

EL2425 - Slip Control

Meeting agenda 2016-11-16

November 22, 2016

1 Done

- Installation of the optimization library in python needed for MPC. However, there are issues with the statement of the problem in terms of the language itself. This stating “does not conform to DCP rules”.

2 Ongoing

2.1 Tracking the centerline of a lane using a PID controller

The problem can be decomposed into two separate and independent components involving a translation and a rotation of the vehicle.

Translational component We assume that the pose of the vehicle at time t is (x_c, y_c, v_c, ψ_v) and that two range scans at $+90^\circ$ and -90° with respect to the longitudinal axis of the vehicle are available, which are denoted as CL and CR respectively. Also, in figure 1, point O' is a future reference and the angle λ is the angle that the vehicle should make in order to reach O' in a future moment.

Then, since $CL + CR = L = OL + OR$ and $CL = OC + OL$, this means that

$$OC = OR - CR = \frac{L}{2} - CR = \frac{CL + CR}{2} - CR = \frac{CL - CR}{2} \quad (1)$$

where L is the width of the lane whose centerline the vehicle is to track.

Furthermore, in the $CC'O'$ triangle

$$\tan \lambda = \frac{O'C'}{CC'} = \frac{OC}{CC'} = \frac{CL - CR}{2CC'} \quad (2)$$

$$\lambda = \tan^{-1} \frac{CL - CR}{2CC'} \quad (3)$$

The length CC' is unknown and can be set beforehand. Its magnitude will determine the vehicle's rate of convergence to the centerline of the lane.

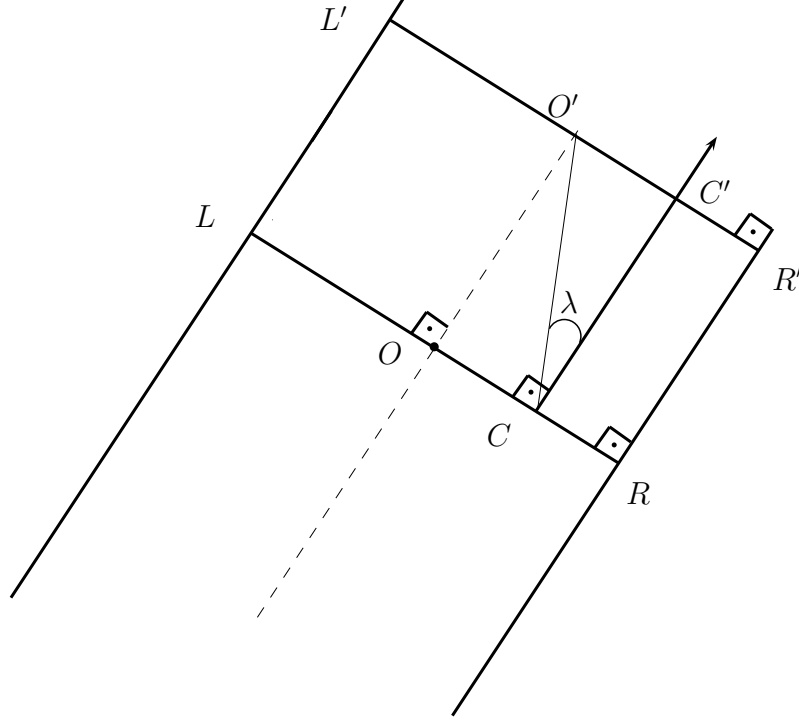


Figure 1: The vehicle's heading angle is equal to that of the lane's. However, its position is off track.

The angular error of the translational component is then

$$e_t = -\tan^{-1} \frac{CL - CR}{2CC'} \quad (4)$$

where the minus sign is introduced due to the convention that a left turn is assigned a negative value.

Rotational component

We assume that the the pose of the vehicle at time t as (x_c, y_c, v_c, ψ_v) and that three range scans at $+90^\circ$, 0° and -90° with respect to the longitudinal axis of the vehicle are available, which are denoted as CL , CF and CR respectively.

Here we can distinguish two cases: one where the vehicle is facing the right lane boundary and one where it is facing the left one. It is not obvious in this configuration where the vehicle is heading: the only available information so far is only the three range scans.

In the first case, the heading angle error is

$$\phi = \tan^{-1} \frac{CR}{CF} \quad (5)$$

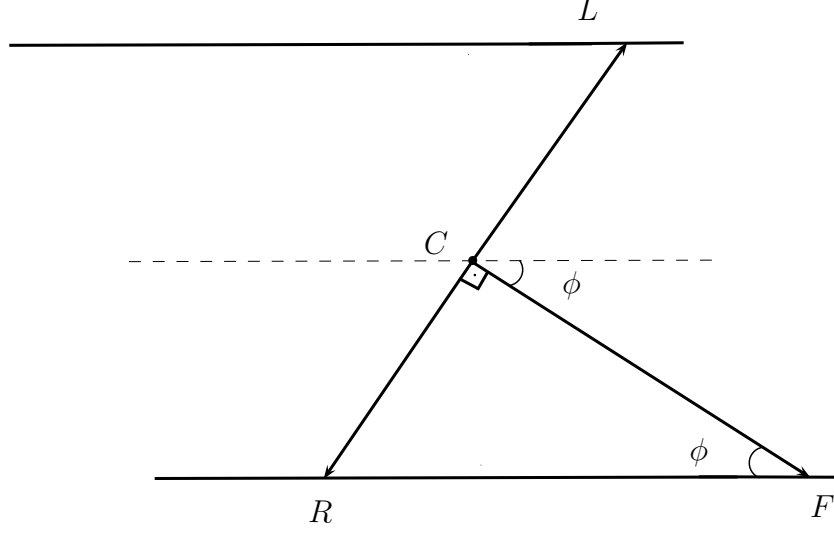


Figure 2: The vehicle's position and its reference are equal. However, the vehicle's heading is off track. Its heading is towards the right lane boundary.

