Autonomous Vehicle to Sprint on Indian Roadways using CARLA Simulator

Utsav Parashar, Pooja N, Utkarsh Gupta, Moulya DG

Bachelor of Engineering, Department of Electronics and Communication
Dayananda Sagar College of Engineering, Bengaluru -78, Karnataka, India

---

Abstract - *In the modern era, the vehicles are focused to be automated to give human driver relaxed driving. In the field of automobile various aspects have been considered which makes a vehicle automated. There are several levels to this automation scale which also shows the dependency on the human driver. Many big companies are already working on self-driving cars for many years and developing new features to give a whole new level to the automated vehicles. In this paper we focus on a simulation of an autonomous vehicle on a dynamic environment; the software used is CARLA which is an open-source software and works on the Unreal gaming engine. The code for the simulation is uploaded in our car to take care of everything just like a human would do to reach the destination, while we remain seated carefree on the seats. This code drives it on the roads, bridges, through the crowded places, past the crossings to the destination, also without violating the traffic rules. A fully computerized car is capable of doing almost everything someone would ask for. As a matter of fact, vehicles all over the world are now fitted with intelligent devices that make the vehicles to respond to various factors – be it climate control, sudden accelerations or braking or even self-repair of modules.*

***Key Words*: python client, lateral control, state estimation, PID, DARPA.**

1. INTRODUCTION

A high majority of accidents today happen from human errors including traffic conjections. A lot of cities in India have became some of the world's largest traffic congested cities. Self-Driving car is a sleeping gaint which can change everything including road safety by lowering the no. of accidents, mobility, while dramatically reducing the cost and pollution of driving. Before we program self-driving cars on roads of real-world we must do the same on virtual world of simulation. Because a mistake in program may end up dangerously and may cost life sometimes as has happened earlier.

The Computer Vision Center of Barcelona and UnrealEngine have developed a simulating environment in Java and C languages which provides a random simulating environment same as compared to the real-world and is capable of designing a client based autonomous vehicle to sprint on roadways. We have constructed longitudinal and lateral dynamic models for a vehicle and created controllers that regulate speed and path tracking performance using python. We have tested the limits of our control design and learned the challenges faced in driving also at the limit of vehicle performance.

2. LITERATURE REVIEW

By the passage of time, a lot of work has been done to make cars self-driven, but due to technological advancement in the roads and the increasing population it has been difficult for "perfection" to become true.

In the pre-computer days of the 1930s, the driverless cars were only science fiction things. But the development of the digital computer made possible to think of self-driven vehicles outside the fiction. The concept of self-driving car is as old as the invention of car itself; in year 1933 Norman Bel Geddes represented driverless car in an exhibition sponsored by General Motors at the World Fair depicting electric cars which was powered by embedded circuits and were radio controlled *[2]*. By the 1960s the self-driven cars were only thought to navigate on ordinary streets on their own. German pioneer Ernst Dickmanns, in the 1980s, got a Mercedes van to drive hundreds of miles autonomously on highways, a great achievement with respect to the computer technology at those times *[3]*. In the mid-2000s, the Defense Advanced Research Projects Agency (DARPA) created some "Grand Challenges" where groups would compete with self-driving vehicles *[4]*. In 2009, Google began its own self-driving car venture, including colleagues who had effectively devoted years to the innovation. By 2012 the Google car was in the road for testing *[5]*. By the passing years, the car was developed and equipped with multiple sensors, Radars, Lasers, Global Positioning System (GPS); it uses heavily detailed maps, and many other things to safely drive and navigate itself with no human interaction.

The Computer Vision Center of Barcelona and Unreal Engine developed a simulating environment to provide a random simulating environment same as compared to real-world and is capable of designing a client based autonomous vehicle to sprint on Roadways named CARLA and published its paper in August 2014 *[6]*. In 2016 This software has became an open source to check our codes and efficiency for the self-driving car control. In this paper, we have designed program for two controls of an autonomous vehicle control namely, longitudinal control *[7]* and lateral dynamics *[8]* which can help the driver to relax for the certain duration of time. This paper presents an overview of our work on CARLA using these control algorithms.

