PROJECT SPECIFICATION

Model Predictive Control (MPC)

Compilation

CRITERIA	MEETS SPECIFICATIONS
Your code should compile.	Code compiled without errors with cmake and make.

Implementation

<table>
<tr><th>CRITERIA</th><th>MEETS SPECIFICATIONS</th></tr>
<tr><td>The Model</td><td>
The model used is a kinematic bicycle model.

State fo the the model <u>state</u> includes:
x position of the vehicle : x
y position of the vehicle : y
orientation of the vehicle : psi
velocity of the vehicle in m/s : v
the cross-track error : cte
the orientation error : epsi

The control inputs for the <u>Actuators</u> are:
steering angle : delta
acceleration : a

The moving model of the vehicle is showing below describes how the state of the vehicle at time t evolves towards the state at the next time step, t+1 = t + dt, where dt is the timestep duration. As shown below,

// values at timestep [t+1] based on values at timestep [t] after dt seconds
// Lf is the distance between the front of the vehicle and the center of gravity

x[t+1] = x[t] + v[t] * cos(psi[t]) * dt;
y[t+1] = y[t] + v[t] * sin(psi[t]) * dt;
psi[t+1] = psi[t] + v[t]/Lf * delta[t] * dt;
v[t+1] = v[t] + a[t] * dt;
cte[t+1] = f(x[t]) - y[t] + v[t] * sin(epsi[t]) * dt;
epsi[t+1] = psi[t] - psi_des + v[t]/Lf * delta[t] * dt;
</td></tr>
</table>

CRITERIA	MEETS SPECIFICATIONS
Timestep Length and Elapsed Duration (N & dt)	The prediction horizon T is the product of the timestep length N and elapsed duration dt. Timestep length refers to the number of timesteps in the horizon and elapsed duration is how much time elapses between each actuation. I tried values 25 15,10 for N and 1.2, 0.5, 0.8s for dt. The prediction that worked well and performed with best result was one second, with N = 10 and dt = .1.
Polynomial Fitting and MPC Preprocessing	A polynomial is fitted to waypoints. By subtracting each point from the current position of the vehicle, the points are first transformed into the vehicle's coordinate system, making the first point the origin Orientation is transformed to 0 so that the heading is straight forward. Each point is rotated by psi degrees. Subsequently, the vector of points is converted to an Eigen vector hence it is ready to be an argument in the polyfit function where the points are fitted to a 3rd order polynomial. That polynomial is then evaluated using the polyeval function to calculate the cross-track error. Making the first point as origin, the points were transformed into Vehicle's co-ordinate system.
Model Predictive Control with Latency	A delay of 100 ms need to be taken care of after the MPC works. When this latency was first introduced, the model oscillated about the reference trajectory and, at high speeds, drove off the track. In order to deal with the latency, the initial states to be the states after 100 ms. This could enable vehicle to "look ahead" and correct for where it will be in the future instead of where it is currently positioned.

Simulation

CRITERIA	MEETS SPECIFICATIONS
The vehicle must successfully drive a lap around the track.	Simulator Result Video is uploaded @ : https://youtu.be/tlsCYVPDK_k In the video, Vehicle can found to be - Maintaining on the track properly, Decreasing the speeds @ curves & Increasing the speeds @ straight drives