Implementing Lateral Control: Simulink

In last week's lab, we compared the kinematic bicycle model with the dynamic bicycle model for lateral control. This week we will run a closed loop simulation of a vehicle driving on a road while using an on-board camera to find the lanes.

Recall the dynamic bicycle model with a modification where we use linear velocity v_x as an input:

$$\begin{aligned}
\dot{v}_y &= \frac{C_f}{m}\left(\delta_f - \frac{v_y + l_f\omega_z}{v_x}\right) + \frac{C_r}{m}\left(-\frac{v_y - l_r\omega_z}{v_x}\right) - \omega_z v_x \\
\dot{\omega}_z &= \frac{l_f C_f}{I_z}\left(\delta_f - \frac{v_y + l_f\omega_z}{v_x}\right) - \frac{l_r C_r}{I_z}\left(-\frac{v_y - l_r\omega_z}{v_x}\right) \\
\dot{x} &= v_x\cos(\psi) - v_y\sin(\psi) \\
\dot{y} &= v_x\sin(\psi) + v_y\cos(\psi) \\
\dot{\psi} &= \omega_z
\end{aligned} \tag{1}$$

with states $z = \begin{bmatrix} v_x, & v_y, & \omega_z, & x, & y, & \psi \end{bmatrix}^T$ and inputs $u = \begin{bmatrix} v_x, & \delta_f \end{bmatrix}^T$ where

$$\begin{aligned}
x &= \text{inertial X coordinate of CoM} \\
y &= \text{inertial Y coordinate of CoM} \\
\psi &= \text{global heading angle} \\
v_x &= \text{longitudinal velocity of the vehicle} \\
\delta_f &= \text{steering angle of the front wheel} \\
l_r &= \text{distance from the CoM to the rear axle} \\
l_f &= \text{distance from the CoM to the front axle} \\
m &= \text{vehicle mass} \\
\omega_z &= \text{yaw rate} \\
C_r &= \text{rear tire cornering stiffness} \\
C_f &= \text{front tire cornering stiffness} \\
I_z &= \text{yaw moment of inertia}
\end{aligned}$$

We define the following parameter values for our system:

$$\begin{aligned}
m &= 1650[\text{kg}] \\
I_z &= 2235\ [\text{kg m}^2] \\
l_r &= 1.15214\ [\text{m}] \\
l_f &= 1.69286\ [\text{m}] \\
C_r &= 20000\ [\text{N/rad}] \\
C_f &= 20000\ [\text{N/rad}]
\end{aligned}$$

Task 1 Understanding the Simulator (No Deliverable)

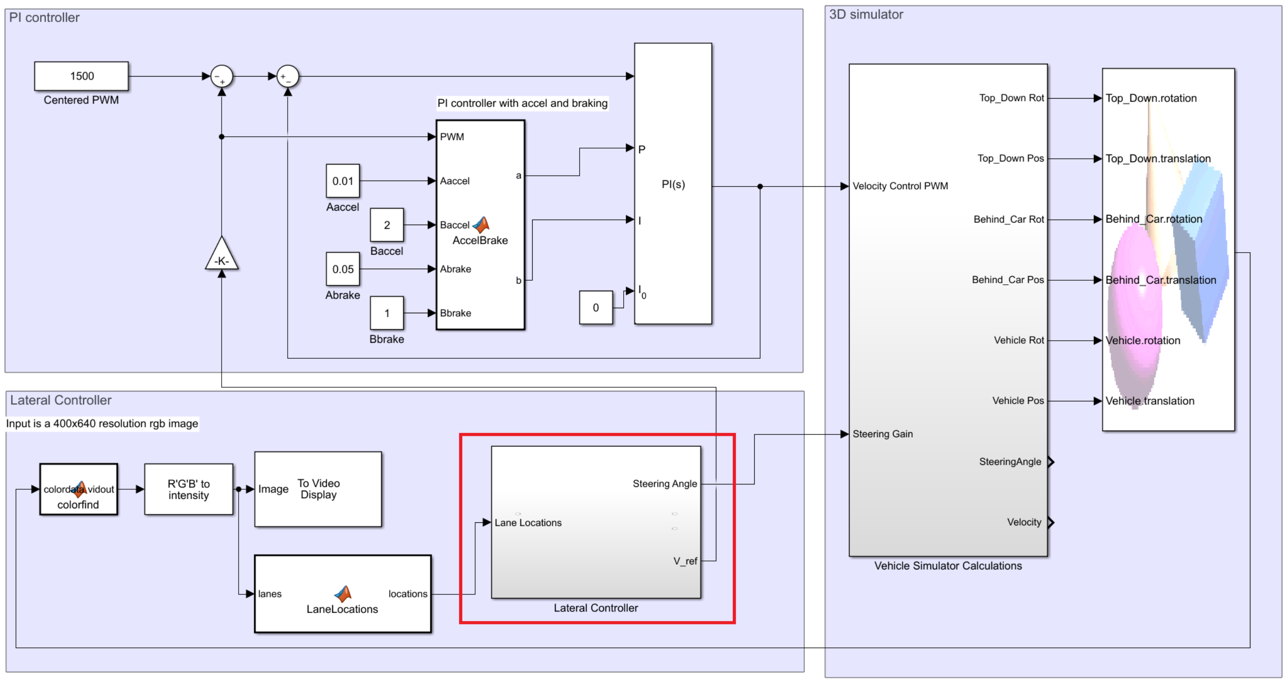

Figure 1: What the sim will look like when first opened.