3. SIMULATION

CARLA Simulator has been developed with integrating the Unreal Engine in Java and C programming languages to provide an environment compared to real world scenario with calling a random function – rand() in order to gain the uncertainty and unpredictability as in the real world. We needed to program our vehicle in python client's API by including those modules and programming files and dumping the codes into python client's folder. With some commands and those codes we had achieved the longitudinal and lateral controls on the vehicle, as the vehicle stops on red light, seeing a static vehicle, or when a pedestrian crosses. Also the car is capable of increasing and decreasing its speed to the speed limit provided along the road-curbs. We have developed the complete model of a vehicle's motion, defining a PID controller for longitudinal control, defining a path following controller for lateral vehicle control and tested our own control designs in the CARLA Simulator.

The state estimation diagram is given below. State estimator combines the vehicle's position with the world model to produce a discrete and semantically rich representation of the vehicle's logical position with the RNDF. Goal selector uses the current logical location as reported by state estimator to generate the next series of local goals for execution by the motion planner; these will be either lane goals or zone goals.

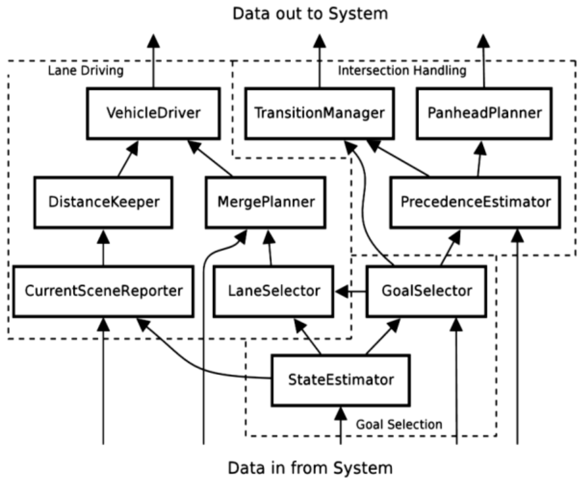

Figure 1: State Estimation Diagram

CARLA, an open-source simulator has been developed from the ground up to support development, training, and validation of autonomous urban driving systems. In addition to open-source code and protocols, CARLA provides open digital assets (urban layouts, buildings, vehicles) that were created for this purpose and can be used freely. The simulation platform supports flexible specification of sensor suites and environmental conditions. We use this software to study the performance of three approaches to autonomous driving: a classic modular pipeline, an end to-end model trained via imitation learning, and an end-to-end model trained via reinforcement learning. The approaches are evaluated in controlled scenarios of increasing difficulty, and their performance is examined via metrics provided by CARLA, illustrating the platform's utility for autonomous driving research.

4. MATHEMATICAL MODELING

From the previous read papers we discovered various methods for motion planning of self-driving cars in terms of lateral as well as longitudinal control. The longitudinal control is achieved by simple constant gain method, matrices and linear equations, we have complex algorithms like normalized optimization techniques. With the above mentioned algorithms we have implemented PID (proportional, integral and derivative) controls for linear motion. We have also analyzed its steady responses for constants used in equation in MATLAB software.

The PID control algorithm is given for an input, u(t) which has a reference velocity of v_d with instantaneous speed of v. The difference between v_d & v is considered to be an error and denoted by e(t).