The rotational error of the vehicle in this case is

$$e_r = -\tan^{-1} \frac{CR}{CF} \quad (6)$$

therefore the overall angular error of the vehicle in this case is

$$e = -\tan^{-1} \frac{CL - CR}{2CC'} - \tan^{-1} \frac{CR}{CF} \quad (7)$$

where the minus signs are introduced due to the convention that a left turn is assigned a negative value.

In the second case, the heading angle error is

$$\phi = \tan^{-1} \frac{CL}{CF} \quad (8)$$

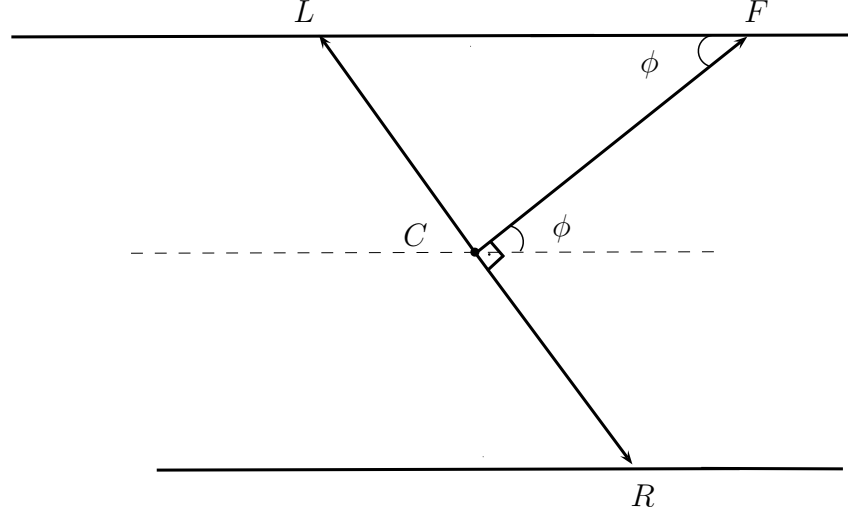


Figure 3: The vehicle's position and its reference are equal. However, the vehicle's heading is off track. Its heading is towards the left lane boundary.

The rotational error of the vehicle in this case is

$$e_r = \tan^{-1} \frac{CL}{CF} \quad (9)$$

therefore the overall angular error of the vehicle in this case is

$$e = -\tan^{-1} \frac{CL - CR}{2CC'} + \tan^{-1} \frac{CL}{CF} \quad (10)$$

where the minus sign of the translational error is introduced due to the convention that a left turn is assigned a negative value.

In order to deduce the correct value of ϕ ($-\tan^{-1} \frac{CR}{CF}$ or $\tan^{-1} \frac{CL}{CF}$) further ranges scans are needed. To this end, a difference between ranges around point F is taken: starting at the right of F and moving anti-clockwise, we calculate the difference between two range scans for a given angle between them. If its sign is negative then the vehicle is facing the right lane boundary; if not, it is facing the left lane boundary. This concept is depicted in figures 4 and 5.

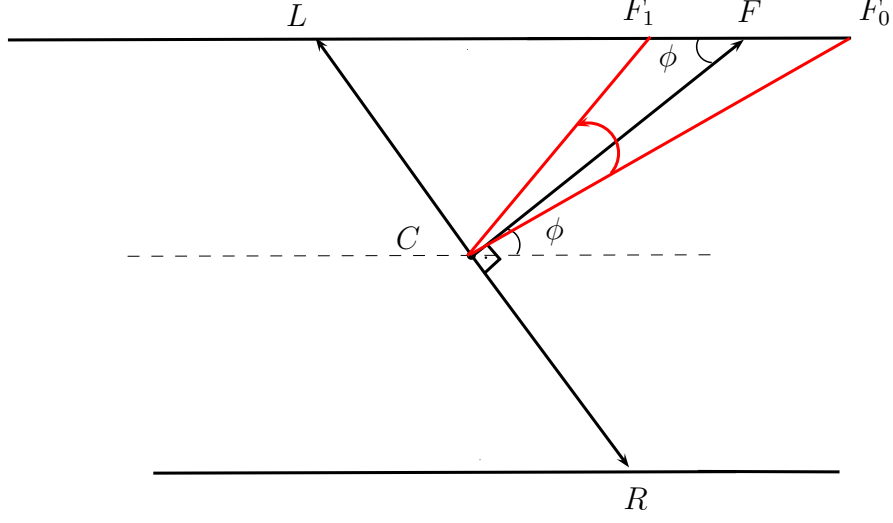


Figure 4: $CF_0 > CF > CF_1$, hence $CF_0 - CF_1 > 0$ and $\phi = \tan^{-1} \frac{CL}{CF}$

