EL2425 - Slip Control
Meeting agenda 2016-11-16

November 16, 2016

1 Done

- **Microcontroller messaging**. Teensy was eventually flushed with new firmware that allows only messages of type `slip_control_communications/input_drive` to be passed to it. (Previously, a message type of `f1tenth_msgs/drive_values` was hard-coded into Arduino's firmware, making communication within package `slip_control` impossible due to incompatibility.)
- **Time constant** pertaining to the velocity response of the vehicle found, although significant differences between time constants for different velocity references have been found.
- **ROS infrastructure** set.
- **MPC python package** found: `cvxopt`. It seems to be able to capture the essence of what our goal is. See `http://nbviewer.jupyter.org/github/cvxgrp/cvx_short_course/blob/master/intro/control.ipynb`
- Theoretical solution involving PID and traveling on the centerline of a lane found, and code for it written. However, the involved gains have to be tuned experimentally.
- Theoretical solution involving MPC and tracking the centerline of a lane found, however code for it has not been written yet.

2 Ongoing

2.1 Theoretical solution involving PID and tracking the centerline of a lane

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 $+90°$ 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

$$OC = OR - CR = \frac{L}{2} - CR = \frac{CL + CR}{2} - CR = \frac{CL - CR}{2} \tag{1}$$

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

Furthermore, in the $CC'O'$ triangle

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

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

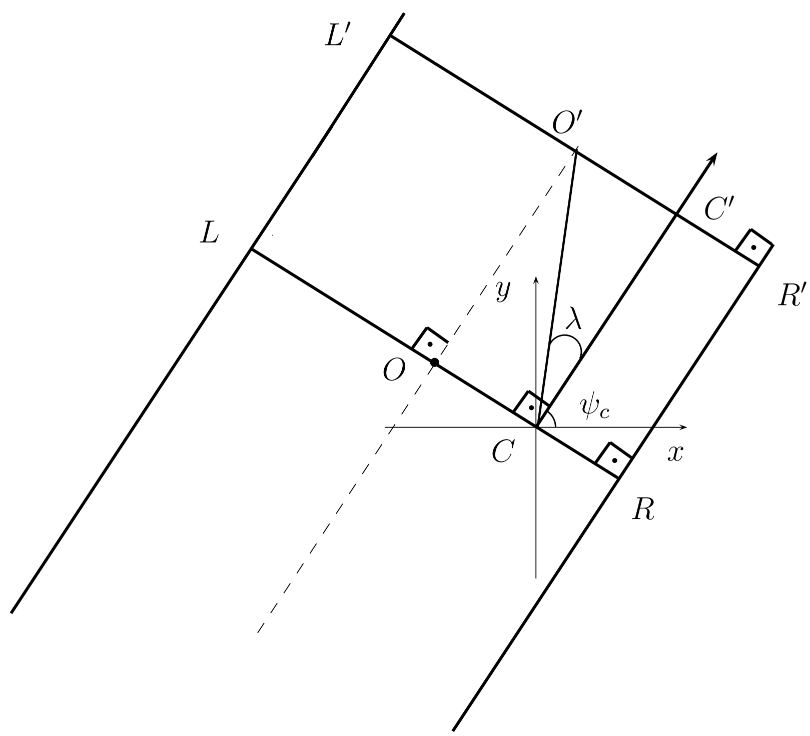

Figure 1: The vehicle's heading angle is equal to that of the lane's. However, its position is off track.

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 $+90°$, $0°$ and $-90°$ with respect to the longitudinal axis of the vehicle are available, which are denoted as CL, CF and CR respectively. Then, the heading angle error is

$$\phi = \frac{\pi}{2} - \tan^{-1}\frac{CF}{CR} \tag{4}$$

since $\mu + \phi + \dfrac{\pi}{2} = \pi$ and $\tan(\mu) = \dfrac{CF}{CR}$.

