Simulation of Autonomous Racing Vehicle

Kasra Sinaei
kks6117@psu.edu

Course Advisor:
Dr. H. Pangborn
Department of Mechanical Engineering

EE-582

Revision 1.0

October 25, 2023

Table of Contents

1 Motivation **3**

2 Brief Literature review **3**

3 System Model & Simulation **4**

3.1 Bicycle Model 4

3.2 Simulation 5

4 Goals **5**

References **7**

List of Tables

1 Vehicle's physical properties 5

List of Figures

1 Bicycle model for a rear axle drive vehicle (reference: Kabzan, 2020 [1]) 4

2 Closed loop simulation of car around two F1 tracks (25 minutes at 10 $\frac{km}{hr}$) 6

Abstract

In this report, description and simulation of the chosen system for course project is introduced and elaborated. Autonomous racing comprises of variety of challenging problems in fields of automotive and control engineering. There is a trade-off between accurate modeling of the four-wheel racing vehicle's motion and keeping the computation complexity of the controller low enough to a degree that it can handle real-time operation of the system. High-fidelity models for tire simulation and vehicle's aerodynamic are not viable for controller design. Instead, simplified yet accurate mathematical descriptions such as bicycle model and Pajecka tire model are well known and commonly used for establishing controllers. In this report I briefly cover recent literature, details of getting equations of motion, formulating the model, simulating the system, and preliminary design specifications for Model Predictive Controller design.

1 Motivation

Driving a racing car around a circuit as fast as possible is intriguing as sporting event and research topic as well [2]. Model predictive control (MPC) is one of the widely used control approaches in driver-less racing control problems [3, 4]. An extensive review comprising of a comparison between different control approaches for autonomous driving is presented in [4] which points out utilization of Linear Quadratic Regulators (LQR), adaptive controllers, MPC and robust control in different settings of this problem. Advances in autonomous racing controller designs can improve the quality of commercial cars and driver's assistant technology as well [5, 6].

Upon successful completion of this project, I could practice modeling, linearization and discretization of a highly non-linear dynamic system, and also gain an insight over choosing the best MPC formula and its efficient implementation that suits such a model to achieve specific performance criteria.

2 Brief Literature review

Back in 2004 and 2005 a series of DARPA challenges attracted the focus of researcher to the problem of autonomous driving and racing [7]. Herein, we focus on the control problem only while the autonomous vehicle control involves overcoming challenging barriers regarding mechanical/aerodynamic design, perception, state estimation and mapping [1, 8].

Over the past couple of decades MPC has proven to be one of the most efficient tools in both optimizing the racing line and driving the vehicle on the designed path by achieving certain tracking goals [2, 9, 10]. Rosolia and Boreli formulated the racing line generation problem as a minimum time optimization and developed a Learning MPC (LMPC) such that it can reduce the computation burden of the existing approaches and resulted in improved lap time for a scaled RC car [3]. Wang et. al. utilized MPC for steering control used in the driver assistant systems; Linearization and discretization of systems dynamic using Euler method is also elaborated in [5]. Zeng et. al. proposed another MPC approach that better handles obstacle avoidance related constraints in the context of autonomous driving, they proposed a Control Barrier Function (CBF) based constraint for MPC optimization problem such that its performance does not diminish significantly by shortening the length of prediction horizon [11].

3 System Model & Simulation

3.1 Bicycle Model

Bicycle model is one of the well known and efficient approaches for deriving a 4 wheel vehicle's motion equations [12]. Physical parameters of the vehicle, exerted forces to the tires and velocity vectors have been illustrated in figure 1.

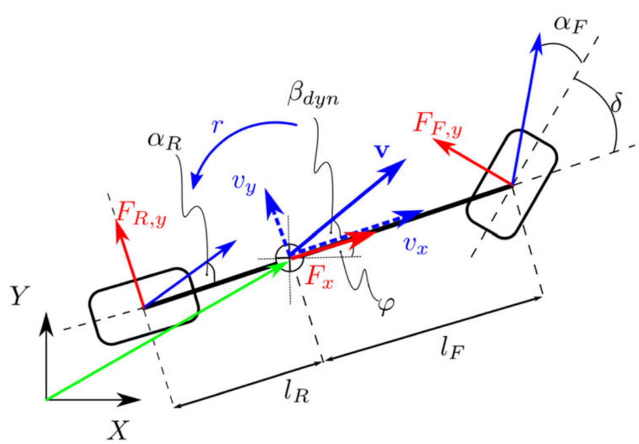

Figure 1. Bicycle model for a rear axle drive vehicle (reference: Kabzan, 2020 [1])