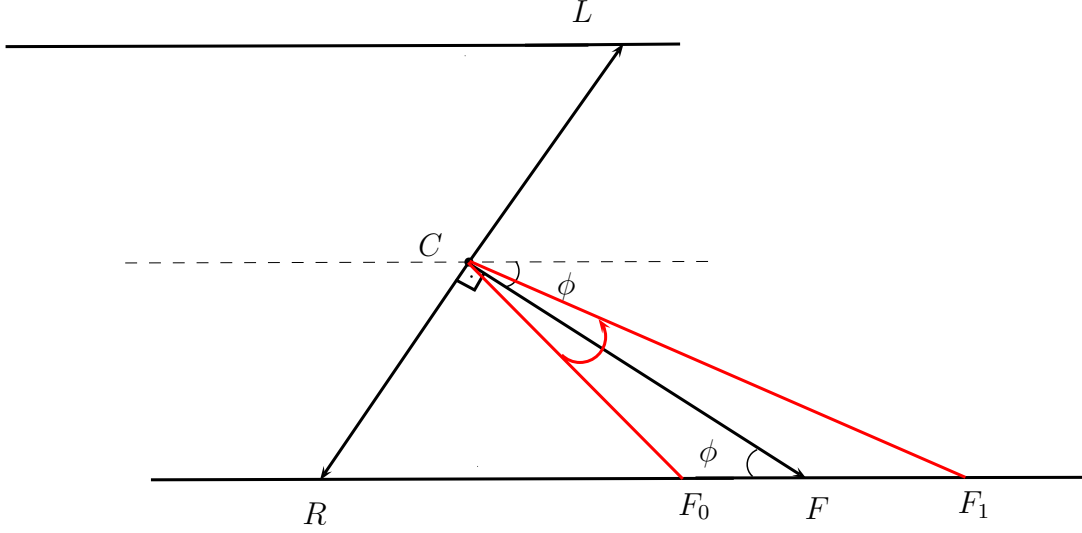


Figure 5: $CF_0 < CF < CF_1$, hence $CF_0 - CF_1 < 0$ and $\phi = -\tan^{-1} \frac{CR}{CF}$

Since the input to the plant is in terms of angular displacement, this is in fact the error that the PID controller can include and utilize in order to determine the plant's input.

The angular input to the vehicle will then be

$$\delta = K_p \cdot e + K_d \cdot \frac{de}{dt} + K_i \cdot \int e dt \quad (11)$$

2.2 Tracking the centerline of a lane using a MPC controller

In figure 6, the x axis is fixed on the lane's centerline, while axis y is perpendicular to it. The origin is at point O . The vehicle is represented by point C . The orientation of the

vehicle with respect to the lane (the x -axis) is ϕ . Given this configuration and three scans, at -90° , 0° and 90° with respect to the longitudinal axis of the vehicle, denoted by CR , CF and CL respectively, the objective is to find the distance OC and the angle ϕ so that a MPC optimization problem can be solved with OC and ϕ acting as initial conditions. The velocity of the vehicle, which is constant, and its displacement along the x -axis are at this point irrelevant to the optimization problem given this configuration. The only source of information is the lidar itself.

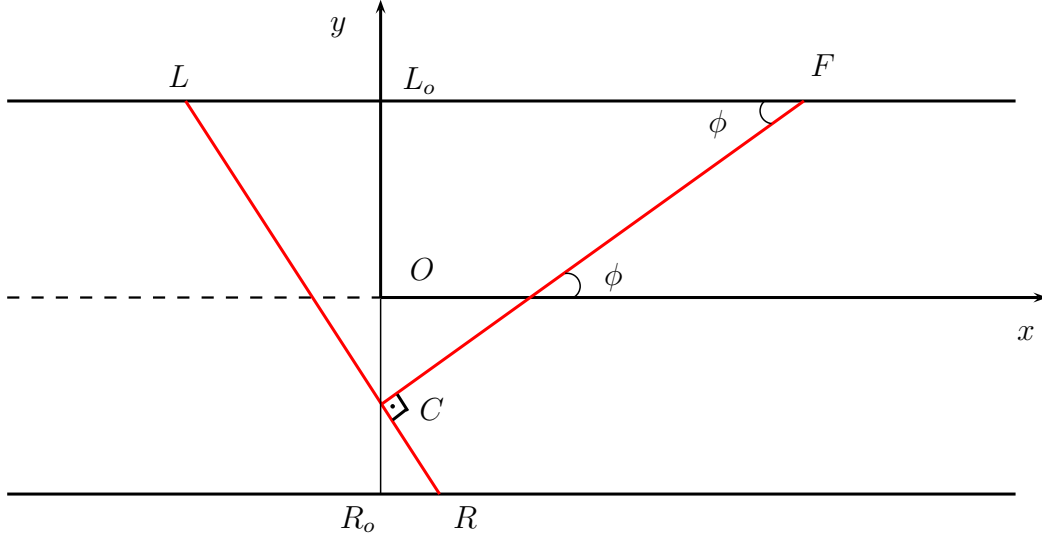


Figure 6

Initial conditions

Finding ϕ

Here we can distinguish two cases: one where the vehicle is facing the right lane boundary and one where it is facing the left one. It is not obvious in this configuration where the vehicle is heading: the only available information so far is only the three range scans.

