

Methods and Tools for Industrial Automation

First Project

M. Cardano - M. Cardellini - S. Lavaggi

June 5, 2020

1 Goal of the project

In this project we were asked to build a controller able to adjust the movement of an autonomous vehicle in a way to reduce the error on the path to follow. Our goal was not to plan a path that the vehicle should follow, but instead, given a path, to build a controller able to bring the vehicle on the path.

2 Structure of the paper

In Section 3 we give a small review of the models used to simulate a vehicle. We will start with a simple model and then build into one more complicated but that allows to better track the movement of the vehicle. The models presented are all taken from [1] which Prof. Sacile kindly assigned to us. In Section 4 we give an explanation of our controller and the code implemented in Matlab in order to achieve the result.

3 Models for Vehicle Simulation

3.1 Geometric Bicycle Model

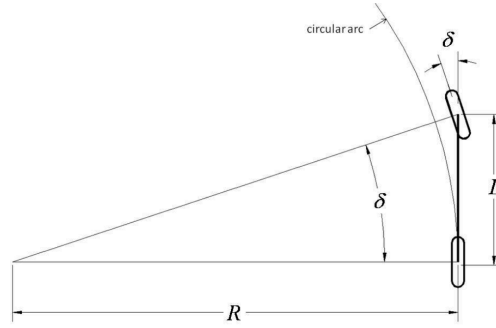


Figure 1: Geometric Bicycle Model

A common simplification of a vehicle is the bicycle model which simplifies the four wheel car by combining the two front wheels together and the two rear wheels together to form a two wheeled model, like a bicycle. The second simplification is that the vehicle can only move on a plane. Figure 1 illustrates the model and the variables used to model it.

The simple geometric relationship between the steering angle and the curvature that the rear axle will follow is

$$\tan(\delta) = \frac{L}{R} \quad (1)$$

where δ is the steering angle of the front wheel, L is the distance between the front axle and the rear axle and R is the radius of the circle that the rear axle will travel along at the given steering angle.

3.2 Dynamic Bicycle Model

It is clear that neglecting vehicle dynamics in the previous models has a negative impact on tracking performance as speeds are increased and path curvature varies. The dynamics of a car are very complicated and high fidelity models are very non-linear, discontinuous and computationally expensive.

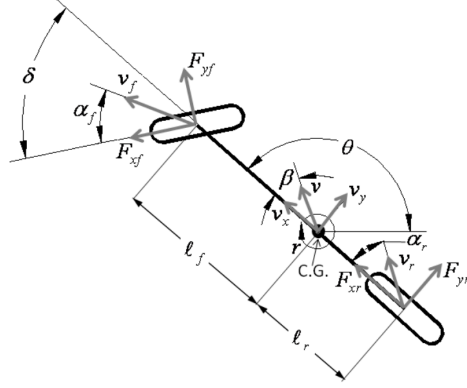


Figure 2: Dynamic Bicycle Model

The bicycle model introduced in the last section can be extended to include dynamics. Lateral forces on the vehicle are of primary concern as path tracking is an exercise in lateral control. For this section, longitudinal velocity is assumed to be controlled separately. Summing the lateral forces illustrated in Figure 4, given vehicle mass m , yields

$$F_{yf}\cos(\delta) - F_{xf}\sin(\delta) + F_{yr} = m(\dot{v}_y + v_x r) \quad (2)$$

Considering only motion in the plane, a center of gravity C.G. along the center line of the vehicle, and yaw inertia I_z , balancing the yaw moments gives

$$\ell_f(F_{yf}\cos(\delta)) - \ell_r(F_{yr} - F_{xf}\sin(\delta)) = I_z \dot{r} \quad (3)$$

where r is the angular rate about the yaw axis. The slip angles of the tires are given as

$$\alpha_f = \tan^{-1} \left(\frac{v_y + \ell_f r}{v_x} \right) - \delta \quad (4)$$