One can derive the non-linear dynamics of this physical system by picking the state vector as $x = \begin{bmatrix} v_x & v_y & \omega_z & e_\psi & s & e_y \end{bmatrix}$ where its entries refer to **longitudinal velocity**, **lateral velocity**, **heading rate of change**, **heading error** (w.r.t. path), **traversed distance on path** and **lateral position error** (w.r.t. track center line) respectively. Equation 1 shows the dynamics of the system [1]. This equation is used in the simulation.

$$x(k+1) = x(k+1) + \Delta T \begin{bmatrix} a - \frac{1}{m}(F_{y,F}\sin(\delta) + m v_y \omega_z) \\ \frac{1}{m}(F_{y,F}\cos(\delta) + F_{y,F}) - m v_x \omega_z) \\ \frac{1}{I_z}(F_{F,y} l_F \cos(\delta) - F_{R,y} l_R + \tau_{TV}) \\ \omega_z - (v_x \cos(e_\psi) - v_y \sin(e_\psi))\frac{\kappa}{1-\kappa e_y} \\ (v_x \cos(e_\psi) - v_y \sin(e_\psi))\frac{1}{1-\kappa e_y} \\ (v_x \cos(e_\psi) + v_y \sin(e_\psi)) \end{bmatrix} \tag{1}$$

System's parameter used in the equation are presented in table 1. Input of the system is $u = \begin{bmatrix} \delta & a \end{bmatrix}$ where first entry refers to steering angle and the second entry is thrust/break pressure ($a \in [-1, 1]$). Subscripts R/F refer to rear or front axle, while x and y refer to longitudinal and lateral respectively.

Table 1. Vehicle's physical properties

m	l_a	$F_{a,b}$	I_z
vehicles mass	distance from CoG to axle a	tire force of axle a, along axis b	inertial moment

Tire forces are derived using the Pacejka Model [13]. Outline of these equations are shown in equation set 2.

$$\begin{aligned} \alpha_R = -tan^{-1}\left(\frac{v_y - l_R\omega_z}{v_x}\right), \quad \alpha_F = \delta - tan^{-1}\left(\frac{v_y - l_F\omega_z}{v_x}\right) \\ F_{R,y} = D_R sin(C_R tan^{-1}(B_R\alpha_R)), \quad F_{F,y} = D_F sin(C_F tan^{-1}(B_F\alpha_F)) \end{aligned} \tag{2}$$

Note that $D_{R/F}$ are tires cornering stiffness, they could be determined from experimental data. Slip angles of rear tire and front tire are denoted by α_R and α_F respectively. The trust vectoring torque in the last row of state matrix of equation 1 is controlled using a simple proportional controller shown in equation 3. This torque vectoring controller determines how much force should be delivered to individual drive wheels to avoid tire slip.

$$r_{target} = \delta\frac{v_x}{l_F + l_R}, \qquad \tau_{TV} = (r_{target} - r) \tag{3}$$

3.2 Simulation

Using the discretized equations of system's motion and the GPS data of Formula One circuits [14], a simple proportional controller has been implemented to better observe the behavior of the simulated car. This controller adjusts the acceleration input and steering input such that the car travels around the selected circuit while maintaining a low velocity (smaller than 20 $\frac{km}{hr}$).

$$u_{cl} = \begin{bmatrix} K_\psi e_\psi \\ K_v(v_d - v_x) \end{bmatrix} \tag{4}$$

The proportional controller consists of two constant gains, K_ψ is steering gain that keeps the car close to the track center line and K_v tries to regulate the vehicle's velocity at desired value of v_d.

The only implemented object at the moment is *Simulation.m*; this class contains all the vehicle's property and its step function can take the input and simulates system for one step. Note that δt is the simulation step time and ΔT is the discretization time step.

4 Goals

Next step in this project will be implementing an appropriate form of MPC capable of driving the simulated vehicle in a predefined 2D track. Primary focus of this project will be on following a predefined racing line as fast as possible, while avoiding the track limit violations and crashing the vehicle.