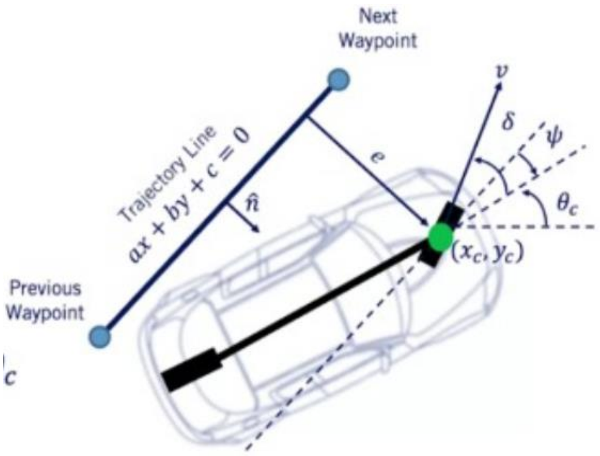

Figure 2: Parameters for Lateral Control

The final equation of the PID Control is given by:

$$u(t) = K_p\, e(t) + K_i\, e(t) + K_d\, e'(t) \text{ or,}$$

$$u(t) = K_p\, (v_d-v) + K_i \int (v_d-v)dt + K_d\, d(v_d-v)/dt$$

While for lateral control we have found different errors through papers and included them for our programming.

Cross track error :

$$e=axc+byc+c \,/\, \sqrt{a2b2}$$

Cross track steering :

$$tan^{-1}(k_e/v)$$

Heading error :

$$\Psi=tan^{-1}(-a/b) - \theta_c$$

With Total steering input,

$$\delta = \Psi+tan^{-1}\,(k_e/v)$$

The above mentioned equations are written in python language and are implemented in our simulation software.

5. RESULT

The simulation has been run successfully on a laptop computer and the output has been shown below. The software environment shows various parameters and values in the environment. Parameters like speed limit in the current street are recorded by the vehicle and speed is maintained according to that. Location and orientation coordinates are also specified for reference.

Figure 3: Simulation Output

We have implemented lateral and longitudinal controls along with mathematical modelling and state estimation of a vehicle in the simulator and have achieved the partial simulation results. The software accesses the scenarios in waypoints and store that data as a text file named —trajectory.txt in its folder named controller output.

6. CONCLUSION

CARLA is an advanced step for development of autonomous vehicles driving as we can write and try codes for them and simulate them in virtual world before executing on roads to avoid fatal accidents. With the help of this algorithm, vehicles can be set to automatically navigate according to the planning for destination location by continuously receiving the direction from another vehicle moving ahead to the same destination (including lane changing, Intersection merging, etc.). The navigation planning is also one of the vital aspects of autonomous systems. When the autonomous vehicle actually starts to move towards the planned route it may find random unknown obstacles in-between the existing location to the destined location, hence the needs for proper lateral and longitudinal controls come into picture. And the lateral and longitudinal dynamics algorithm implemented in coding for the simulation.

The potential applications of this robotic vehicle are to use them on highways or heavy traffic roads. These types of autonomous vehicles with further developments can even be used when a driver travels to newer areas.

REFERENCES

[1] "How will the driverless car impact society and the way people commute?" Cristina Bingham, National University, March 30, 2013

[2] Driverless car Documentation, Suddi Reddy Prashanth, Feb 23, 2011

[3] Self-Driving and Driver Relaxing Vehicle, Azam Rafique, Mehran University of Engineering and Technology, November 2016

[4] DARPA Urban Challenge Technical Paper, Team 'VictorTango', April 2007

[5] Waymo Safety Report by Google, 2018

[6] CARLA: An Open Urban Driving Simulator by Alexey Dosovitskiy, German Ros Felipe Codevilla, Antonio L´opez and Vladlen Koltun. 1. Collaboration between Intel Labs, 2. Toyota Research Institute, 3. Computer Vision Center, Barcelona

[7] "Modeling and design of cruise control system with feed-forward for all terrain vehicles", Khaled Sailan and Klaus. Dieter Kuhnert Siegen university, electrical engineering department Real time system institute Siegen, Germany

[8] Chalmers Vehicle Dynamics Compendium, Bengt Jacobson, Vehicle Dynamics Group, Division of Vehicle and Autonomous Systems, Department of Applied Mechanics, Chalmers University of Technology, Sweden