# Assignment 8 Report

## Trajectory Generation

1. How is lane following achieved?

Lane following is achieved using the s and d Fernet coordinates by incrementing s by the desired distance to move forward in the lane (which is determined based on the desired speed), while keeping the d value constant (to remain in the same lane). Then, XY coordinates with respect to the global frame of reference for the ego vehicle can be determined using geometry transformations.

1. How to use spline to generate a smooth trajectory?

The current location of the ego vehicle is used as the starting reference point. The calculated trajectory is a set number of points, and spline will use the previously calculated points to add onto the existing trajectory instead of recalculating the full trajectory each time.

The trajectory is calculated by specifying a distance in the horizon and finding the correlated point on the generated spline, this will represent the endpoint of the smooth trajectory plan. Next, a linear distance between the ego vehicle and endpoint is calculated and this distance is used to calculate the number of equidistance points located along the linear trajectory based on the desired velocity and number of points visited per second.

Once the number of points is calculated, these points can then be plugged into the spline to find the corresponding y-values on the spline. These generated coordinates are then transformed into XY coordinates in the global frame of reference.

1. How to avoid collision with the car in front?

Collisions with the car in front of the ego vehicle are avoided by specifying a minimum safe following distance (30m) between the ego vehicle and cars in front. Based on the estimated future locations of the vehicles in the next state, if the ego vehicle will be following too closely then it will decelerate and slow down until a safe following distance is re-established.

1. How to avoid a cold start?

Avoiding a cold start can be done by setting the initial speed of the ego vehicle to zero and then incrementally accelerating until cruising speed is achieved. This avoids violating the maximum jerk and acceleration limits and is more realistic driving behaviour.

## Behaviour Planning Overview

There are three key changes made to the provided ego vehicle behaviour: speed matching vehicles in front of the ego vehicle, determining when to switch lanes, and which lane to switch into using a cost-based heuristic.

First, to eliminate the stop and go behaviour of the default code when following a car too closely, the behaviour is updated to try and match the speed of the vehicle in front. However, the ego vehicle must still maintain a minimum safe following distance and will decelerate if this threshold is crossed.

Next, a cost-based heuristic is selected as the maximization of the ego vehicle speed. This is achieved by determining the speed of the current lane and the adjacent lane(s). If the lane is clear then the lane speed is the maximum allowable speed, if the lane is occupied and unsafe to merge into, then the lane speed is -1.0, and finally, if the lane is safe to merge into but there is a car nearby ahead, then the lane speed is the speed of the car ahead. For determining when to switch lanes, the ego vehicle will look to change lanes if it must slow down due to following the vehicle in front too closely. Additionally, scenarios that are illegal or unsafe, such as passing on a shoulder, are eliminated from the finite state machine. As a note, with this logic the current lane could be selected as the optimal lane even if the ego vehicle speed is limited by the car in front and it is safe to merge into another lane.