consists of 5 simple modules (fig.12). This modular architecture is a key factor to achieve good results. The basics modules functions are explained below [[**?**]]:

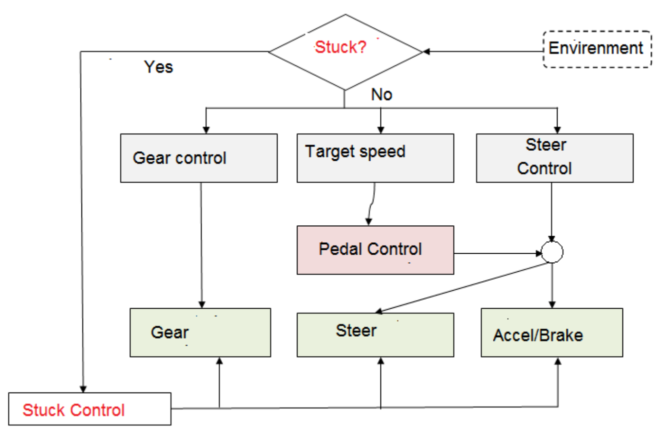


**Fig. 12.** Controller's Architecture.

1. The gear control is mainly responsible for switching between first and sixth gear. Its functionality is completed with the task of detecting stuck situations to apply the reverse.
2. The target speed unit affects the speed limit for a track segment.
3. speed control uses the results of others modules to update or maintain certain speed by managing the throttle and the brake pedal. In addition, two techniques are implemented to prevent the car from slipping. A traction control (TCL) and an anti-lock anti-brake system (ABS) that will reduce, if necessary, actions over the accelerator and brake, respectively.
4. Steer control module manages the car's direction by acting on the vehicle wheel.[[**?**]]
5. The learning module detects segments of the track where the target speed can be increased or decreased with information on the previous laps, straight segments or segments where the car is off the track [[**?**]].

## 4 Simple driver

TORCS comes with a simple driver often used to validate new designed controllers [[**?**]][[**?**]].
This project was provided by the TORCS software, and it was developed by *Daniele Loiacono* in 2007. It presents very basic functions for controlling the race car to give developers an idea of what the controller should look like. It contains simple functions to control the speed, steering angle and speed without dealing with opponents. Below are the driving principles in this system [[**?**]]:

1. If the car is at an angle that exceeds 30 degrees with the axis of the track for at least 25 consecutive ticks, then it is considered in stuck.In this case, the simple driver sets the car in reverse with an angle which is the negative of the current angle. It drives the car that way at low speed until the front of the car is facing the border of the track. For example, if the car is on the left side, then it should be turned to the left. At that time, the car moves into first gear and the steering angle is reversed, so that the car can start moving forward again[[**?**]].

2. If the car is not in stuck, the simple driver proceeds as follows [[**?**]]:

   – First, it calculates the target speed. To do this, it gets the forward distance along the axis of the car by the front sensor FS and the sensor at 5 degrees to the left LS and 5 degrees to the right RS, as shown in Figure 13. Suppose RS¿ FS. Then, the driver made a first estimation of the "Steering angle" as the angle between the tangent to the road so that the car is facing the direction of car to the left else the angle is taken to the right.

– Then, the target speed is calculated as follows

$$targetSpeed = \frac{(maxSpeed * FS * \sin(turnAngle))}{maxSpeedDist} \tag{1}$$

where: maxSpeed and maxSpeedDist are constant of the car.

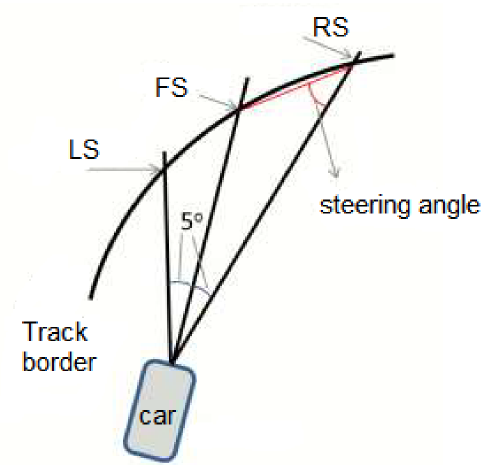


**Fig. 13.** Computing steering angle in simple driver.

## 5 Fuzzy based controllers design (AD1)

The proposed controller "AD" has the same modular architecture as the simple driver where the target speed and speed values are computed via fuzzy controller using five 5 position sensors.

### 5.1 Fuzzy target speed

To estimate the target speed based on fuzzy rules, two cases are considered.

1. If the car is in straight line, the target speed will take a maximum value (maxSpeed).

2. If it is near a curve, the controller will decrease its current speed to a value included in the interval [minSpeed; maxSpeed] km / h for example if the turn is a strict turn the target speed will have a minimum value and a wide turn, it will have an average value.

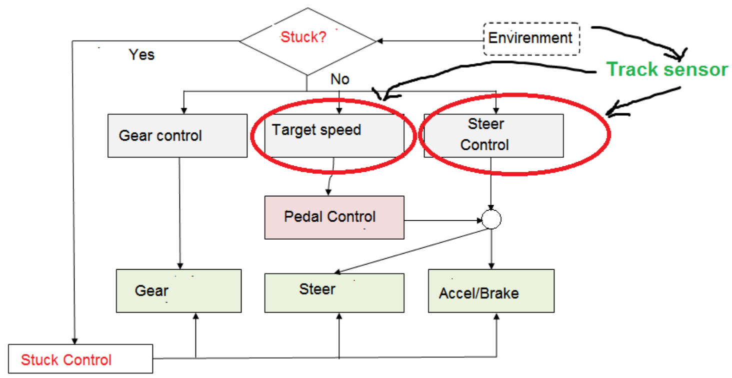


**Fig. 14.** Target speed fuzzy control Architecture.

In case the car is off the track or near a curve, the brake system is activated, ie the ABS and TCS will be loaded to avoid the car skidding.
The obtained target speed will be used in calculating the value of acceleration (see fig.**??** et fig.**??**).

$$Gas(speed - Target_{speed}) = -1 + \frac{2}{1 + e^{speed - Target_{speed}}} \qquad (2)$$

Gas is for acceleration, Speed is the current speed of the car.
Our fuzzy controller has 3 input values and one output: the target speed value(See figure 15):

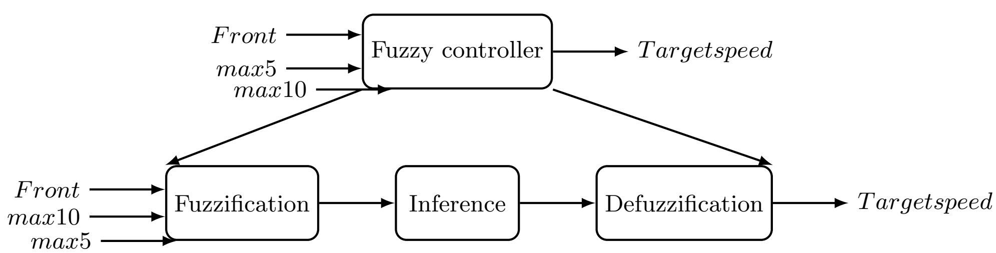


**Fig. 15.** Fuzzy control of target speed.

The "AD" controller is a Mamdani based fuzzy system (See fig.15) with trapezoidal membership functions for input variables. It uses three values among the 19 of the track sensor (See fig.16):

1. Front = Track[9] with angle = 0 , the front distance between the car and the border of track.
2. max5 = max (Track[8]; Track[10]) the max distance with +5 and -5 angle.
3. max10 = max (track[11]; track[7]) , the max distance with +10 and -10 angle.

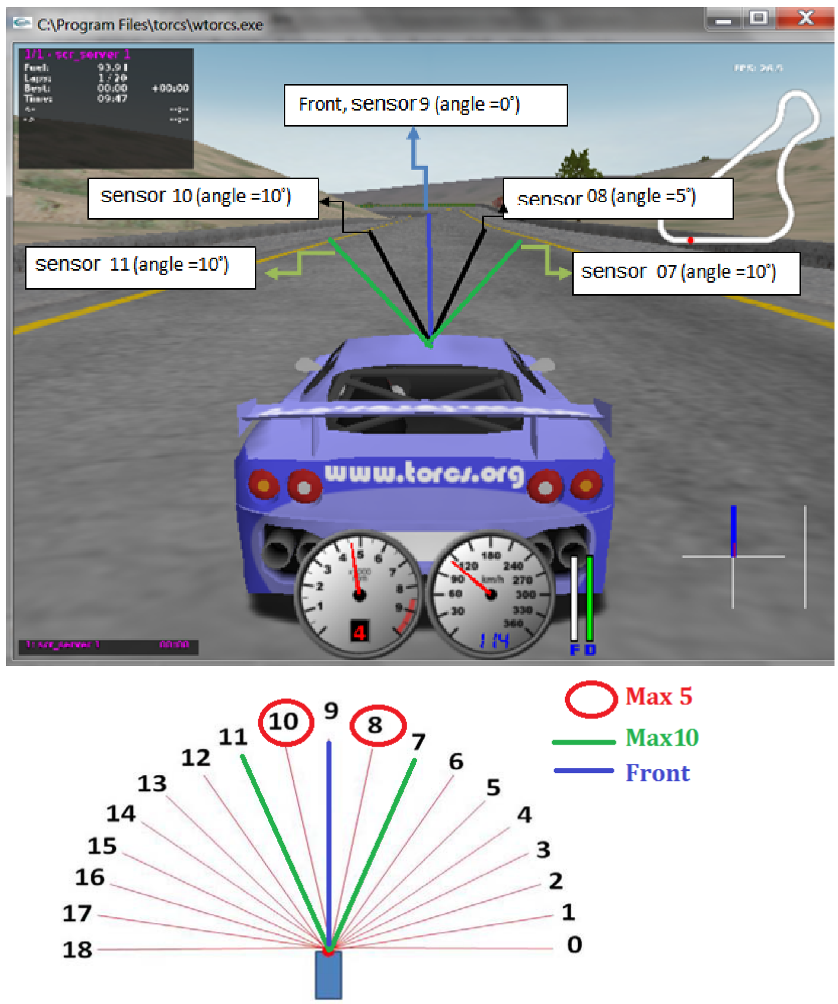


**Fig. 16.** Fuzzy inputs.

1. **Fuzzification**
   Each input variable is represented by three membership functions Low, Medium and High as shown in the figures (fig. **??** , Fig. 18 , fig, 19 , Fig. 20) The description of fuzzy inputs and output are represented in table 1 .