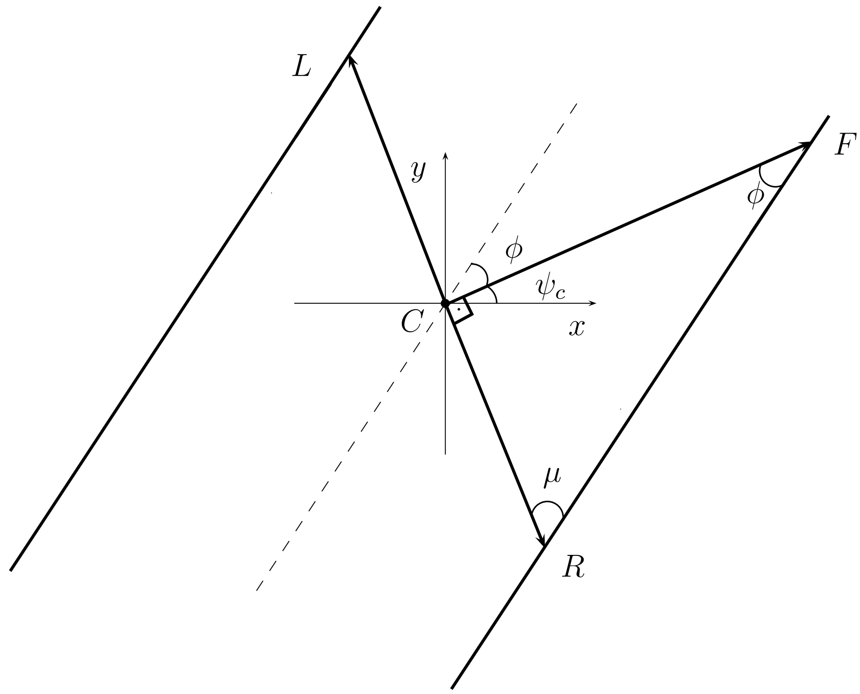

Figure 2: The vehicle's position and its reference are equal. However, the vehicle's heading is off track.

The overall angle error of the vehicle is then

$$e = \tan^{-1}\frac{CL - CR}{2CC'} + \frac{\pi}{2} - \tan^{-1}\frac{CF}{CR} \tag{5}$$

Here, 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.

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

$$\delta = K_p \cdot e + K_d \cdot \frac{de}{dt} + K_i \cdot \int edt \tag{6}$$

2.2 Theoretical solution involving MPC and tracking the centerline of a lane

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

Translational component Given the pose of the vehicle at time t as (x_c, y_c, v_c, ψ_v) and two range scans at $+90°$ and $-90°$ with respect to the longitudinal axis of the vehicle which are denoted as CL and CR respectively, the error in translational terms is

$$e_x = -\frac{CL - CR}{2}\sin\psi \tag{7}$$

$$e_y = \frac{CL - CR}{2}\cos\psi \tag{8}$$

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

$$OC = OR - CR = \frac{L}{2} - CR = \frac{CL + CR}{2} - CR = \frac{CL - CR}{2} \tag{9}$$

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

Furthermore, in the COM triangle:

$$O'C = OC\cos\mu = OC\cos(\frac{\pi}{2} - \psi) = OC\sin\psi = \frac{CL - CR}{2}\sin\psi \tag{10}$$

$$O'O = OC\sin\mu = OC\sin(\frac{\pi}{2} - \psi) = OC\cos\psi = \frac{CL - CR}{2}\cos\psi \tag{11}$$

In other words, at time t the vehicle should have been at point $O(x_o, y_o)$:

$$x_o = x_c - \frac{CL - CR}{2}\sin\psi \tag{12}$$

$$y_o = y_c + \frac{CL - CR}{2}\cos\psi \tag{13}$$

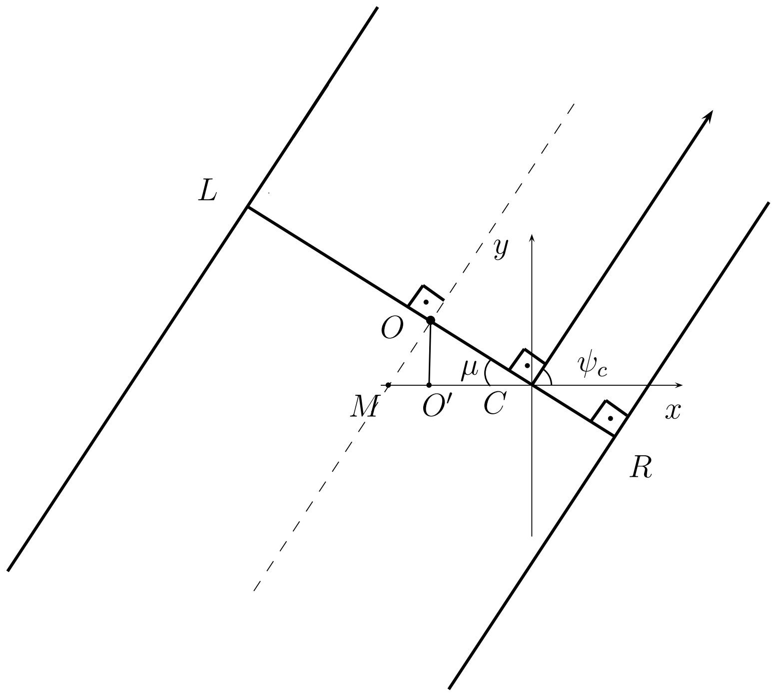

Figure 3: The vehicle's heading angle is equal to that of the lane's. However, its position is off track.

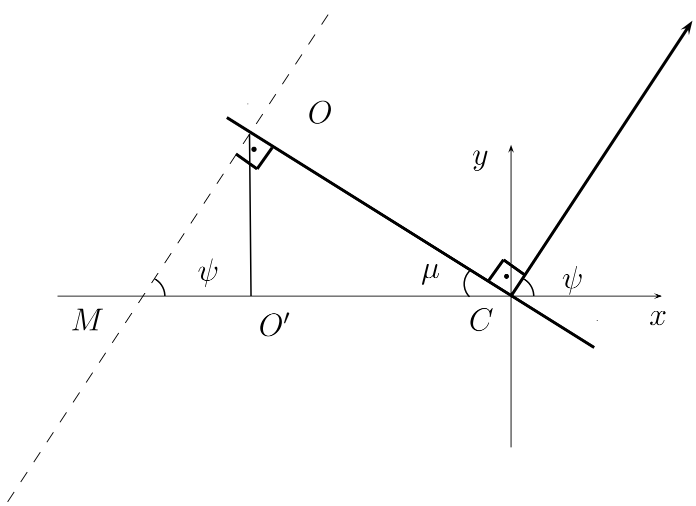

Figure 4

Rotational component With regard to rotation, given the pose of the vehicle at time t as (x_c, y_c, v_c, ψ_v) and three range scans at $+90°$, $0°$ and $-90°$ with respect to the longitudinal axis of the vehicle which are denoted as CL, CF and CR respectively, the heading angle error is

$$\phi = \frac{\pi}{2} - \tan^{-1}\frac{CF}{CR} \tag{14}$$

since $\mu + \phi + \frac{\pi}{2} = \pi$ and $\tan(\mu) = \frac{CF}{CR}$.

In other words, at time t the vehicle should have a heading angle of

$$\psi_o = \psi_c + \phi = \psi_c + \frac{\pi}{2} - \tan^{-1}\frac{CF}{CR} \tag{15}$$