$$\alpha_r = \tan^{-1} \left(\frac{v_y - \ell_r r}{v_x} \right) \quad (5)$$

Modelling the force generated by the wheels as linearly proportional to the slip angle, the lateral forces are defined as

$$F_{yf} = -c_f \alpha_f \quad (6)$$

$$F_{yr} = -c_r \alpha_r \quad (7)$$

Where c_f and c_r are the proportional constants between the forces and the slip angle. Assuming a constant longitudinal velocity $\dot{v}_x = 0$, allows the simplification

$$F_{xf} = 0 \quad (8)$$

Non Linear Model Substituting Eqs. 7 and 8 into Eqs. 2 and 3 and solving for \dot{v}_y and \dot{r}

$$\dot{v}_y = \frac{-c_f \left[\tan^{-1} \left(\frac{v_y + \ell_f r}{v_x} \right) - \delta \right] \cos(\delta) - c_r \tan^{-1} \left(\frac{v_y - \ell_r r}{v_x} \right)}{m} - v_x r \quad (9)$$

$$\dot{r} = \frac{-\ell_f c_f \left[\tan^{-1} \left(\frac{v_y + \ell_f r}{v_x} \right) - \delta \right] \cos(\delta) + \ell_r c_r \tan^{-1} \left(\frac{v_y - \ell_r r}{v_x} \right)}{I_z} \quad (10)$$

gives the dynamic bicycle model.

Linearized Model To apply linear control methods to the dynamic bicycle model, the model must be linearized. Applying small angle assumption to eqs 9 and 10 and writing it in state space form gives

$$\begin{bmatrix} \dot{v}_y \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{-(c_f + c_r)}{m v_x} & \frac{\ell_r c_r - \ell_f c_f}{m v_x} - v_x \\ \frac{\ell_r c_r - \ell_f c_f}{I_z v_x} & \frac{-(\ell_f^2 c_f + \ell_r^2 c_r)}{I_z v_x} \end{bmatrix} \begin{bmatrix} v_x \\ r \end{bmatrix} + \begin{bmatrix} \frac{c_f}{m} \\ \frac{\ell_f c_f}{m} \end{bmatrix} \delta \quad (11)$$

Path coordinates It is useful to express the dynamic bicycle model with respect to the path.

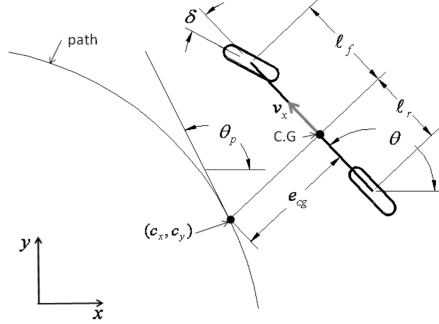


Figure 3: Dynamic Bicycle Model in path coordinates

Defining \dot{e}_{cg} as the velocity of drift from the path and $\theta_e = \theta - \theta_p$ as the error of the angle and knowing that

$$\dot{e}_{cg} = v_y + v_x \sin(\theta_e) \quad (12)$$

we can write the state space model in tracking error variables as

$$\begin{bmatrix} \dot{e}_{cg} \\ \ddot{e}_{cg} \\ \dot{\theta}_e \\ \ddot{\theta}_e \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-(c_f+c_r)}{mv_x} & \frac{c_f+c_r}{m} & \frac{\ell_f c_f - \ell_r c_r}{mv_x} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{\ell_r c_r - \ell_f c_f}{I_z v_x} & \frac{\ell_f c_f - \ell_r c_r}{I_z} & \frac{-(\ell_f^2 c_f + \ell_r^2 c_r)}{I_z v_x} \end{bmatrix} \begin{bmatrix} e_{cg} \\ \dot{e}_{cg} \\ \theta_e \\ \dot{\theta}_e \end{bmatrix} \quad (13)$$

$$+ \begin{bmatrix} 0 \\ \frac{c_f}{m} \\ 0 \\ \frac{\ell_f c_f}{I_z} \end{bmatrix} \delta + \begin{bmatrix} 0 \\ \frac{\ell_r c_r - \ell_f c_f}{mv_x} - v_x \\ 0 \\ \frac{-(\ell_f^2 c_f + \ell_r^2 c_r)}{I_z v_x} \end{bmatrix} r(s) \quad (14)$$