In the first case, the heading angle error is

$$\phi = \tan^{-1} \frac{CR}{CF} \quad (12)$$

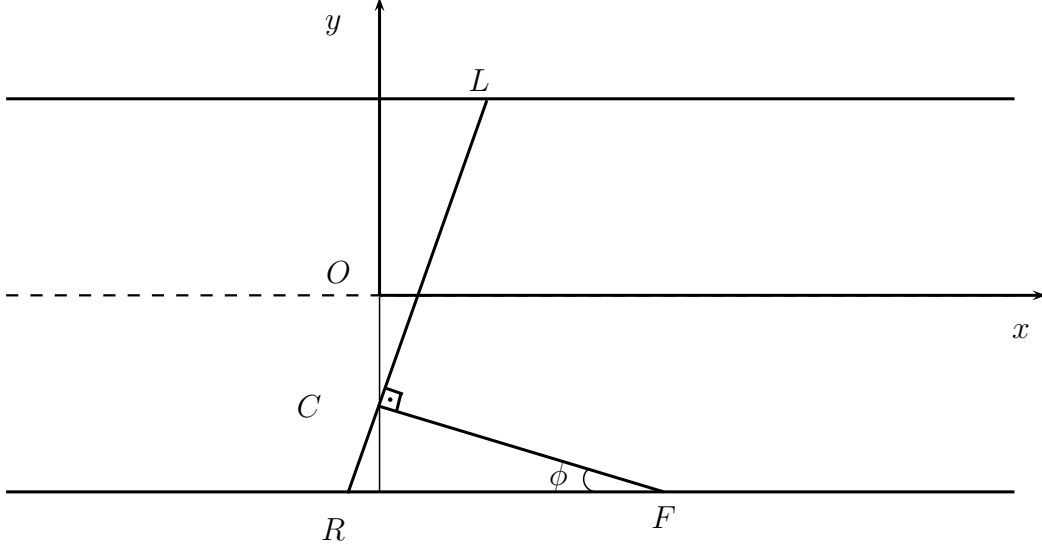


Figure 7: The vehicle's heading is towards the right lane boundary.

In the second case, the heading angle error is

$$\phi = \tan^{-1} \frac{CL}{CF} \quad (13)$$

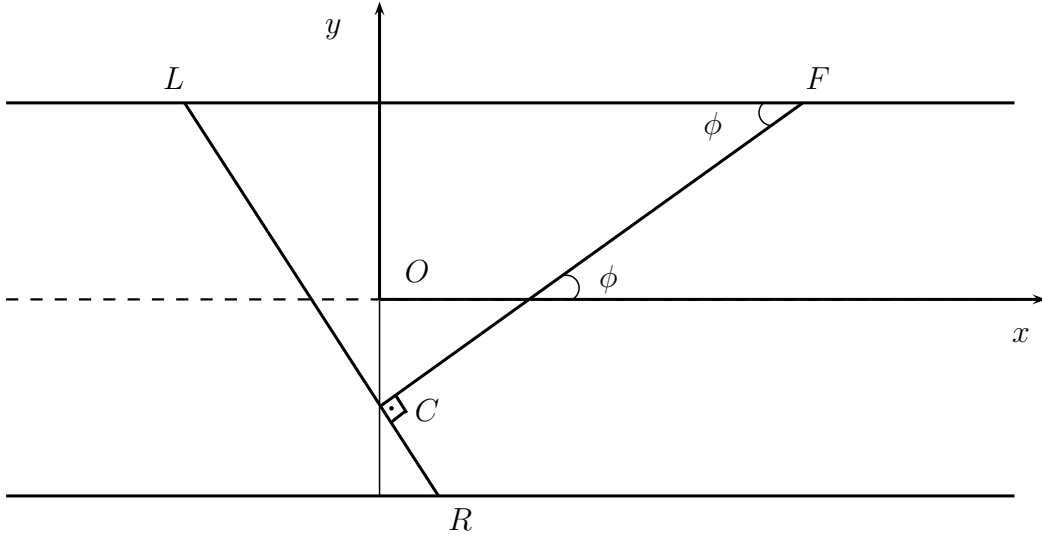


Figure 8: The vehicle's heading is towards the left lane boundary.

In order to deduce the correct value of ϕ ($-\tan^{-1} \frac{CR}{CF}$ or $\tan^{-1} \frac{CL}{CF}$) further ranges scans are needed. To this end, a difference between ranges around point F is taken: starting at the right of F and moving anti-clockwise, we calculate the difference between two range scans for a given angle between them. If its sign is negative then the vehicle is facing the right

lane boundary; if not, it is facing the left lane boundary. This concept is depicted in figures 9 and 10.