**Table 4.** Fuzzy variables description

| Variable | Range | Name | MF | Low | Medium | High |
|---|---|---|---|---|---|---|
| Input | [0-100] m | Front | trapezoidal | [0-50] | [20-80] | [80-100] |
| Input | [0-100] m | max 5 | trapezoidal | [0-40] | [10-70] | [40-100] |
| Input | [0-100] m | max 10 | trapezoidal | [0-30] | [0-60] | [30-100] |
| Output | [0-200]m/s | TS | singleton | / | / | / |

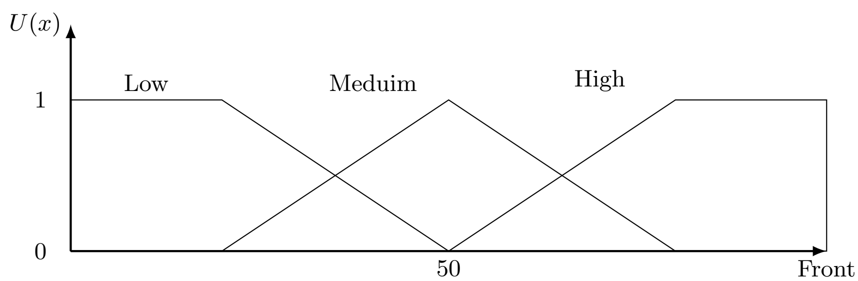


**Fig. 17.** Membership functions of "Front".

2. **Fuzzy inference system** The rules base is designed on the principle that if the front distance is maximal, then target speed should be maximal, and this value should be less when the frontal distance is lower. The fuzzy rules are listed below:

   (a) If Front is High Then TargetSpeed is TS1
   (b) If Front is Medium Then TargetSpeed is TS2
   (c) If Front is Low and Max5 is High Then TargetSpeed is TS3
   (d) If Front is Low and Max5 is Medium Then TargetSpeed is TS4
   (e) If Front is Low and Max5 is Low and Max10 is High Then TargetSpeed is TS5
   (f) If Front is Low and Max5 is Low and Max10 is Medium Then TargetSpeed is TS6

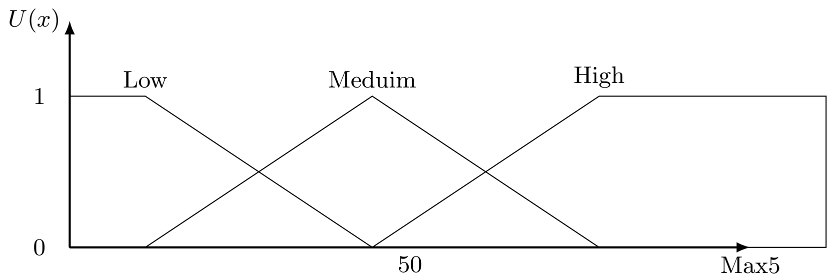


**Fig. 18.** Membership functions of "Max5".

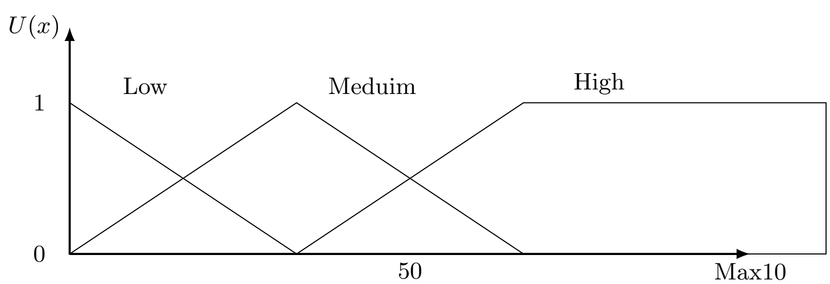


**Fig. 19.** Membership functions of "Max10".

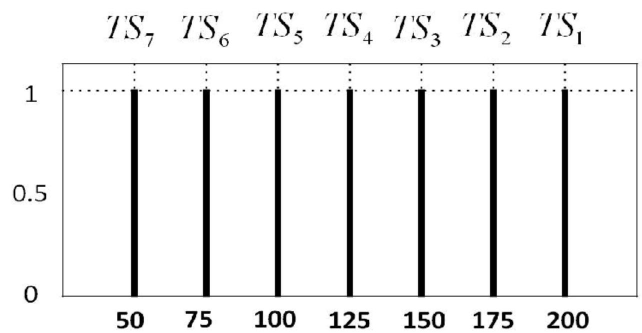


**Fig. 20.** Target speed values.

(g) If Front is Low and Max5 is Low and Max10 is Low Then TargetSpeed is TS7
In addition, a crisp rule is added to rule base to obtain a maximum value of a target speed when the value of the three input variables as far as possible, less than 100 m:

(h) If Front = Maxdistspeed or Max5 = Maxdistspeed or Max10 = Maxdistspeed Then TargetSpeed = Maxspeed

3. **Deffuzzification** The output value (target speed) is encoded by seven singletons. This phase is to transform the fuzzy set of output in a real value for the controlled system .

### 5.2 Fuzzy Steer

We can use another fuzzy controller in the control of the Steer to estimate and determine the target position of the car:
if the car is straight line then it will take as target position half width of the race track.

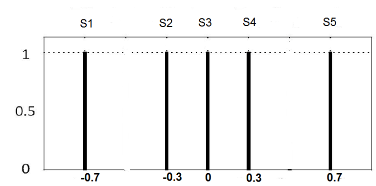


**Fig. 21.** Steer values.

If the car is near a right curve,it will approach the path leading to the right, with a space between the car and the border of the track to avoid the loss of control and shocks .The same if the car is near a left curve (see fig. 22 )

Detection of curves is based on the sensor values (max10, max5, front).

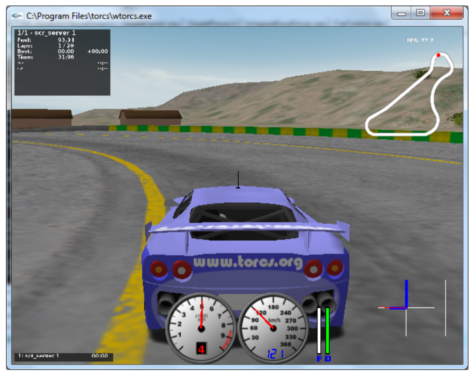


**Fig. 22.** 'Steer control' of a car in a curve.

the rule base is presented as follows:

- If Front is High Then steer is S3
- If Front is Medium Then steer is TS2
- If Front is Low and Max5 is High Then steer is TS3
- If Front is Low and Max5 is Medium Then steer is TS4
- If Front is Low and Max5 is Low and Max10 is High Then steer is TS5
- If Front is Low and Max5 is Low and Max10 is Medium Then steer is TS6
- If Front is Low and Max5 is Low and Max10 is Low Then steer is TS7

# 6 Turning Radius based fuzzy control (AD3)

In this section, we focus on controlling the direction of the car "Steer control" and control of the target speed using the turning radius.

## 6.1 Car Turning geometry

There are many techniques used to turn a car, but the most effective one is to study the geometry of the driving line of the car and the geometry of the track such as detecting track borders ...

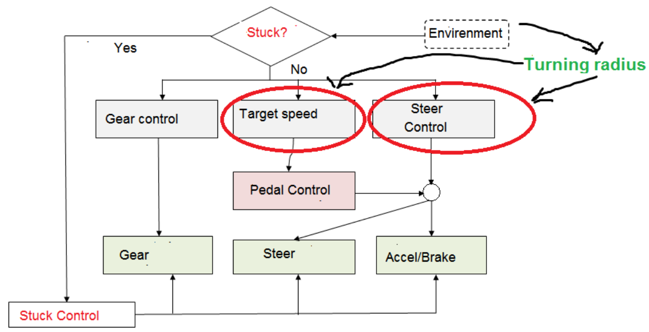


**Fig. 23.** Turning radius fuzzy control architecture.

Three cases are considered in control of the wheel (see equation (3) and (4))

$$Steer = angle - 0.5 * \frac{position}{Steer_lock} \tag{3}$$

$$Steer = \frac{angle}{Steer_lock} \tag{4}$$

*SteerLock* is a constant value from the description available in TORCS of the car and its value is 0.785

1. The car is inside the track, it takes a restrained line

2. The car is outside of the track, the system initiates the stuck control function.

3. The car is near a curve to the right or left, we study the turning geometry.

**turning geometry** The turning radius is the minimum radius of the circle described by a car during a turn. His goal in our case is to have the target position when the car is a curve. (See fig. 24)

Figure 25 represents the path of the turn (right / left), it is cut into 4 parts-such as:

1. The entrance area (Blue):
   its goal is to put the car near the centre line and to determine a braking point when the sensors show that the turn is not far away. (See 26)
2. discovery area (Green).
   Here, while keeping the car close to the center line and starting to register curve, will, look, go for a sign which materializes the end of the turn.
   The acceleration must remain constant and the same as from the entrance. (See Fig 26)

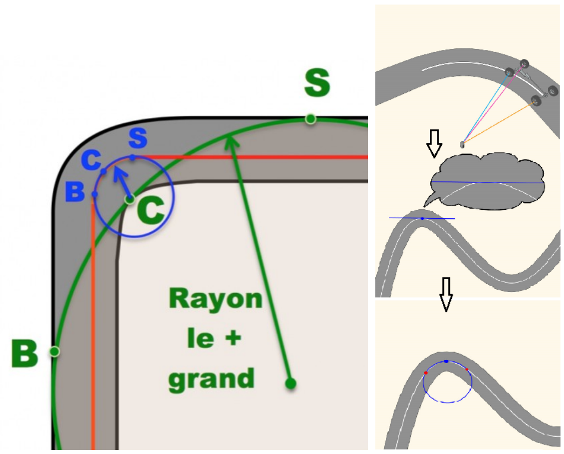


**Fig. 24.** Turning radius.

3. Solicitation area (red).
   Here, the driver starts to determine the inside turning radius. (See Fig 26)
4. Stability recovering area (yellow)
   The car is gets outside the curve. In fact, in this area, the car straightens and we leave to the next turn.

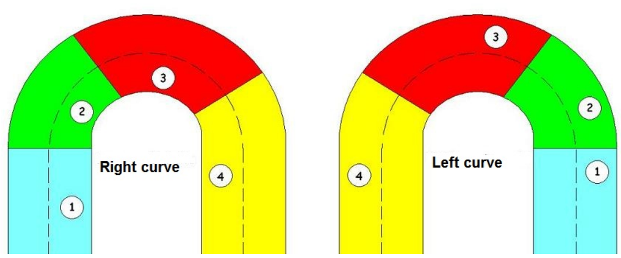


**Fig. 25.** Right and left curves.

The curve used is a circle see Figure 26, in order to have a single look point (the target) possible. The coordinates are given relative to the car centre.

The Y axis: the axis is formed by the straight line passing through the middle of the car in the direction of movement
. The X axis passes through the middle of the vehicle perpendicular to the direction of movement to the right
.

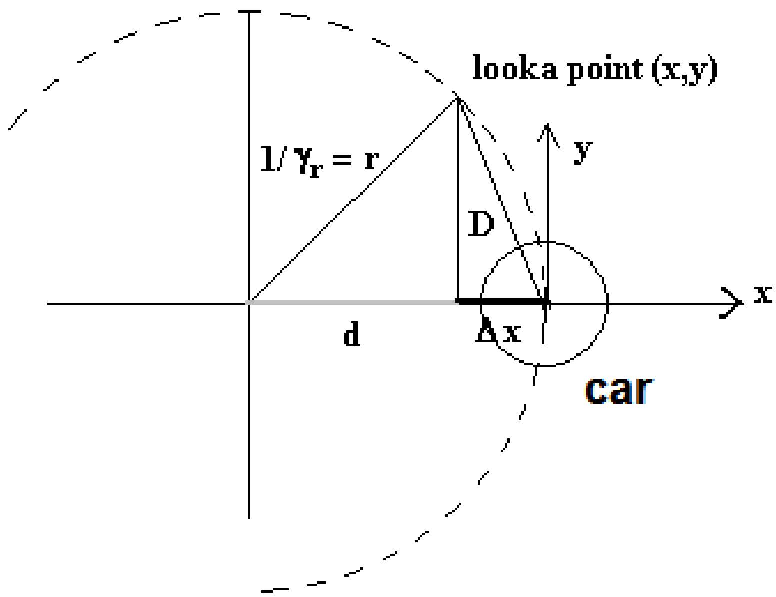


**Fig. 26.** Turning radius.

(xr, yr) is the current position of the vehicle, and (xg, yg) is the target to be converted the vehicle mark.

$$xgv = (xg - xr)\cos(\theta) + (yg - yr)\sin(\theta) \tag{5}$$
$$ygv = -(xg - xr)\sin(\theta) + (yg - yr)\cos(\theta). \tag{6}$$

Where (xgv, ygv) are the coordinates of the target in the vehicle mark and $\theta$ is the current direction of the car.
D is the distance between the centre of the vehicle and the target. $1/\gamma$ is the radius of the circle that pass through the centre of the car and the target. The curvature of the vehicle's trajectory path is calculated as follows:

$$\gamma r = 2\delta x/D^2 \tag{7}$$

This equation is obtained using the following equations:

$$x^2 + y^2 = D^2 \tag{8}$$
$$x + d = r \tag{9}$$

(x,y) are the target coordinates. We have:

$$d = r - x \tag{10}$$
$$(r - x)^2 + y^2 = r^2 \tag{11}$$
$$r^2 - 2rx + x^2 + y^2 = r^2 \tag{12}$$
$$2rx = D^2 \tag{13}$$
$$r = D^2/2x \tag{14}$$
$$\gamma r = 2x/D^2 \tag{15}$$
$$\tag{16}$$

### 6.2 Turning radius based Fuzzy controller design

Now it is easy to set up the architecture of our fuzzy controller to determine the target position of the car (see Figure 27). We will have two fuzzy controllers with the same input, one is for the target position and the other is for the target speed.

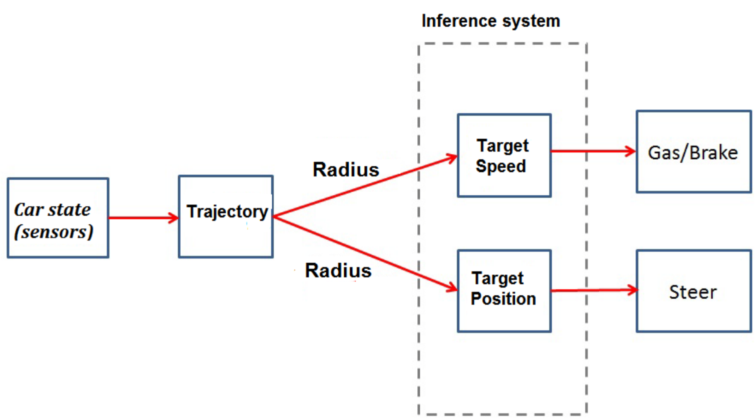


**Fig. 27.** Fuzzy controller Structure.

**Fuzzyfication** In our case we have real input for the two fuzzy controllers which is the turning radius R and as output "The target position TP for the first one and the target speed TS for the other one .(Table 6.2) (see Figure 28 29)

**Table 5.** Fuzzy controllers description

| | Radius | TP | TS |
|---|---|---|---|
| Type | Input | Output first FC | Output of second FC |
| Ranges | [0-250] m | [0,1]m | [0-120]km/h |
| fuzzy variable floue | Radius | TP | TS |
| Membership functions | trapezoidal | singleton | singleton |
| SEF "Low" | [0-125] | —— | —— |
| SEF "Medium" | [50-200] | —— | —— |
| SEF "High" | [125-250] | —— | —— |

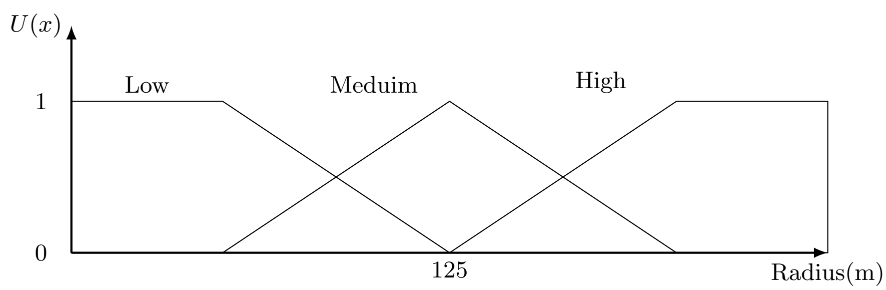


**Fig. 28.** Membership function of the input: radius.

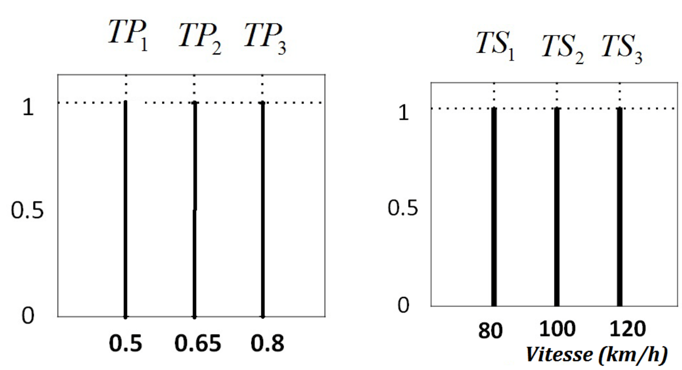


**Fig. 29.** Membership function of the outputs: TP and TS.

**Fuzzy inference** The rules basis is designed on the principle that if the turning radius is equal to a very minimum then not turn so we will adjust the position of the car to the middle of the track,

otherwise if the turning radius is equal to a maximum value, so the position of the car will change its direction to the rotation path (right or left). (See Fig. 25) And the same thing if the turning radius is equal to an average value, so we will approximate the rotation path (see Fig. 25)
This concept can also be applied to the estimated target speed

1. If *Radius* is *High* then $Target_{speed}$ is $TS_3$
2. If *Radius* is *Medium* then $Target_{speed}$ is $TS_2$
3. If *Radius* is *Low* then $Target_{speed}$ is $TS_1$
4. If *Radius* is *High* then $Target_{position}$ is $TP_1$
5. If *Radius* is *Midium* then $Target_{position}$ is $TP_2$
6. If *Radius* is *Low* then $Target_{position}$ is $TP_3$

**Fig. 30.** Rule base of radius based fuzzy controller .

### 6.3 AD5 fuzzy controller design

The AD5 controller is a speed-steer based fuzzy controller with opponents consideration.

