**Planning part**

Before introducing our planning algorithm, the defined structure will be introduced first:

1. Point, containing coordinates x and y.
2. ReferencePoint, containing a Point, theta (angle of the path), kappa (curvature), dkappa (radius of curvature).
3. PathPoint, containing a ReferencePoint, s(distance to the first PathPoint).
4. TrajectoryPoint, containing a Point, velocity and acceleration of the vehicle, and relative time from now.
5. ADCTrajectory, containing an array of PathPoints, and a timestamp of milliseconds.

The overall planning process can be roughly divided into two steps, namely generating the reference line and path planning.

1. How to generate the reference line

Reference line means the line we hope our vehicle would follow. In actual motorsports like Formula 1, it is necessary to consider issues like cutting the corners to achieve highest speed. But in the track we are facing, this method would cause lots of difficulties if cutting is considered. Therefore, to simplify this situation, a simple yet effective way is to directly choose the center line of the track as the reference line. But this line is not obtainable directly, information obtained by the program is only scattered points of the buckets that distinguish the internal and external trajectories, which demands our further process.

The method we apply is Delaunay triangulation, using the obtained bucket point coordinates, this area will be Delaunay subdivision to form a triangular net. After that, we can connect the center line of each triangle to provide one feasible center line. After this, a Cost function is defined to evaluate each path, the value is effected by curvature, the distance to the sidelines, etc. This method can minimize the impact of incorrect pile bucket recognition. Here is an ideal example of Delaunay trangulation:

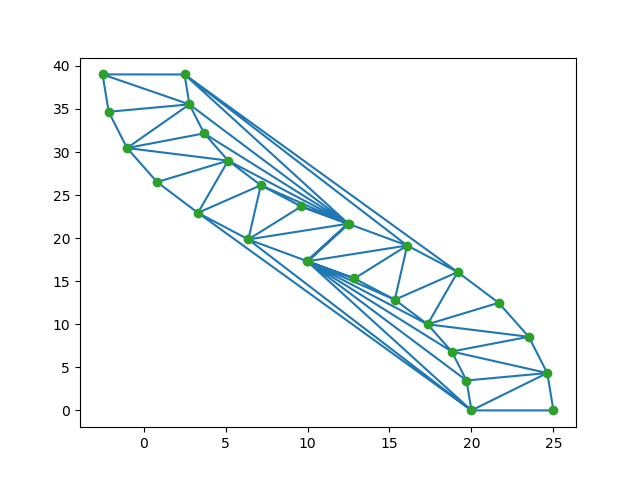


Figure 1. Delaunay Triangulation

However, in actual practice, the Delaunay triangulation process is much too complicated, consuming long time and the effect of identification is not that ideal as the example shows. Thus, we have simplified this method.

First, filter and sort the acquired bucket data according to the current position of the vehicle, and keep only few of the points in the longitudinal direction of the vehicle to eliminate interference. Then, connect the inner and outer buckets to form many segments, and then take the midpoint of them, approximately recognizing this to be the center line of the road. Next, the points are smoothly connected by Cubic Bezier Curve, then the curvature, pointing angle and other information of each ReferencePoint can be calculated.

Detailed application of Bezier Curve is shown in the picture below. With points(anchor points) we already have, we are able to find out the middle point between each one, and then link them, as the black lines show. Then adjust the line, moving till it goes through corresponding anchor point while keeping parallel with the origin one. To better discrete the two segments, we would also reduce the length of it by 0.6, two edges of the segments are treated as control points.