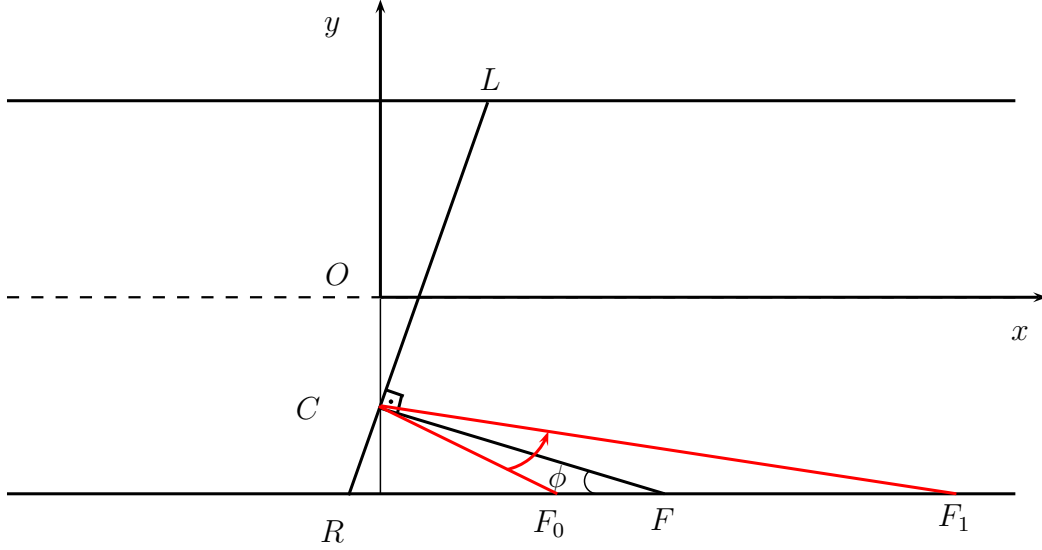


Figure 9: $CF_0 < CF < CF_1$, hence $CF_0 - CF_1 < 0$ and $\phi = -\tan^{-1}\frac{CR}{CF}$

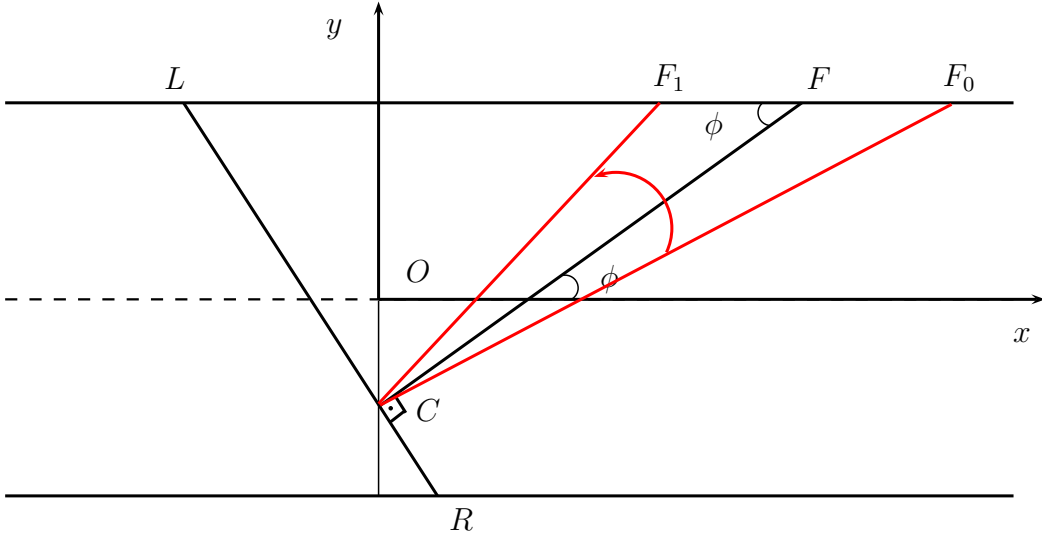


Figure 10: $CF_0 > CF > CF_1$, hence $CF_0 - CF_1 > 0$ and $\phi = \tan^{-1}\frac{CL}{CF}$

Finding OC

First we note that OC does not depend on the orientation of the vehicle. Then

$$L_oO + OC + CR_o = W = CR_o + CL_o \quad (14)$$

where W is the width of the lane. But $L_oO = \frac{W}{2}$ hence

$$OC + CR_o = \frac{1}{2}(CR_o + CL_o) \Leftrightarrow \quad (15)$$

$$OC = \frac{1}{2}(CL_o - CR_o) \quad (16)$$

But $CL_o = CL \sin \lambda$ and $CR_o = CR \cos \phi$, hence

$$OC = \frac{1}{2}(CL \sin \lambda - CR \cos \phi) \quad (17)$$

From triangle LCF in the case where the vehicle is facing the left lane boundary, we note that $\phi + \lambda + \frac{\pi}{2} = \pi$, hence $\lambda = \frac{\pi}{2} - \phi$. Then, we conclude that

$$OC = \frac{1}{2}(CL - CR) \cos \phi \quad (18)$$

Obtaining the relevant linearized kinematic model

The model constitutes the equations of motion of the vehicle, and has two states (y and ϕ) and one input (δ). The equations of the vehicle's motion that are relevant here are

$$\dot{y} = v \sin(\phi + \beta) \quad (19)$$

$$\dot{\phi} = \frac{v}{l_r} \sin \beta \quad (20)$$

Sampling with a sampling time of T_s gives

$$y_{k+1} = y_k + T_s v \sin(\phi_k + \beta_k) \quad (21)$$

$$\phi_{k+1} = \phi_k + T_s \frac{v}{l_r} \sin \beta_k \quad (22)$$

where

$$\beta_k = \tan^{-1} \left(\frac{l_r}{l_r + l_f} \tan \delta_k \right) \quad (23)$$