Using the previous approaches, we will adapt the driving behavior when opponents are nearby. It will modify or replace the outputs obtained from the steering, throttle and brake controllers.

The modification of throttle control outputs, brake and steering will be mainly using information from opponents sensors .
Three types of updates are performed:

1. on the steering output to avoid opponents (on hold)
2. on the steer, but with a strong movement in the collision risk (collision avoidance)
3. the brake value when there is an adversary front of the car in a dangerous distance.
   Both actions on the steer are performed simultaneously.

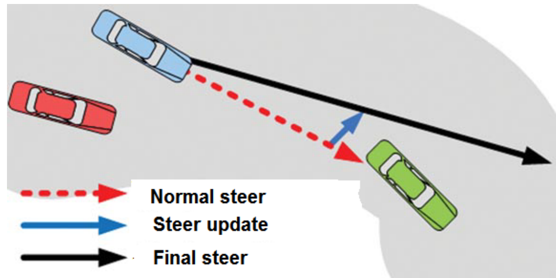


**Fig. 31.** Car behaviour when opponent is nearby

# 7 Simulation results

In this section we present all the experiments that we performed in order to measure the performance of our controller "AD" (Algerian Driver) after developing many functions and adding many features. We first describe the methodology we have used and next, we present the experimental results of the implementation of the "AD", with opponents and using special criteria in each case.

We have established a set of tests to measure the performance of the controller "AD". We compared the newly developed Controller with existing methods: the single driver. We will run every single control car by choosing the car called "SRC-server1" in several race tracks.

## 7.1 Simulation settings

**Tracks settings** TORCS provides different tracks to choose from. These race tracks are designed by different developers in order to test the performance of the controllers on different difficulty circuits and on different types of roads.

The TORCS training environment provides three main tracks: Road tracks, dirt tracks and oval tracks. There are 21 tracks which are circuits of a substantial length with many curves of different difficulty. In addition, the system offers 8 dirt tracks, also very long and represents a growing challenge in terms of control of the car. There are 9 oval tracks, which are shorter and more predictable, and designed primarily to optimize speed.

During competitions, drivers can expect to be exposed to unfamiliar roads to challenge their ability to win the race with minimum damages in record time.

In our case, we chose three of road tracks, a dirt road, and an oval track.

On oval tracks, E-Track5 seems to be the most interesting, as it has curves in both directions. On dirt roads, Dirt1 has a good variety of curves. For road tracks, we selected three tracks: Forza, E-Road and CG-Speedway Number1. Table 6 presents the properties and description of each selected track.

**Table 6.** Description of Selected tracks

| **Track name** | E-Track5 | Dir1 4 | Forza | E-Road | CG-Speedway Number1 |
|---|---|---|---|---|---|
| **Shape** | fig.32 (4) | fig.32 (2) | fig.32 (3) | fig.32 (2) | fig.32 (1) |
| **Track Type** | Oval | Dirty | Road | Road | Road |
| **Description** | simple track race | Track for outdoor race | Very fast and smooth | Track race | rhythmic track and fast enough |
| **Length** | 1621.73 m | 3260.43 m | 5784.10 m | 3260.43 m | 2057.56 m |
| **Width** | 20.0 m | 16.0 m | 11.0 m | 16.0 m | 15.0 m |


**Fig. 32.** Tracks shape.

As Figure 33 shows, the E-Road track is a road track with many curves of all kinds: fast, medium and slow curves. It allows us to test the performance more effectively.


**Fig. 33.** A car in E-Road track.

**Cars settings** SCR 1 server is a NASCAR car and member of the SCR Server team (A team can used different types of car as car1-stock1, car1-tbr1 ...).
A Sprint Cup Series NASCAR-type car weighs at least 1542 kg for 850 horses with . The engine is a V8 with $5,866cm^3$. Its chassis is tubular type. It has a Borg-Warner gearbox with four-speed for road circuits.

In our simulations, we used the car "car1-tbr1" see Fig.34

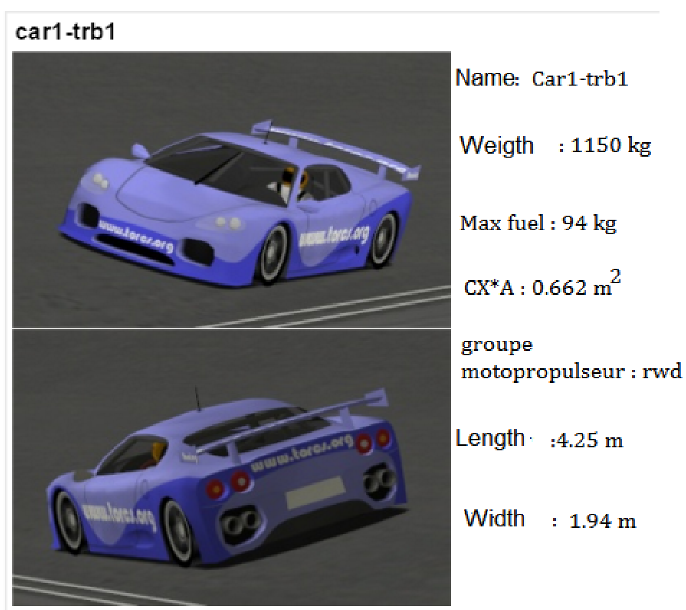


**Fig. 34.** car1-trb1 features .

**Controllers settings** To validate the fuzzy controllers, we have considered the four controllers based on the the driving approaches in the sections above.
Our controllers are also compared to the simple driver SD(See Table 7 ).

**Table 7.** Used controllers

| **Controller** | **used approach** |
|---|---|
| SD | / |
| AD 1 | Fuzzy Target speed |
| AD 2 | 1. Fuzzy Steer<br>2. Fuzzy Target speed<br><br>with track borders sensors |
| AD 3 | 1. Fuzzy Steer<br>2. Fuzzy Speed<br><br>using turning radius |
| AD 5 | 1. Fuzzy Steer<br>2. Fuzzy Target speed<br>3. Opponents consideration Target speed<br><br>with track and opponents' sensors |

Table 7 represents the features of the different controllers that we have developed and we tested in TROCS in the case of a simple driving without opponent.

We conducted a series of experiments to measure the performance of the best version control in some race tracks (05 tracks in table 6 ).

### 7.2 Results of alone race

**One lap race:** In this case we set the distance to a value of just one lap, to test the performance of controllers for each circuit such as:

Table 8 we present the results (time, min speed, max speed, the damage rate and fuel) of 6 different controllers for a lap on the 5 tracks. We have noticed that the best results are given by the "AD5" controller in 4 tracks (E-Track 5, Froza, E-Road, CG-Speedway Number1) (see fig. 35) and only in the dirty track that the controller AD 1 and AD 3 give us good results (a minimum time without damage). Figure 35 presents the ranking of the controller based on the time criterion (Best time).