4 Linear Quadratic Regulator

Parameters. The paper gave us all the parameters based on experiments:

$$\begin{aligned} m &= 1140 \\ I_z &= 1436.24 \\ \ell_f &= 1.165 \\ \ell_r &= 1.165 \\ c_f &= 155494.663 \\ c_r &= 155494.663 \\ v_x &= 1.1765 \end{aligned}$$

The state space The state matrix A becomes:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -231.8722 & 272.7977 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -249.7919 \end{bmatrix} \quad (15)$$

The eigenvalues of the matrix A are:

$$\text{eig}(A) = [0 \quad -231.8722 \quad 0 \quad -249.7919]^T \quad (16)$$

and so the system is not stable because of the two eigenvalues in zero.

Riccati's Equation. To control the system we used an LQR. To minimize the cost function

$$J = \sum_{\tau=0}^{N-1} (x_{\tau}^T Q_{\tau} x_{\tau} + u_{\tau}^T R_{\tau} u_{\tau}) + x_N^T Q_f x_N$$

that brings the error to zero by using the least possible control we used the Riccati's equation with the following parameters (as specified in the paper):

$$Q = \begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$R = 1$$

$$Q_f = Q$$

In this way we were able find the time-dependent matrix $K(\tau)$ to bring the tracking error to zero independently on the initial conditions:

$$\begin{aligned} u_{\tau} &= -K(\tau)x_{\tau} \\ x_{\tau+1} &= A_d x_{\tau} + B_d u_{\tau}; \end{aligned}$$

For example the following initial condition

$$x_0 = \begin{bmatrix} e_{cg} \\ \dot{e}_{cg} \\ \theta_e \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} 2 \\ 6.25 \\ -\frac{\pi}{2} \\ 0 \end{bmatrix}$$

denotes a vehicle that is currently distant $2m$ from the path, moving away with a velocity of $6.25m/s$ perpendicular with the path.

We were able to bring the error to zero as seen in the following figure

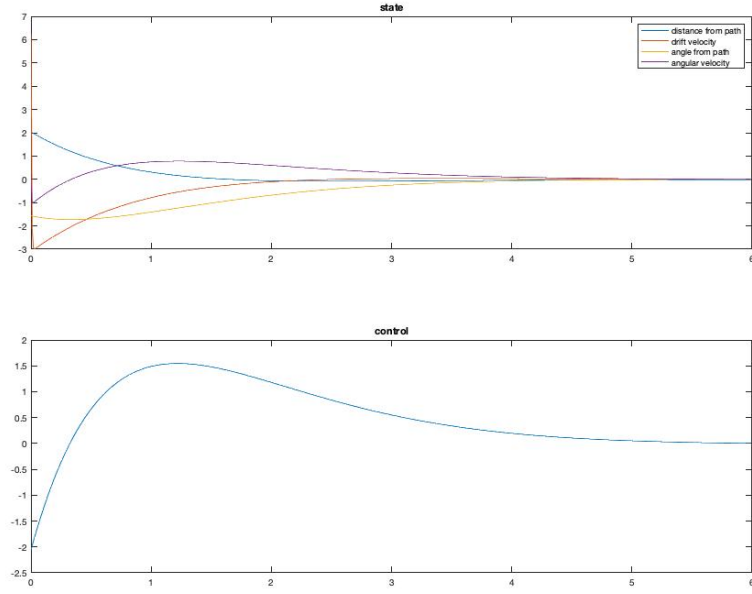


Figure 4: State response to the proportional control $K(\tau)$

References

- [1] J. M. Snider. *Automatic Steering Methods for Autonomous Automobile Path Tracking*. 02 2009.