Forming the Jacobians for matrices A , B and evaluating them at time $t = k$ around $\delta = 0$ (which makes $\beta = 0$) gives

$$A = \begin{bmatrix} 1 & T_s v \cos(\phi_k + \beta_k) \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & T_s v \cos \left(\phi_k + \tan^{-1}(l_q \tan \delta_k) \right) \\ 0 & 1 \end{bmatrix} \quad (24)$$

$$A = \begin{bmatrix} 1 & T_s v \cos \phi_k \\ 0 & 1 \end{bmatrix} \quad (25)$$

$$B = \begin{bmatrix} T_s v \cos(\phi_k + \beta_k) \frac{l_q}{l_q^2 \sin^2 \delta_k + \cos^2 \delta_k} \\ \frac{T_s v}{l_r} \cos(\beta_k) \frac{l_q}{l_q^2 \sin^2 \delta_k + \cos^2 \delta_k} \end{bmatrix} = \begin{bmatrix} T_s v \cos\left(\phi_k + \tan^{-1}(l_q \tan \delta_k)\right) \frac{l_q}{l_q^2 \sin^2 \delta_k + \cos^2 \delta_k} \\ \frac{T_s v}{l_r} \cos\left(\tan^{-1}(l_q \tan \delta_k)\right) \frac{l_q}{l_q^2 \sin^2 \delta_k + \cos^2 \delta_k} \end{bmatrix} \quad (26)$$

$$B = \begin{bmatrix} \frac{T_s l_r v}{l_r + l_f} \cos \phi_k \\ \frac{T_s v}{l_r + l_f} \end{bmatrix} \quad (27)$$

where $l_q = \frac{l_r}{l_r + l_f}$

Now we can express the linear model as

$$s_{k+1} = A s_k + B \delta_k \quad (28)$$

where

$$s = \begin{bmatrix} y_k \\ \phi_k \end{bmatrix} \quad (29)$$

or

$$\begin{bmatrix} y_{k+1} \\ \phi_{k+1} \end{bmatrix} = \begin{bmatrix} 1 & T_s v \cos \phi_k \\ 0 & 1 \end{bmatrix} \begin{bmatrix} y_k \\ \phi_k \end{bmatrix} + \begin{bmatrix} \frac{T_s l_r v}{l_r + l_f} \cos \phi_k \\ \frac{T_s v}{l_r + l_f} \end{bmatrix} \delta_k \quad (30)$$

Stating the optimization problem

We can now form the optimization problem as

$$\text{minimize } \sum_{k=0}^N y_k^2 q_y + \phi_k^2 q_\phi + \delta_k^2 r \quad (31)$$

$$\text{subject to } s_{k+1} = A s_k + B \delta_k, \text{ where } s_k = [y_k, \phi_k]^T \quad (32)$$

$$\delta_{\min} \leq \delta_k \leq \delta_{\max} \quad (33)$$

$$y_0 = -\frac{1}{2}(CL - CR) \cos \phi_0 \quad (34)$$

$$\phi_0 = \tan^{-1} \frac{CL}{CF} \text{ or } \phi_0 = -\tan^{-1} \frac{CR}{CF} \quad (35)$$

2.3 Tracking the circumference of a circle using a MPC controller

2.3.1 Theory

In figure 11, the vehicle C , whose velocity is constant and denoted by v , is to track a circle whose center is O' and whose radius is $O'R$. Its orientation relative to the global coordinate system is ψ . $R(x_R, y_R)$ is the point C is to track. The orientation of R relative to the global coordinate system is ψ_R . The vehicle's coordinates are (x_c, y_c) .