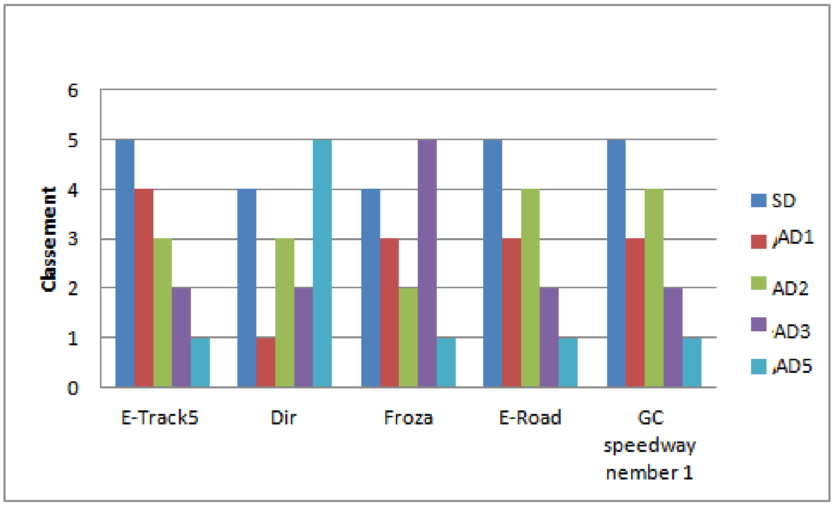


**Fig. 35.** Controllers ranking in one lap.

**Table 8.** Controllers results in one lap

| | SD | AD1 | AD2 | AD3 | AD5 |
|---|---|---|---|---|---|
| **E-Track5** | | | | | |
| | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:11:58 | 01:12:69 | 01:08:36 | 01:05:96 | 37:12 |
| Topspeed | 149 | 146 | 164 | 168 | 199 |
| Minspeed | 59 | 59 | 67 | 61 | 116 |
| Damage 0 | 0 | 0 | 0 | 0 | 0 |
| Fuel | 93:51 | 93:91 | 93:91 | 93:92 | 93:33 |
| **Dir1** | | | | | |
| | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Lap Time | 01:20:74 | 57:87 | 01:09:19 | 58:11 | / |
| Topspeed | 132 | 126 | 136 | 138 | / |
| Minspeed | 21 | 39 | 24 | 36 | / |
| Damage | 0 | 0 | 0 | 0 | 9274 |
| Fuel | 93:51 | 93:03 | 93:15 | 93:22 | 89:68 |
| **Froza** | | | | | |
| | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Lap Time | 03:10:33 | 02:48:91 | 02:46:70 | 03:17:55 | 02:08:67 |
| Topspeed | 149 | 247 | 238 | 251 | 234 |
| Minspeed | 22 | 58 | 19 | -92 | 58 |
| Damage | 0 | 0 | 0 | 706 | 9 |
| Fuel | 92:17 | 77:69 | 90:41 | 88:60 | 60:55 |
| **E-Road** | | | | | |
| | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 02:51:25 | 02:18:31 | 02:32:39 | 02:13:23 | 01:51:81 |
| Topspeed | 149 | 160 | 184 | 184 | 208 |
| Minspeed | 24 | 57 | 41 | 36 | -46 |
| Damage | 0 | 0 | 0 | 0 | 464 |
| Fuel | 92:76 | 91:61 | 93:00 | 93:11 | 90:55 |
| **CG-Speedway Number1** | | | | | |
| | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:32:60 | 01:20:57 | 01:22:60 | 01:16:75 | 51:89 |
| Topspeed | 149 | 190 | 193 | 199 | 202 |
| Minspeed | 38 | 58 | 38 | 55 | 53 |
| Damage | 0 | 0 | 0 | 0 | 62 |
| Fuel | 93:11 | 91:72 | 93:01 | 92:61 | 92:41 |

Figure 36 displays the damage rate for one lap of each controller, note that the AD5 and AD3 controllers have the highest damage values especially in Dirty tracks and Type ”oval” but in the tracks (”CGSpeedWayNumber1”, ”Roed-E”, ”Froza”) we do not have too much risk.

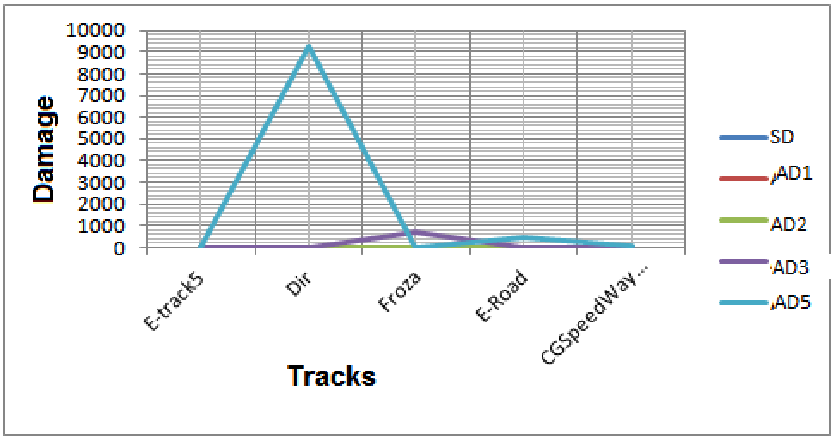


**Fig. 36.** Damage rate in one lap.

**Five lap race:** In this phase, we will do the same but with 5 laps for each track (see table 6) and got the following results (see table 9 ).

**Table 9.** Results of five laps race

| **E-Track5** | | | | | |
|---|---|---|---|---|---|
| **5 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:05:15 | 01:21:24 | 04:55:99 | 59:37 | 29:87 |
| Topspeed | 149 | 148 | 150 | 170 | 199 |
| Minspeed | 46 | 44 | 60 | 78 | 172 |
| Lastlap Time | 01:05:17 | 01:31:36 | 01:02:55 | 59:38 | 30:07 |
| Best lap | 4 | 02 | 01 | 4 | 2 |
| Damage | 0 | 0 | 0 | 0 | 0 |
| Fuel | 71:22 | 71:25 | 77:55 | 77:26 | 77:32 |
| Lap | 5/5 | 5/5 | 5/5 | 5/5 | 5/5 |
| **Dir1** | | | | | |
| **5 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:12:96 | 50:48 | 02:23:12 | 49:84 | 35:51 |
| Topspeed | 132 | 126 | 136 | 145 | 139 |
| Minspeed | 21 | -12 | 43 | 25 | -54 |
| Lastlap Time | 01:13:08 | 55:59 | 02:23:12 | 50:64 | 01:02:91 |
| best lap | 14 | 4 | 0 1 | 2 | 5 |
| Damage | 0 | 502 | 0 | 0 | 9274 |
| Fuel | 85:04 | 76:59 | 86:69 | 86:44 | 86:06 |
| Lap | 5/5 | 5/5 | 1/5 | 5/5 | 5/5 |
| **Froza** | | | | | |
| **5 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 03:02:89 | 02:39:82 | 07:14:90 | 02:47:14 | 02:20:27 |
| Topspeed | 149 | 248 | 209 | 250 | 246 |
| Minspeed | 23 | 41 | -53 | -86 | 63 |
| Lastlap Time | 03:02:91 | 02:48:06 | 03:20:16 | 03:34:52 | 03:00:55 |
| Best lap | 4 | 4 | 01 | 1 | 04 |
| Damage | 0 | 0 | 2269 | 2621 | 0 |
| Fuel | 85:51 | 85:18 | 86:46 | 86:06 | 76:55 |
| Lap | 5/5 | 5/5 | 4/5 | 5/5 | 5/5 |

from Table 9, we noticed that for each lap, our controller tries to minimize driving time and use a maximum speed without consuming more fuel and decreasing damage in all tracks, it improves driving to give us more optimal results then the last lap.

**20 lap race:** In this section, we set the distance to 20 laps for each track and for each controller.

| **E-Road** | | | | | |
|---|---|---|---|---|---|
| **5 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 02:44:93 | 02:11:39 | 04:55:92 | 02:09:77 | 01:25:66 |
| Topspeed | 149 | 202 | 175 | 209 | 207 |
| Minspeed | 24 | -58 | -50 | -58 | -55 |
| Lastlap Time | 02:44:93 | 01:29:02 | 06:08:49 | 05:56:16 | 01:27:38 |
| Best lap | 3 | 5 | 1 | 2 | 03 |
| Damage | 0 | 2013 | 3018 | 4266 | 2239 |
| Fuel | 87:97 | 92:51 | 91:00 | 90:76 | 84:99 |
| **CG-Speedway Number1** | | | | | |
| **5 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:25:01 | 01:25:00 | 01:18:13 | 01:12:24 | 53:61 |
| Topspeed | 149 | 190 | 194 | 200 | 178 |
| Minspeed | 37 | 57 | 32 | -54 | -51 |
| Lastlap Time | 01:25:14 | 01:14:59 | 01:29:65 | 01:12:81 | 01:46:47 |
| Best lap | 3 | 3 | 2 | 2 | 02 |
| Damage | 0 | 0 | 0 | 108 | 114 |
| Fuel | 90:51 | 91:00 | 90:76 | 84:99 | 80:65 |
| Lap | 5/5 | 5/5 | 3/5 | 5/5 | 5/5 |

The table (7.2 ) above shows that among the five drivers, the minimum time has been reached by the AD5 controller in each of the five tracks, this was due to the steering unit of the target. The target position allows the car to adjust safely the required steering angle with the maximum speed so the controller AD5 gave the best time.
However, taking a smaller target angle required a greater distance to cover by car by taking a less efficient trajectory. In addition, the number of times the car was off the road on Dir1 and Forza was higher for AD5 and, accordingly, the time that the car has passed out of the track was potentially higher than the other controllers.

Thus, higher damages happened after the collision with the outer walls of the track when the car is off the track. The simple driver SD driver and AD1, AD2 AD3 especially got less damage compared the version of AD5. The simple SD driver finished the race without damage, while AD5 was able to complete all the tracks safely except the Dir1t4 and Forza circuit where the damage is a little high for the AD3 driver (fig.10) .But the result may be more favourable than the other results by SD, DA1, DA2.
For the fuel consumption depends percentage of shocks and category of circuits.

**Time trial races for 2 seconds** Now, we set the race stopping criterion to 2s and TORCS takes a decision in 0.02s (20 ms) so each controller is tested 100 actions. we got the results that are shown in Figure 37 and Figure 38 which

**Table 10.** Results for 20 laps

| E-Track5 | | | | | |
|---|---|---|---|---|---|
| **20 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:05:15 | 01:21:24 | 04:33:24 | 59:37 | 29:87 |
| Topspeed | 149 | 148 | 203 | 160 | 199 |
| Minspeed | 46 | 44 | 60 | 78 | 172 |
| Lastlap Time | 01:05:17 | 01:31:36 | 01:02:55 | 59:37 | 30:07 |
| Best lap | 14 | 02 | 18 | 4 | 8 |
| Damage | 0 | 0 | 0 | 0 | 0 |
| Fuel | 71:22 | 71:25 | 77:45 | 77:45 | 77:02 |
| Lap | 20/20 | 20/20 | 20/20 | 20/20 | 20/20 |
| **Dir1** | | | | | |
| **20 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:12:96 | 50:43 | 02:23:12 | 49:84 | 35:51 |
| Topspeed | 132 | 126 | 136 | 145 | 139 |
| Minspeed | 21 | -52 | -43 | 25 | -54 |
| Lastlap Time | 01:13:08 | 02:33:43 | 02:23:12 | 02:23:94 | 01:02:91 |
| Best lap | 14 | 07 | 01 | 2 | 5 |
| Damage | 0 | 7438 | 1889 | 905 | 9274 |
| Fuel | 65:04 | 79:26 | 79:98 | 79:26 | 66:06 |
| Lap | 20/20 | 14/20 | 01/20 | 11/20 | 20/20 |
| **Froza** | | | | | |
| **20 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 03:03:66 | 02:39:82 | 07:14:90 | 02:47:14 | 02:20:27 |
| Topspeed | 149 | 248 | 209 | 250 | 246 |
| Minspeed | 22 | -41 | -53 | -86 | 63 |
| Lastlap Time | 03:03:66 | 02:47:26 | 09:55:62 | 03:14:23 | 09:21:23 |
| Best lap | 20 | 04 | 01 | 2 | 04 |
| Damage | 0 | 468 | 2269 | 9704 | 0 |
| Fuel | 62:61 | 71:88 | 60:55 | 65:85 | 60:55 |
| Lap | 20/20 | 20/20 | 04/20 | 20/20 | 20/20 |

represent the change in the target angle and "steer" function for each controller in a second.