To run this Simulink model a few toolboxes are required.

1. **Required**: Computer Vision System https://www.mathworks.com/products/computer-vision.html
2. **Required**: Simulink 3D Animation https://www.mathworks.com/products/3d-animation.html

The dynamic bicycle model is already implemented in the **POSupdate** block which is inside the **Vehicle Simulator Calculations** block. Feel free to poke around in the vehicle simulator calculations block. The only part that you will be required to fill out is the Lateral Controller subsystem boxed in **red**.

To run the simulator:

1. Run the **LateralLaneKeeping.m** file to make sure that the initial conditions as well as controller gains are updated every run.
2. To view the video feed of the lane keeping open up the **VR to Video** block (icon with a sphere, cone, and cube).

Task 2 Image Processing and Lateral Control

In Simulink, a virtual camera generates a road image from the top view as shown the left image in Figure 2. This image is always given with respect to the COM of the car at (0,0) in the inertial frame. The RGB image is then converted into a image of just the lane edges. In this task, you are asked to steer the vehicle within the lane boundary by following the centerline.

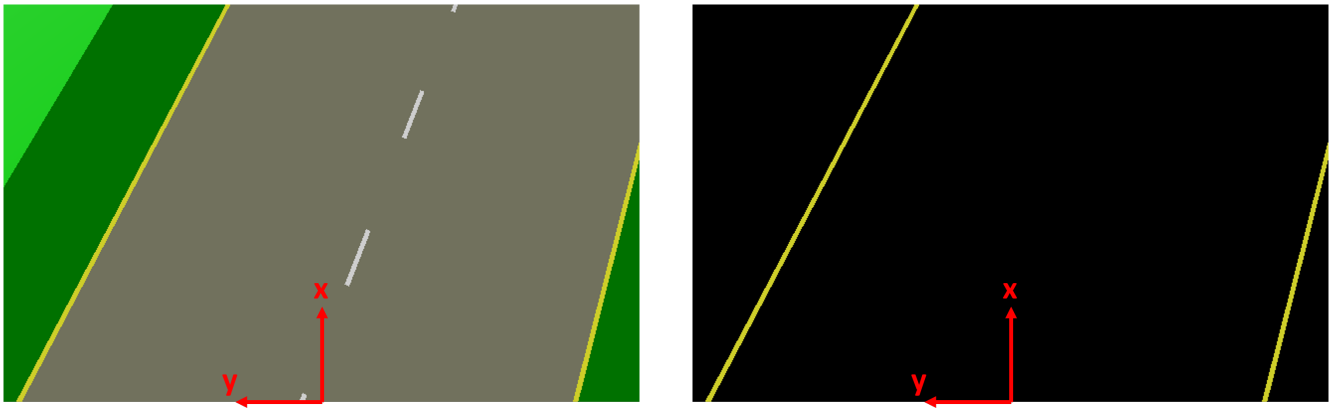

Figure 2: Lanes from Camera View

As an input to the **Lateral Controller** subsystem, you are given an $n \times 2$ array of the lane edges locations [meters] in the inertial frame. It is up to you to decide how to use those lane points in order to find an appropriate steering angle [rad] and velocity [m/s] to send as a controller output.

$$LaneLocations = \begin{bmatrix} x_0, y_0 \\ x_1, y_1 \\ \vdots \\ x_n, y_n \end{bmatrix} \qquad u = \begin{bmatrix} \delta_f \\ v_x \end{bmatrix}$$

The lane edge locations are given starting at the bottom left and searched along the same row. At the end of the row, it searches from the start of the next column above. An example is shown below where the non-zero matrix entries are locations of the edge.

(5,-4)	0	0	0	0
(4,-4)	0	0	0	0
(3,-4)	0	0	0	0
(2,-4)	0	0	(2,2)	0
(1,-4)	0	0	0	(1,4)

(x_6, y_6)	0	0	0	0
(x_5, y_5)	0	0	0	0
(x_4, y_4)	0	0	0	0
(x_2, y_2)	0	0	(x_3, y_3)	0
(x_0, y_0)	0	0	0	(x_1, y_1)

1. ******Deliverable***: Complete the **Lateral Controller** subsystem. Use any controller (LQR, PI, PID, etc) you'd like and explain how the controller works. Include a screenshot of the controller in your PDF.
2. *****Deliverable**: Submit a video recording of the lane keeping simulation as a video link in the PDF. The car should follow the road for the entire duration of the simulation and be able to start with any reasonable initial condition.