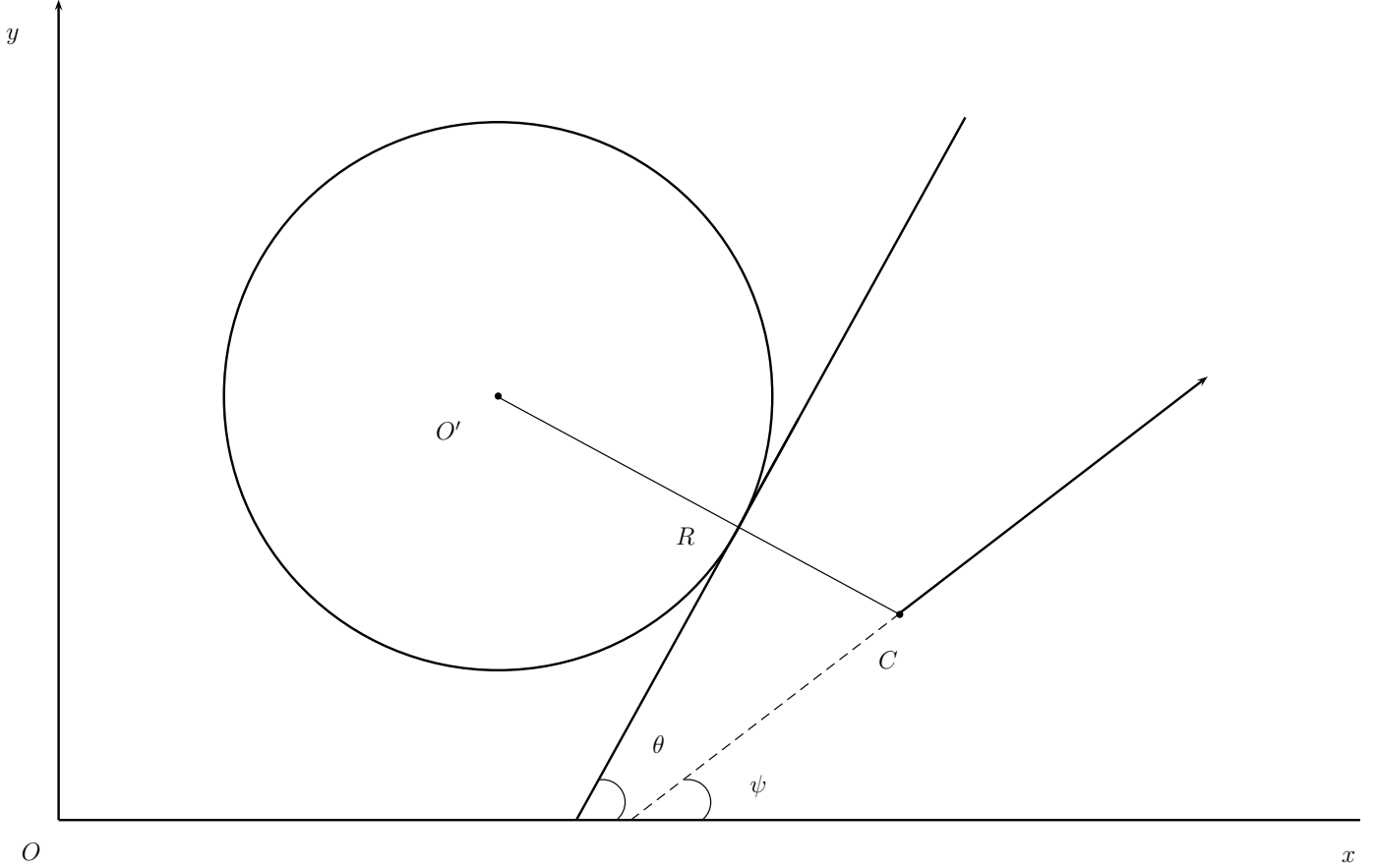


Figure 11

R is a moving reference point and its pose is found as follows. Given C and the trajectory of the circle, point R is the point with the least distance to C among all points of the trajectory. Since the circle is known a priori and comprised by a set of poses, R is known.

The aim is for the vehicle C to minimize its deviation from R , in other words we want to drive the differences

$$\begin{bmatrix} x_c - x_R \\ y_c - y_R \\ v_c - v_R \\ \psi_c - \psi_R \end{bmatrix} \rightarrow 0 \quad (36)$$

where v_c and v_R are the velocities of the vehicle and the reference point, respectively.

Obtaining the linearized kinematic model

The model constitutes the equations of motion of the vehicle, and has four states (x , y , v and ψ) and two inputs (v_i and δ). The equations of the vehicle's motion that are relevant here are

$$\dot{x} = v \cos(\psi + \beta) \quad (37)$$

$$\dot{y} = v \sin(\psi + \beta) \quad (38)$$

$$\dot{v} = \frac{v_i - v}{\tau} \quad (39)$$

$$\dot{\psi} = \frac{v}{l_r} \sin \beta \quad (40)$$

Sampling with a sampling time of T_s gives

$$x_{k+1} = x_k + T_s v \cos(\psi_k + \beta_k) \quad (41)$$

$$y_{k+1} = y_k + T_s v \sin(\psi_k + \beta_k) \quad (42)$$

$$v_{k+1} = v_k + \frac{T_s}{\tau} (v_{i,k} - v_k) \quad (43)$$

$$\psi_{k+1} = \psi_k + T_s \frac{v}{l_r} \sin \beta_k \quad (44)$$

where

$$\beta_k = \tan^{-1} \left(\frac{l_r}{l_r + l_f} \tan \delta_{k-1} \right) \quad (45)$$

Forming the Jacobians for matrices A , B and evaluating them at time $t = k$ around $\psi = \psi_R$, $v = v_R$ and $\delta = \delta_{k-1}$ (δ_k is to be determined at time k):

$$A = \begin{bmatrix} 1 & 0 & T_s \cos(\psi_R + \beta_k) & -T_s v_R \sin(\psi_R + \beta_k) \\ 0 & 1 & T_s \sin(\psi_R + \beta_k) & T_s v_R \cos(\psi_R + \beta_k) \\ 0 & 0 & 1 - \frac{T_s}{\tau} & 0 \\ 0 & 0 & \frac{T_s}{l_r} \sin(\beta_k) & 1 \end{bmatrix} \quad (46)$$

$$A = \begin{bmatrix} 1 & 0 & T_s \cos\left(\psi_R + \tan^{-1}(l_q \tan \delta_{k-1})\right) & -T_s v_k \sin\left(\psi_R + \tan^{-1}(l_q \tan \delta_{k-1})\right) \\ 0 & 1 & T_s \sin\left(\psi_R + \tan^{-1}(l_q \tan \delta_{k-1})\right) & T_s v_k \cos\left(\psi_R + \tan^{-1}(l_q \tan \delta_{k-1})\right) \\ 0 & 0 & 1 - \frac{T_s}{\tau} & 0 \\ 0 & 0 & \frac{T_s}{l_r} \sin(\tan^{-1}(l_q \tan \delta_{k-1})) & 1 \end{bmatrix} \quad (47)$$

$$B = \begin{bmatrix} 0 & -T_s v_R \sin(\psi_R + \beta_k) \frac{l_q}{l_q^2 \sin^2 \delta_{k-1} + \cos^2 \delta_{k-1}} \\ 0 & T_s v_R \cos(\psi_R + \beta_k) \frac{l_q}{l_q^2 \sin^2 \delta_{k-1} + \cos^2 \delta_{k-1}} \\ \frac{T_s}{\tau} v_R & 0 \\ 0 & \frac{T_s v}{l_r} \cos(\beta_k) \frac{l_q}{l_q^2 \sin^2 \delta_{k-1} + \cos^2 \delta_{k-1}} \end{bmatrix} \quad (48)$$