We have noticed:

-The Trajectory of the target angle in SD controllers, AD1, AD2, AD3 are almost identical and the same for 2 seconds in a road type of track (see fig. **??** ).
-According to Figure 37, the car seems to be lost in the first seconds while the steer of the AD5 controller has a constant value.

| E-Road | | | | | |
|---|---|---|---|---|---|
| **20 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 02:44:93 | 02:11:39 | 04:33:59 | 01:85:16 | 01:25:66 |
| Topspeed | 149 | 202 | 203 | 150 | 207 |
| Minspeed | 24 | -58 | 32 | -58 | -55 |
| Lastlap Time | 02:45:08 | 03:00:20 | 04:51:53 | 05:34:85 | 01:27:38 |
| Best lap | 3 | 3 | 17 | 2 | 03 |
| Damage | 0 | 6056 | 7309 | 9602 | 2239 |
| Fuel | 62:11 | 78:25 | 79:25 | 72:01 | 55:22 |
| Lap | 20/20 | 9/20 | 20/20 | 10/20 | 20/20 |
| **CG-Speedway Number1** | | | | | |
| **20 laps** | **SD** | **AD1** | **AD2** | **AD3** | **AD5** |
| Best Time | 01:25:00 | 01:12:99 | 01:12:23 | 01:12:24 | 53:61 |
| Topspeed | 149 | 191 | 194 | 200 | 178 |
| Minspeed | 49 | 57 | 32 | -53 | -51 |
| Lastlap Time | 01:25:01 | 01:12:99 | 01:29:65 | 03:15:39 | 01:46:47 |
| Best lap | 12 | 20 | 02 | 07 | 02 |
| Damage | 0 | 0 | 0 | 1790 | 114 |
| Fuel | 66:66 | 62:58 | 73:38 | 75:66 | 60:65 |
| Lap | 20/20 | 20/20 | 03/20 | 13/20 | 20/20 |

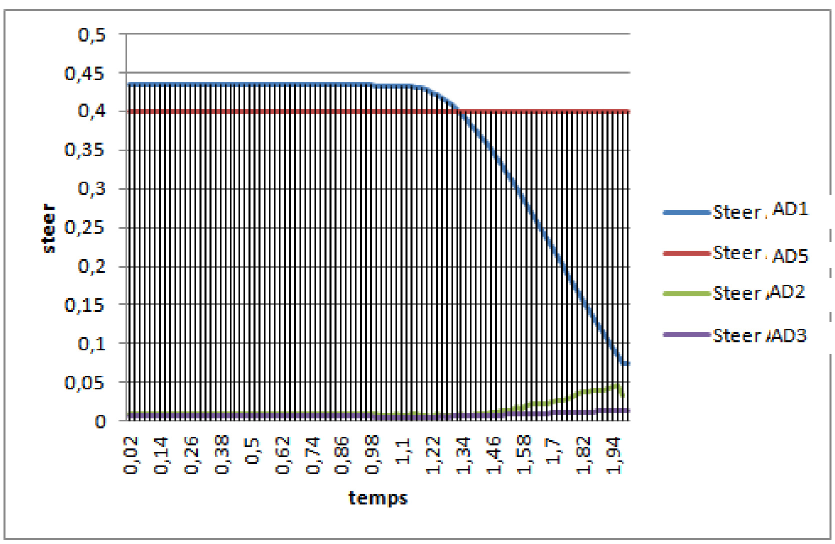


**Fig. 37.** Steer values for 2seconds in CG speedWay Number1 track

**Time trial races for 1 minute** Now, we set the race stopping criterion to 60s and TORCS. The results are in table11 and Figures Fig.40 , Fig.39, Fig.41.

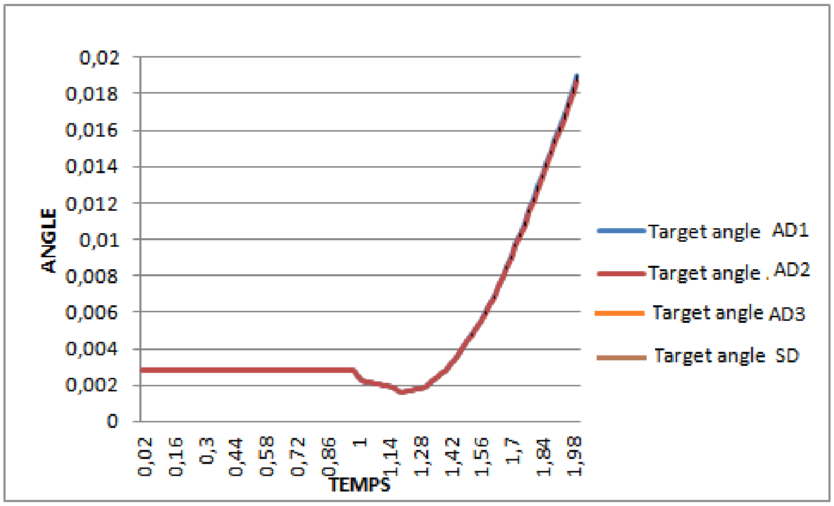


**Fig. 38.** Target angle variations for 2 s in CG speedWay Number1 rack

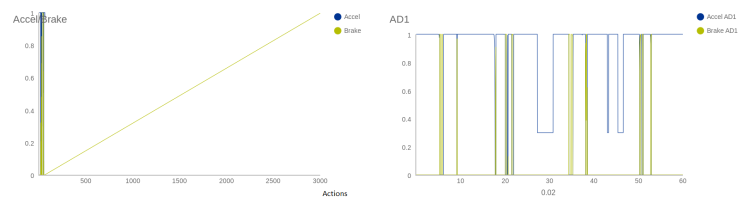


**Fig. 39.** Acceleration and brake values of AD5 and D1 in 1 minute race(CG-SpeedWay Number1 track)..

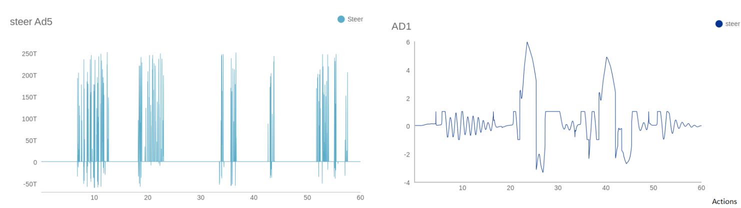


**Fig. 40.** Steer values for of AD5 and D1 in 1 minute race(CG-SpeedWay Number1 track).

According to the figures above (Fig.40 , Fig. 39, Fig. 41 ) and Table 11 , we deduce that the AD5 controller remains the most stable than other .only in special cases the other controllers can have better results.

**Table 11.** Results of the controllers in 1 minute time trial race

| | | | |
|---|---|---|---|
| **E-Track5** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 00:00 | 810,865 | 152 |
| AD2 | 00:00 | 1304,172 | 149 |
| AD3 | 00:00 | 5202,99 | 154 |
| AD5 | 29:20 | 489 | 198 |
| **Dir1** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 57:52 | 3260,43 | 166 |
| AD2 | 00:00 | 1156,22 | 137 |
| AD3 | 00:00 | 2934 | 139 |
| AD5 | 00:00 | 827 | 153 |
| **Froza** | | | |
| | **Best Time** | **Distance(m)** | **Top speed(m/s)** |
| AD1 | 02:49:12 | 17352,3 | 247 |
| AD2 | 02:49:12 | 17354,12 | 239 |
| AD3 | 03:44:56 | 17355 | 205 |
| AD5 | 57:66 | 17355,2 | 203 |
| **E-Road** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 00:00 | 1630 | 160 |
| AD2 | 00:00 | 1304 | 170 |
| AD3 | 00:00 | 2282 | 174 |
| AD5 | 01:00:00 | 3260 | 199 |
| **CG-Speedway Number1** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 00:00 | 1654 | 175 |
| AD2 | 00:00 | 620,25 | 160 |
| AD3 | 00:00 | 1860,75 | 184 |
| AD5 | 54:22 | 2687,75 | 200 |

**Time trial races for 5 minutes** Now, we set the race stopping criterion to 300s and TORCS. The results are in Table12)

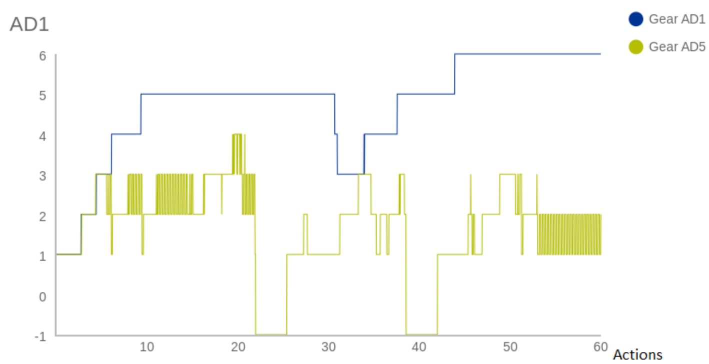


**Fig. 41.** Gear values for of AD5 and D1 in 1 minute race(CG-SpeedWay Number1 track).

### 7.3 Real race

**5 laps race** In this case, we test our controller "AD5", without considering the variables and opponents sensors.

Our main goal is to test the performance of "AD5" in a race, and answer the following questions: Could we have a perfect driving in a race with "AD5" only with the track borders sensors? Can we win the race with AD5?
In this context we tested "AD5" in 5 tracks see Table 6 and for the opponents, we chose from the following cars (see table 13 ).