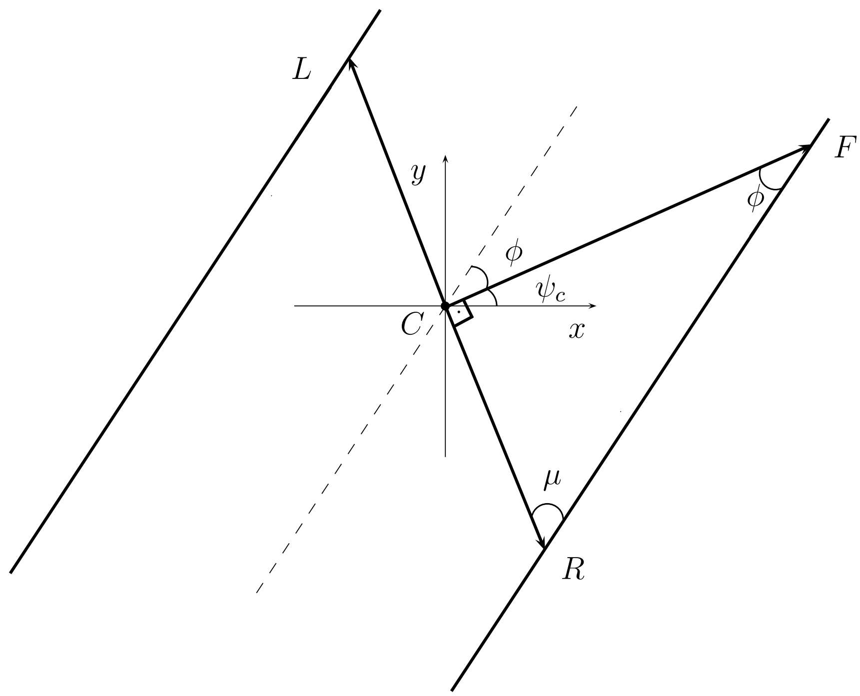

Figure 5: The vehicle's position and its reference are equal. However, the vehicle's heading is off track.

Hence we can formulate the optimization problem as

$$min \sum_{k=0}^{N} (X - X_o)^T Q (X - X_o) + U^T R U \tag{16}$$

$$\text{subject to } X[t+1] = AX[t] + BU[t] \tag{17}$$

$$-U_{min} \leq U \leq U_{max} \tag{18}$$

where $X = [x_c, y_c, v_c, \psi_c]^T$, $X_o = [x_o, y_o, v_o, \psi_o]^T$, N is the horizon and $U = [v, \delta]^T$ is the input vector. Positive definite matrices Q, R will have to be adjusted experimentally.

However, the vehicle's velocity is not measurable since the vehicle does not have encoders connected to its wheels and MOCAP or range scans cannot provide measurements of velocity. Either a Kalman filter will have to be employed in order to estimate the vehicle's velocity, or the ESC feature of the vehicle will have to be investigated with regard to its ability to ensure that the input velocity is indeed the vehicle's velocity.

3 Issues

- The ethernet adapter for the lidar is broken and needs to be replaced. This means that packages `circular_mpc` and `centerline_mpc` cannot be tested until communication with the lidar is fixed.
- The SML lab is booked for the week 14/11-18/11 (what about the weekend?), hence there is no access to MOCAP. This means that packages `circular_pid` and `centerline_pid` (gains need adjusting) cannot be tested until at least Saturday 19/11.
- Package `circular_pid`, which was to be working out-of-the-box, does not work. The fault lies somewhere within ROS: it appears that when ROS_MASTER runs outside Jetson, sometimes communication between Jetson and the nodes running outside it is not established. When it is established, no messages are getting through to teensy.

4 To do

- Consult with Mohamed about the newly formed solution to the `centerline_pid` problem. If correct, include it in the code.
- Tune gains of the PID concerning package `centerline_pid`.
- Implement `centerline_mpc`.
- Implement `circular_mpc`.
- A node that handles the linearization of the kinematic model of the vehicle has to be written in ROS.
- Construct a Kalman filter to estimate the velocity of the vehicle. Even in the case where we had encoders, in the case of high slip, an encoder is irrelevant. For instance, think about the case where the surface that the vehicle's wheels touch is ice.

  However, the incorporation of a Kalman filter for estimating a state presupposes that this state *is being measured*, however large its uncertainty might be. In our case the velocity is not being measured at all. Or is our input velocity considered to be a measurement of the vehicle's velocity?

  Can measuring displacements in x, y from MOCAP and then deducing the vector of the vehicle's velocity be considered as velocity measurements?

  For instance, let us consider two consecutively sampled points in 2D, at time t and at time $t+1$. Then, the magnitude of the vehicle's velocity at time t *was*

$$V_t = \frac{\sqrt{(x_{t+1} - x_t)^2 + (y_{t+1} - y_t)^2}}{T_s} \tag{19}$$

  and the orientation of that velocity was

$$\underline{/V_t} = \beta_t + \psi_t = \tan^{-1}\frac{y_{t+1} - y_t}{x_{t+1} - x_t} \tag{20}$$

