**Model Documentation – Highway Driving**

1. **Introduction:**

The goals of the project are:

* No collision at any time with other vehicles
* Maximum speed of 50 MPH (~ 80 KMH)
* Maximum acceleration of 10 m/s²
* Maximum jerk of 10 m/s³
* Vehicle cannot be in between lanes for more than 3 seconds
* Vehicle cannot go outside the 3 lanes of the highway
* Vehicle cannot drive on the wrong side of the highway

The path planning concept is implemented in this project. There are three components for path planning in this project:

* Predicting the future location of other vehicles on the road
* Deciding when to execute a maneuver (slow down, change lanes etc) based on project goals
* Building a trajectory for maneuvers based on project goals

1. **Rubric Points:**

**Compilation:**

**The code compiles correctly**

The code compiles correctly without any errors

**Valid Trajectories:**

**The car is able to drive at least 4.32 miles without incident.**

The car drives more than 6 miles without incident.



**The car drives according to the speed limit.**

The car is always within the speed limit and does not go too fast nor too slow. It slows down only when obstructed

**Max Acceleration and Jerk are not Exceeded.**

The max limits are not exceeded.

**Car does not have collisions**

There are no collisions

**The car stays in its lane, except for the time between changing lanes.**

The car doesn’t spend more than 3s length outside the lane line during lane changing maneuver. Every other time it stays inside one of the 3 lanes

**The car is able to change lanes**

The car smoothly changes lanes when there is a slow car in front of it and when the adjacent lanes are clear from other traffic

**Valid Trajectories:**

The concepts and code used are the same from Q&A video provided by Udacity in the project section. Additional comments are added to improve readability and for a better understanding of the code. The function code is divided into 4 sections:

* 1. Prediction Step 1– Identify other vehicles in different lanes and their respective velocities
  2. Prediction Step 2 - Identify where the vehicles would be in the future
  3. Behavior Planning – Identify when to change lanes depending on other vehicles speed and position
  4. Trajectory Generation – Generate a smooth trajectory for lane change

Prediction Step 1:

This step is coded from lines 132 – 167

This step iterates through all the sensor fusion data and identifies the lanes ids for all the other cars present in the road. It also calculates the velocities of the cars from the sensor fusion data

Prediction Step 2:

This step is coded from lines 167 – 198

This step iterates through all the sensor fusion data and calculates where the target cars would be present in the future. It also sets the flag for car in front/left/right. This flag is set if the target car would be within 30m of the model car, in either of the lanes in the future.

Behavior Planning:

This step is coded from lines 200 – 239

The required maneuver is performed depending on the state of the vehicle. If the car in front flag is set:

* The model car speed is reduced
* If the model car’s lane is either middle/right and the immediate left lane is open, then change lane to the left
* If the model car’s lane is either middle/left and the immediate right lane is open, then change lane to the right

If the car in front flag is not set:

* Increase the speed to the Max speed limit (49.5 MPH)
* If model car is not in middle lane
  + If the middle lane is empty while being in left or right lane, move to middle lane

The condition of going from left to right lane and vice versa is not considered for simplicity

Trajectory Calculation:

This step is coded from lines 276 – 419. The concepts and logic are taken from the Q&A section of the project section. The trajectories are calculated with the help of splines based on the speed and lane output from the behavior planning module, the car coordinates, and the previous path points

Spline Calculation: The last two points of the previous trajectory or the car’s position (lines 289 – 322) along with three equally spaced points (30, 60, 90 m) (lines 324 - 336) are used to initialize the spline calculation. The coordinates are transformed to the local vehicle coordinate system for easier calculation (lines 338 – 354)

Path Planner: For smooth trajectory, the two points from the previous path are added first to the planner vector (lines 363 – 372). The next points are interpolated for 30m ahead with the help of the generated spline function and basic trigonometry. These are then converted into the global co-ordinate system and added to the planner vector (lines 374 – 418)