**Table 12.** Results of the controllers in 5 minute time trial race

| **E-Track5** | | | |
|---|---|---|---|
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 01:11:98 | 4540,844 | 155 |
| AD2 | 02:15:26 | 8803,161 | 155 |
| AD3 | 02:13:46 | 20233,85 | 160 |
| AD5 | 29:30 | 32600 | 199 |
| **Dir1** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 52:27 | 7825,032 | 126 |
| AD2 | 00:00 | 5202,99 | 134 |
| AD3 | 00:00 | 652 | 145 |
| AD5 | 00:00 | 620,25 | 153 |
| **Froza** | | | |
| | **Best Time** | **Distance(m)** | **Top speed(m/s)** |
| AD1 | 02:49:12 | 17352,3 | 247 |
| AD2 | 02:49:12 | 17354,12 | 239 |
| AD3 | 03:44:56 | 17355 | 205 |
| AD5 | 57:66 | 17355,2 | 203 |
| **E-Road** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 03:62:33 | 11586,2 | 160 |
| AD2 | 02:41:41 | 11588 | 208 |
| AD3 | 02:13:73 | 23136,4 | 210 |
| AD5 | 01:15:16 | 23139 | 217 |
| **CG-Speedway Number1** | | | |
| | **Best Time** | **Distance** | **Top speed** |
| AD1 | 01:15:19 | 23136 | 195 |
| AD2 | 01:15:80 | 17352,2 | 193 |
| AD3 | 01:10:20 | 28920,4 | 200 |
| AD5 | 44:67 | 28921,6 | 206 |

After the launch of the race "AD5" against the 10 cars of each team (list in table 13) for 5 laps in each track we obtained the following results (see table 14)

**Table 13.** TORCS teams and cars

| Num | Team | Drivers | Cars |
|---|---|---|---|
| 1 | **berwin** | berwin 1, berwin 2, berwin 3, berwin 4, berwin 5, berwin 6, berwin 7, berwin 8, berwin 9, berwin 10 | car1-stock1, car1-stock1, car1-trb1, car2-trb1, car3-trb1, car4-trb1, car5-trb1, car6-trb1, car7-trb1, car1-trb3, |
| 2 | **bt** | bt1 bt2 ,bt3 ,bt4 ,bt5 ,bt6 ,bt7 ,bt8 ,bt9 ,bt10. | car1-stock1, car1-stock1, car1-trb1, car2-trb1, car3-trb1, car4-trb1, car5-trb1, car6-trb1, car7-trb1, car1-trb3, |
| 3 | **damned** | damned1 ,damned2 ,damned3 ,damned4 ,damned5 ,damned6 ,damned7 ,damned8 ,damned9 ,damned10 . | Ford Fucus WRC-offroad-4WD GrA, Peugeot 206 WRC -offroad-4WD GrA, Peugeot 306 Maxi -offroad-4WD GrA, Mitsibishi lancer -offroad-4WD GrA, Subaru Inperza -offroad-4WD GrA,, Toyata Corolla -offroad-4WD GrA, car5-trb1, car6-trb1, car7-trb1, car1-trb3. |
| 4 | **inferno** | inferno1, inferno2, inferno3, inferno4, inferno5, inferno6, inferno7, inferno8, inferno9, inferno10. | Car1-ows1, Peugeot 406-Track FWD Grb, car1-trb1, car2-trb1, car3-trb1, car4-trb1, car5-trb1, car6-trb1, car7-trb1, car1-trb3. |
| 5 | **tita** | tita1, tita2, tita3, tita4, tita5, tita6, tita7, tita8, tita9, tita10. | Car1-ows1, Peugeot 406-Track FWD Grb, car1-trb1, car2-trb1, car3-trb1, car4-trb1, car5-trb1, car6-trb1, car7-trb1, car1-trb3. |

From Table 14, we noticed that our controller has won a race only in some types of tracks such as "E-track5" category "Oval track" and only against some car types like cars of category "stock1" ,"ows","trb".

In some cases on other roads, our controller gives us the best results, but in a very short time, for example in the beginning of the race in the E-Road track, against the berwin team or our bt controller could go first turn by a ranking between "1 and 4" and a maximum speed equal to 199 but it has little damage when the other cars overtaken it.

When our controller is in parallel with his opponent, and the distance between them is smaller, and they are in a smaller width track, there is a possibility of damage (see fig.42 ).

According to Fig. **??** despite our controller can win the race on the E-track5 but it had a higher crash rates in the majority of cases.

**Table 14.** Results of AD5 in a real race

| **E-Track5** | **Against berwin team** | **Against bt team** | **Against damned team** | **Against in-ferno team** | **Against tita team** |
|---|---|---|---|---|---|
| Ranking | 1/11 | 2/11 | 1/11 | 7/11 | 4/11 |
| race time | 02:36:78 | 02:40:38+ 03:41 | 02:42:73 | 02:15:81 + lap | 02:18:74 + 28:19 |
| Best time | 29:90 | 30:28 | 30:28 | 30:59 | 30:53 |
| Maxspeed | 198 | 198 | 198 | 199 | 198 |
| Damage | 2267 | 7939 | 5888 | 5232 | 8043 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 5/5 | 5/5 | 5/5 | 4/5 | 5/5 |
| **CG-SpeedWay Number1** | **Against berwin team** | **Against bt team** | **Against damned team** | **Against in-ferno team** | **Against tita team** |
| Ranking | 11/11 | 11/11 | 11/11 | 11/11 | 11/11 |
| Race time | 03:39:12 + 44:76 | 03:34:00 + 43:19 | 03:34:75 + Lap | 02:59:83 + 3 Lap | 03:01:75 + 4 laps |
| Best time | 46:16 | 48:56 | 47:23 | 46:06 | 03:41:41 |
| Maxspeed | 205 | 202 | 199 | 200 | 198 |
| Damage | 5533 | 3096 | 3093 | 5501 | 0 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 5/5 | 5/5 | 4/5 | 2/5 | 1/5 |
| **E-Road** | **Against berwin team** | **Against bt team** | **Against damned team** | **Against in-ferno team** | **Against tita team** |
| Ranking | 11/11 | 11/11 | 11/11 | 11/11 | 11/11 |
| Race time | 06:00:00 +lap | 06:14:64 +3lap | 06:13:26 +2 Laps | 04:50:42 +2lap | 04:56:16 +3lap |
| Race time | 01:21:29 | 02:19:92 | 01:39:54 | 01:27:27 | 01:48:18 |
| Maxspeed | 205 | 201 | 212 | 211 | 208 |
| Damage | 9979 | 10362 | 4421 | 7685 | 5593 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 4/5 | 2/5 | 3/5 | 3/5 | 2/5 |

Figure 43 shows the different values of the speed of our AD controller 5 against some TORCS teams in E-track5, we observed that in the majority of cases our car is very slow compared to the other cars in each team.

### 7.4 Real race with ow1 car

To validate our controllers results, we replace our race car by another car "OW1". We have run each of our controllers against a car from each team after tests

| Froza | Against berwin team | Against bt team | Against damned team | Against inferno team | Against tita team |
|---|---|---|---|---|---|
| Ranking | 10/11 | 11/11 | 11/11 | 10/11 | 10/11 |
| Race time | 7:44:99 + Lap | 8:07:39 + 02:16:32 | 08:09:42 + 1Lap | 6:40:33 + Lap 0 | 6:37:02 + Lap |
| Best time | 01:57:58 | 01:45:30 | 02:03:00 | 02:04:00 | 02:04:37 |
| Maxspeed | 243 | 282 | 260 | 220 | 217 |
| Damage | 2275 | 95 | 475 | 416 | 1185 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 4/5 | 5/5 | 4/5 | 4/5 | 4/5 |
| **Dir1** | **Against berwin team** | **Against bt team** | **Against damned team** | **Against inferno team** | **Against tita team** |
| Ranking | 11/11 | 11/11 | 11/11 | 11/11 | 10/11 |
| Race time | 03:03:41+ 5 laps | 02:55:54 + 5 laps | 03:05:90 + 5 laps | 02:55:54 + 5 laps | 02:48:12+ 4 Laps |
| Best time | 00:00 | 00:00 | 00:00 | 00:00 | 00:00 |
| Maxspeed | 146 | 141 | 145 | 145 | 145 |
| Damage | 10017 | 6462 | 39 | 226 | 271 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |

yielded the following results:

**AD1 in one lap race** Table 15 displays the AD1 Controller test results in a race, it was noted that he can not win the race and there are some difficulties for example it can not overtake a car without damage in a curve, and sometimes it can not finish the race until the end.