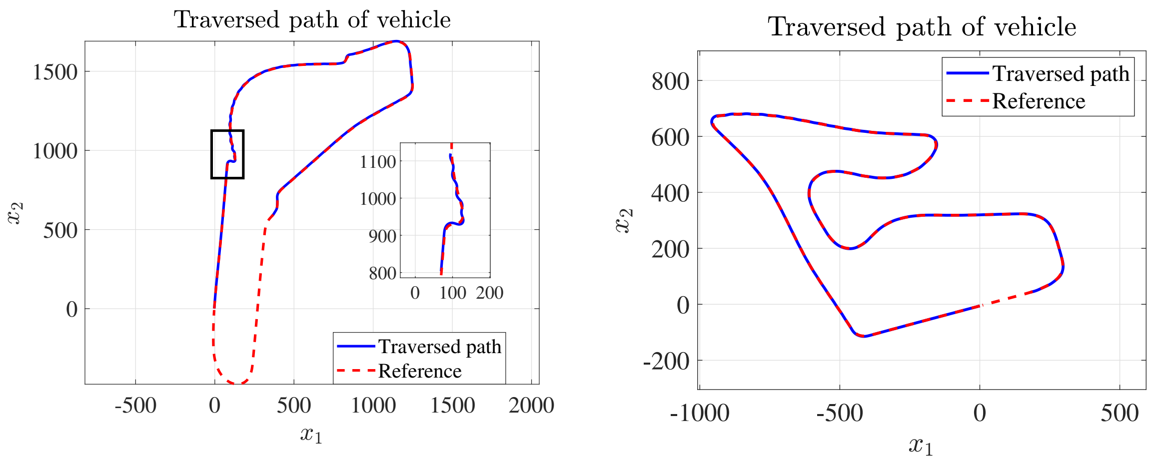

Figure 2. Closed loop simulation of car around two F1 tracks (25 minutes at 10 $\frac{km}{hr}$)

Main objectives of this project are **tracking a reference racing line** while meeting **the track limit constraints**. Upon successful completion of the mentioned performance, making the controller **robust to wind force** is also achievable. The controller hyper-parameters and design should be such that the simulated car can readily does a lap around a well known track.

Attachments

Matlab Simulation files are uploaded to Canvas as a single zip file.

References

[1] J. Kabzan, M. I. Valls, V. J. Reijgwart, H. F. Hendrikx, C. Ehmke, M. Prajapat, A. Bühler, N. Gosala, M. Gupta, R. Sivanesan *et al.*, “Amz driverless: The full autonomous racing system,” *Journal of Field Robotics*, vol. 37, no. 7, pp. 1267–1294, 2020.

[2] J. V. Carrau, A. Liniger, X. Zhang, and J. Lygeros, “Efficient implementation of randomized mpc for miniature race cars,” in *2016 European Control Conference (ECC)*, 2016, pp. 957–962.

[3] U. Rosolia and F. Borrelli, “Learning how to autonomously race a car: a predictive control approach,” *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2713–2719, 2019.

[4] M. Rokonuzzaman, N. Mohajer, S. Nahavandi, and S. Mohamed, “Review and performance evaluation of path tracking controllers of autonomous vehicles,” *IET Intelligent Transport Systems*, vol. 15, no. 5, pp. 646–670, 2021.

[5] H. Wang, B. Liu, X. Ping, and Q. An, “Path tracking control for autonomous vehicles based on an improved mpc,” *IEEE Access*, vol. 7, pp. 161 064–161 073, 2019.

[6] E. Alcalá, V. Puig, J. Quevedo, and U. Rosolia, “Autonomous racing using linear parameter varying-model predictive control (lpv-mpc),” *Control Engineering Practice*, vol. 95, p. 104270, 2020.

[7] K. Iagnemma and M. Buehler, “Editorial for journal of field robotics—special issue on the darpa grand challenge,” pp. 655–656, 2006.

[8] P. Karle, F. Fent, S. Huch, F. Sauerbeck, and M. Lienkamp, “Multi-modal sensor fusion and object tracking for autonomous racing,” *IEEE Transactions on Intelligent Vehicles*, 2023.

[9] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, “Predictive active steering control for autonomous vehicle systems,” *IEEE Transactions on control systems technology*, vol. 15, no. 3, pp. 566–580, 2007.

[10] J. Betz, T. Betz, F. Fent, M. Geisslinger, A. Heilmeier, L. Hermansdorfer, T. Herrmann, S. Huch, P. Karle, M. Lienkamp *et al.*, “Tum autonomous motorsport: An autonomous racing software for the indy autonomous challenge,” *Journal of Field Robotics*, vol. 40, no. 4, pp. 783–809, 2023.

[11] J. Zeng, B. Zhang, and K. Sreenath, “Safety-critical model predictive control with discrete-time control barrier function,” in *2021 American Control Conference (ACC)*. IEEE, 2021, pp. 3882–3889.

[12] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[13] H. B. Pacejka and E. Bakker, “The magic formula tyre model,” *Vehicle system dynamics*, vol. 21, no. S1, pp. 1–18, 1992.

[14] P. Charles, “Race track data base,” https://github.com/TUMFTM/racetrack-database, 2020.