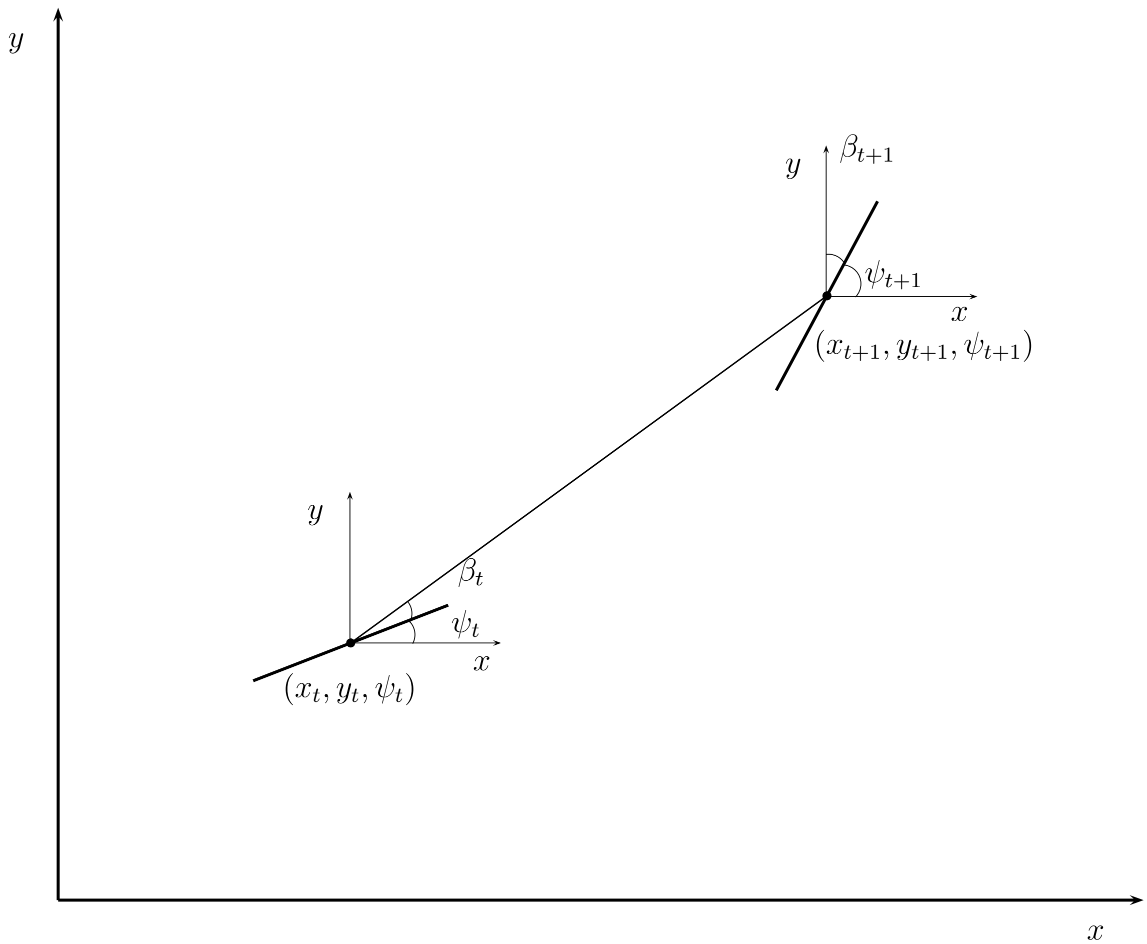

Figure 6

Provided that the sampling time of the vehicle's pose is sufficiently high, its velocity vector can be considered to be roughly equal between sampling times. In principle, MOCAP measures the pose of the vehicle 120 times a second. The maximum velocity that the specific car can achieve is 60 km/h= 16.7 m/s. In this case, the car travels 14 cm in 1/120 seconds. The vehicle's heading is governed by

$$\psi_{t+1} = \psi_t + \frac{T_s V_t}{l_r}\sin\beta \tag{21}$$

and its maximum heading angle difference between sampling times in our case is

$$\Delta\psi_{max} = \psi_{t+1} - \psi_t = \frac{T_s V_t}{l_r}\sin\beta_{max} \tag{22}$$

but β_{max} can be calculated from

$$\beta = +\tan^{-1}(\frac{l_r}{l_r + l_f}\tan\delta) \tag{23}$$

since the function is piecewise strictly increasing. Assuming that the maximum steering angle $\delta_{max} = 60°$, and $\beta_{max} = 0.71$ rad $= 27.2°$, which makes $\Delta\psi_{max} = 21.7°$. In summation, the vehicle's orientation can change by as much as 22° while its displacement can change by as much as 0.14m between sampling instances.

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`