**Table 15.** AD1 in one lap race with ow1 car

| Froza | AD 1 | berwin1 | bt 2 | inferno 3 | tita 4 |
|---|---|---|---|---|---|
| Ranking | 5/5 | 2/5 | 01/5 | 03/5 | 4/5 |
| Race time | 01:45:26 + 58:49 | 01:45:26 + 02:10 | 01:45:26 | 1:45:26 + 02:35 | 1:45:26 + 02:50 |
| Best time | 02:43:58 | 01:47:37 | 01:45:26 | 01:47:62 | 01:47:76 |
| Maxspeed | 299 | 287 | 280 | 289 | 289 |
| Damage | 321 | 0 | 489 | 390 | 0 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 1/1 | 1/1 | 1/1 | 1/1 | 1/1 |

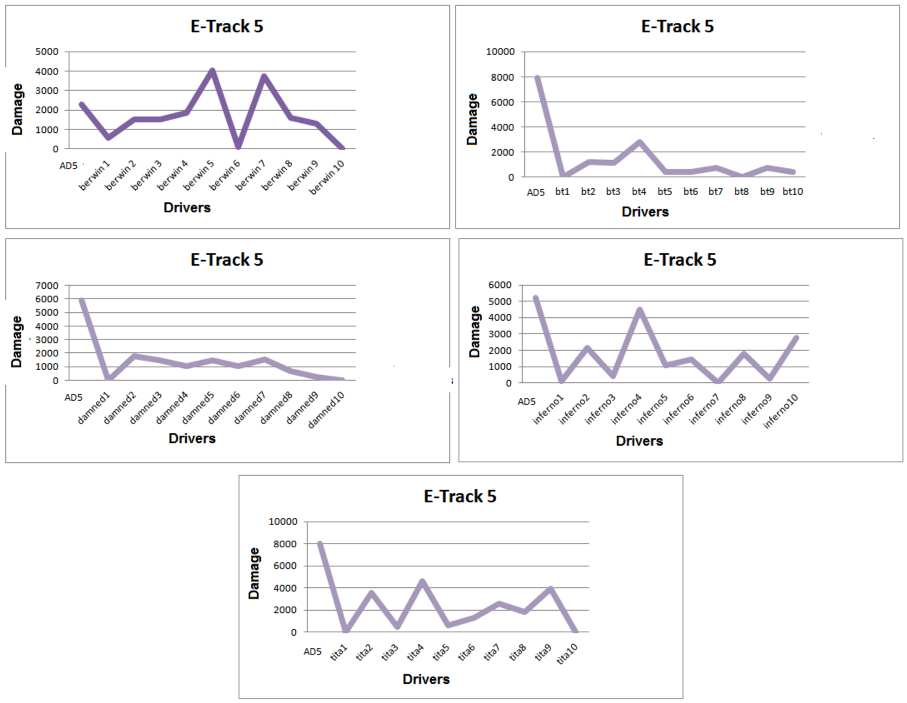


**Fig. 42.** Damage values of AD5 in E-track5.

**AD2 in one lap race** The following table 16 shows that in one lap, although the AD2 controller does not give the best results but at least guarantee correct behaviour with minimum shock rate and speed value maximization comparing with previous results (driving without an opponent)

**Table 16.** Results of AD2 controller in one lap with ow1 car

| **Froza** | **AD 2** | **berwin1** | **bt 2** | **inferno 3** | **tita 4** |
|---|---|---|---|---|---|
| Ranking | 5/5 | 1/5 | 02/5 | 03/5 | 4/5 |
| Race time | 01:40:59+ 01:03:71 | 01:40:59 | 01:40:59 + 04:08 | 01:40:59 + 04:34 | 01:40:59 + 04:74 |
| Best time | 02:44:31 | 01:40:59 | 01:44:68 | 01:44:93 | 01:45:34 |
| Maxspeed | 244 | 297 | 284 | 284 | 282 |
| Damage | 0 | 0 | 18 | 796 | 1198 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 1/1 | 1/1 | 1/1 | 1/1 | 1/1 |

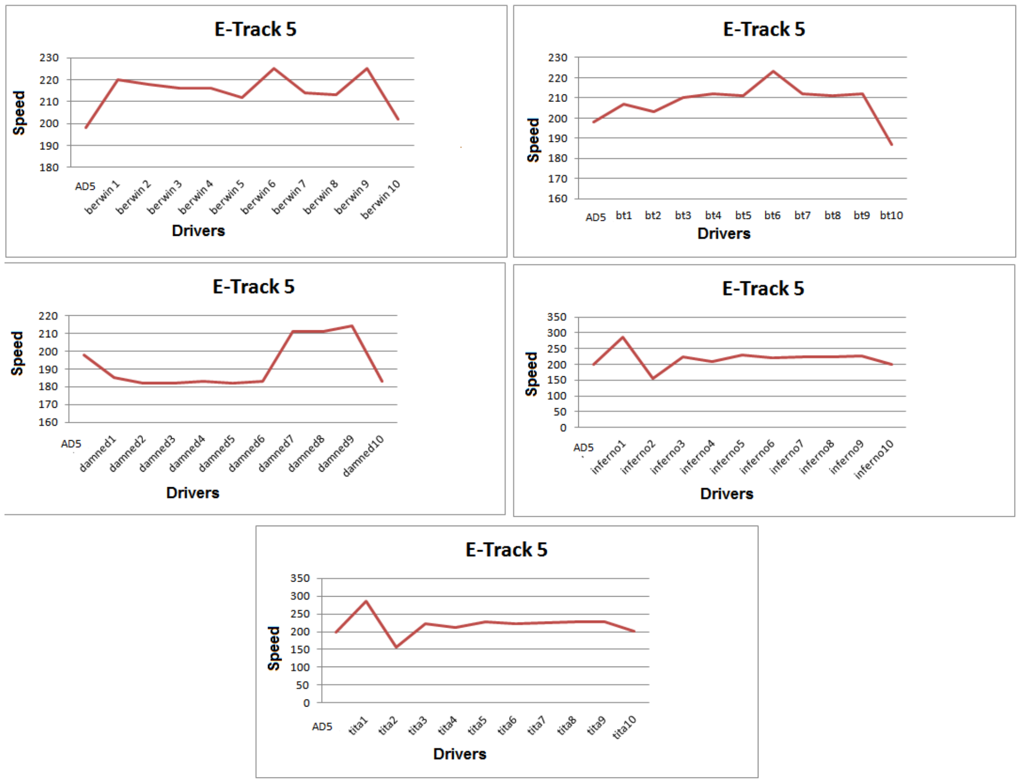


**Fig. 43.** Speed Variation of AD 5 in E-track5.

**AD3 in one lap race** as Table 17 shows, we have the same comments mentioned above with a risk of not finishing the race since the damage is very high in a short distance.

**Table 17.** Results of AD3 controller in one lap with ow1 car

| **Froza** | **AD 3** | **berwin1** | **bt 2** | **inferno 3** | **tita 4** |
|---|---|---|---|---|---|
| Ranking | 5/5 | 4/5 | 01/5 | 02/5 | 3/5 |
| Race time | 01:44:81+ 01:02:99 | 01:44:81+ 03:62 | 01:44:81 | 01:40:59 + 01:02 | 01:40:59 + 02:78 |
| Best lap | 02:47:80 | 01:48:44 | 01:44:81 | 01:45:83 | 01:47:6 |
| Maxspeed | 315 | 287 | 282 | 293 | 280 |
| Damage | 319 | 196 | 9 | 709 | 1192 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 1/1 | 1/1 | 1/1 | 1/1 | 1/1 |

**AD5 in one lap race** In this case, we noticed the same problems except that an improvement in time and speed (see table. 18 )

**Table 18.** Results of AD5 controller in one lap with ow1 car

| Froza | AD 5 | berwin1 | bt 2 | inferno 3 | tita 4 |
|---|---|---|---|---|---|
| Ranking | 5/5 | 1/5 | 02/5 | 03/5 | 4/5 |
| Race time | 01:39:28 +26:23 | 01:39:28 | 01:39:28 + 4:69 | 01:39:28 + 04:87 | 01:39:28 + 10:45 |
| Best time | 02:05:51 | 01:39:28 | 01:43:97 | 01:44:83 | 01:49:47 |
| Maxspeed | 255 | 306 | 297 | 286 | 273 |
| Damage | 289 | 0 | 427 | 563 | 277 |
| Pit stops | 0 | 0 | 0 | 0 | 0 |
| Lap | 1/1 | 1/1 | 1/1 | 1/1 | 1/1 |

**AD5 with opponents sensors** After adding the opponents sensors in the AD5 controller, we obtained the following results after 20 laps on the E-Track5 track :

**Table 19.** Results of AD5 with and without opponents sensors in E-Track5

| **AD5 without opponents sensors** | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Laps** | 1 | 2 | 3 | 4 | 5 | ... | 20 |
| **Distance** | 1621,73 | 3243,46 | 4865,19 | 6486,92 | 8108,65 | ... | 32434,6 |
| **Damage** | 0 | 0 | 0 | 0 | 0 | ... | 0 |
| **Time** | 0 | 0 | 0 | 0 | 0 | ... | 0 |
| **Speed** | 0 | 0 | 0 | 0 | 0 | ... | 0 |
| **with opponents sensors** | | | | | | | |
| **Laps** | 1 | 2 | 3 | 4 | 5 | ... | 20 |
| **Distance** | 1621,73 | 3243,46 | 4865,19 | 6486,92 | 8108,65 | ... | 32434,6 |
| **Damage** | 0 | 0 | 0 | 0 | 0 | ... | 0 |
| **Time** | 0 | 0 | 0 | 0 | 0 | ... | 0 |
| **Speed** | 0 | 0 | 0 | 0 | 0 | ... | 0 |