$$B = \begin{bmatrix} 0 & -T_s v_R \sin\left(\psi_R + \tan^{-1}(l_q \tan \delta_{k-1})\right) \frac{l_q}{l_q^2 \sin^2 \delta_{k-1} + \cos^2 \delta_{k-1}} \\ 0 & T_s v_R \cos\left(\psi_R + \tan^{-1}(l_q \tan \delta_{k-1})\right) \frac{l_q}{l_q^2 \sin^2 \delta_{k-1} + \cos^2 \delta_{k-1}} \\ \frac{T_s}{\tau} v_R & 0 \\ 0 & \frac{T_s v}{l_r} \cos\left(\tan^{-1}(l_q \tan \delta_{k-1})\right) \frac{l_q}{l_q^2 \sin^2 \delta_{k-1} + \cos^2 \delta_{k-1}} \end{bmatrix} \quad (49)$$

where $l_q = \frac{l_r}{l_r + l_f}$

Now we can express the linear model as

$$s_{k+1} = A s_k + B u_k \quad (50)$$

where

$$s_k = \begin{bmatrix} x_k \\ y_k \\ v_k \\ \psi_k \end{bmatrix} \quad (51)$$

and

$$u_k = \begin{bmatrix} v_{i,k} \\ \delta_k \end{bmatrix} \quad (52)$$

Stating the optimization problem

We can now form the optimization problem to be solved at time t as

$$\text{minimize } \sum_{k=0}^N (s_k - s_{ref})^T Q (s_k - s_{ref}) + u_k^T R u_k \quad (53)$$

$$\text{subject to } s_{k+1} = A s_k + B u_k, \text{ where } s_k = [x_k, y_k, v_k, \psi_k]^T, u_k = [v_{i,k}, \delta_k]^T \quad (54)$$

$$u_i^{min} \leq u_{i,k} \leq u_i^{max} \quad (55)$$

$$\delta^{min} \leq \delta_k \leq \delta^{max} \quad (56)$$

$$s_{ref} = (x_R, y_R, v_R, \psi_R) \quad (57)$$

$$s_0 = (x_t, y_t, v_t, \psi_t) \quad (58)$$

2.3.2 Simulation

The prototyping is being done in matlab. Figure 12 shows the reference trajectory in blue and the trajectory of the vehicle in red. Figure 13 shows the reference velocity of the vehicle in blue and the velocity of the vehicle in red.

These results act purely as a proof of concept. Tuning of the weights on the states and the inputs is not an easy task, and it will have to be performed primarily experimentally.

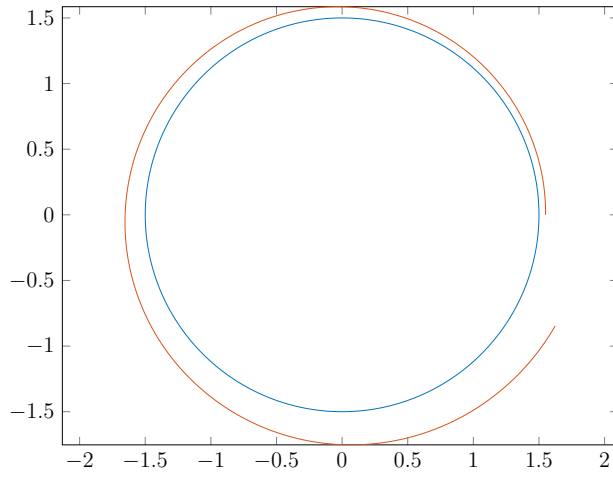


Figure 12

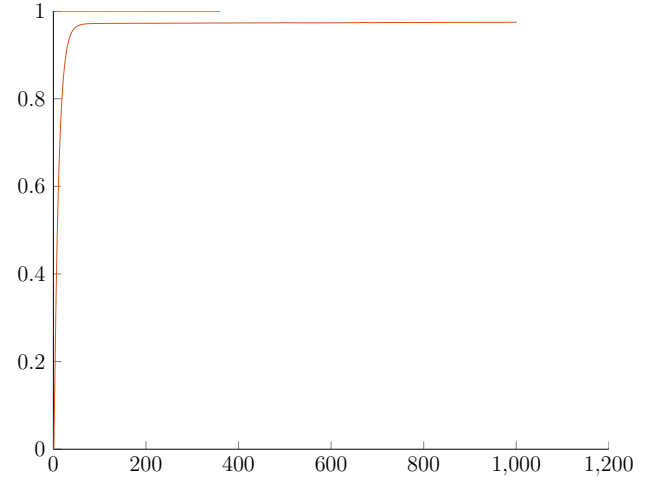


Figure 13

3 Issues

- There might be an issue regarding the execution time of the optimization algorithm on the vehicle.

4 To do

5 Misc.

The progress of the project can be observed in `trello` and `github`:

- <https://trello.com/b/uEP0jl0B/slip-control>
- https://gits-15.sys.kth.se/alefil/HT16_P2_EL2425
- https://gits-15.sys.kth.se/alefil/HT16_P2_EL2425_resources