Trajectory Generation

1) **how is lane following achieved? (1 points)**
   The Frenet coordinate system (s,d) is used to track the lane positions based on the yellow center curves that are used as reference. The d coordinate represents the perpendicular/lateral distance from the center of the yellow curve and since the lanes have a fixed width, lane following is achieved by setting the position of the car within the limits of the desired lane.

2) **how to use spline to generate a smooth trajectory? (2 point)**
   The spline uses a piecewise function of polynomials to fit onto the waypoints of the trajectory. The anchor points are mapped onto the desired path of the car from the reference point. Hereafter, these points are shifted and rotated such that the reference point is at the origin with a yaw of 0. Using these points, we can create a spline from the reference point to the look-ahead point (next set of points generated at a distance d from spline). N equally spaced x-coordinate points are created in between these 2 points based on the distance between them, the velocity of the car and the time step interval. The corresponding y-coordinates of these points are obtained from the spline function and are then transformed back into the original co-ordinate system. These points obtained from the spline function are appended to the path to generate a smooth trajectory.

3) **how to avoid collision with the car in front? (0.5 points)**
   Once a car in the same lane is detected using the sensor fusion data, the position of the car relative to the ego is calculated. If the car is in front of the ego and is within an unsafe distance threshold, the ego is set to decelerate at a constant rate to prevent collision until there is a safe distance gap between the 2 vehicles.

4) **how to avoid cold start? (0.5 points)**
   A cold start is avoided by having the ego car increment its velocity gradually until the maximum speed of 49.5mph is reached. This is done by incrementing the reference velocity at each time interval with an acceleration rate that does not exceed the threshold of 10 m/s^2.

Writeup

Briefly explain your approach for behaviour planning and any modifications to the provided trajectory generation code.

The objective of this assignment was to perform behavior planning such that the ego can decide when to switch lanes and which lane to switch to. It was determined that lane switching was only necessary when the lane in which the ego is, gets blocked by a car ahead that is too close. The starter code provided code to determine this but was modified from a safe distance of 30m to 24m instead. This was modified due to the rate at which the car decelerated upon being too close.

A for loop is used to obtain sensor fusion data of all the cars around the ego. If the car is not in the same lane as the ego, we first check if the ego is at one of the edge lanes of the road and if so, we block the ability to switch lanes past the road edges. Hereafter, the lane position of the other cars is obtained using a function call *lane_pos()* that evaluates the d co-ordinate of the car and assigns it a lane. Once the lane of another car is found, we first check that it is not blocking the potential lane change of the ego. This is done by evaluating the difference of the s co-ordinate between the ego and the car with a minimum threshold of 22m for safety (*abs(check_car_s - car_s) < 22*). If the potential lanes are not being blocked, a cost function is applied to the right and left lane change based on the available space in the desired lane.

Once the for loop gathers the spatial data of the cars around the ego, the ego decides to make a lane change if the following criteria are met for safe lane change: there is a car ahead that is too close, at least one of the lanes to the left or right are not blocked, the ego has a relative velocity above 40mph and the ego has completed more than 130 cycles in the same lane. Hereafter, a lane change is performed giving preference to empty lanes first. If both lanes are occupied, the lane with a higher cost function will be given preference.