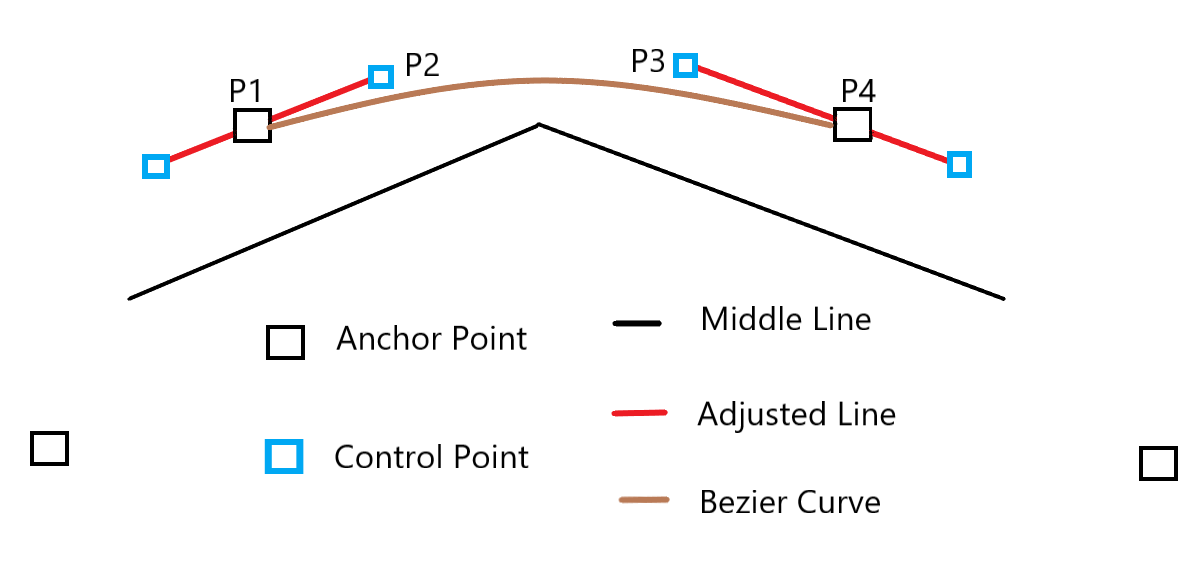


Figure 2. Beizier Curve

With two anchor points along with two control points between them, we can form the bezier equation. By a certain order, we name the four points as P0, P1, P2 and P3. The curve is described by a parameter equation of t. When t = 0, it represents P1, while t = 1, it represents P4. In this equation, the position is expressed as a vector.

Of course, with this method, certain situations cannot be taken into account. For example, only the inner or outer buckets are recognized, and the number of buckets recognized number on the other side is insufficient. In this case, our reference line will no longer be the center of the road. Instead, it tends to keep the distance with edge unchanged to deal with emergency.

(2)Trajectory Path Planning

The purpose of path planning is to construct a route for a period of time in the future, based on the obtaining reference line. As the vehicle itself would inevitably deviate from the reference line, correction is of necessity. Steps are as follows:

1. Discrete the reference line of the ReferencePoints. The parameter difference between PathPoint and ReferencePoint has only one value of s. As the distance between trajectory points is very small, thus by accumulating the distance between adjacent points, s value can be obtained approximately, that is, the curve distance from the first trajectory point.

2. Use the position of the vehicle, to match the discrete PathPoints, and find the matching point. First find the two points closest to the current position in the reference points, and then use vector expressions to derive the data of the points that match the current position.

In this figure, P represents position of the vehicle, while P1 and P2 are the two buckets closest to the vehicle, using projection, we could obtain the matched point in the path, we name it as P3. The formula would be like:

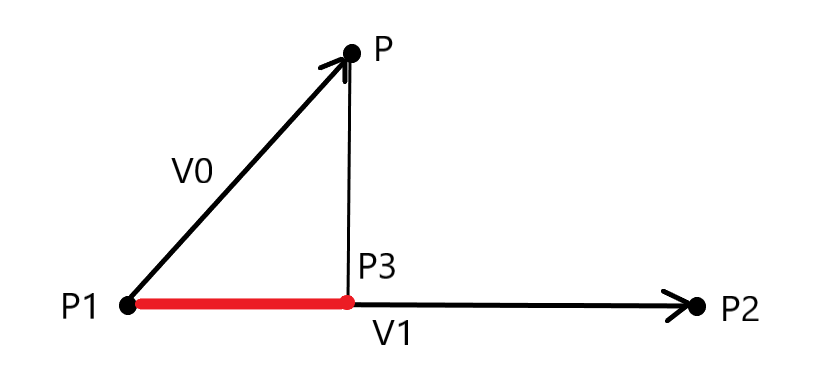


Figure 3. Vector Projection

1. Next, performs coordinate system conversion. Because the longitudinal trajectory is a two-dimensional curve, it is quite hard to accurately describe the path using mathematical expressions. So, traditional Cartesian coordinate system will be converted to the Frenet coordinate system, dividing the path into two processes: the direction of the reference line, or longitudinal direction, shown as s(t) in the figure. And the normal direction of the reference line, or lateral direction, shown as d(t) in the figure.

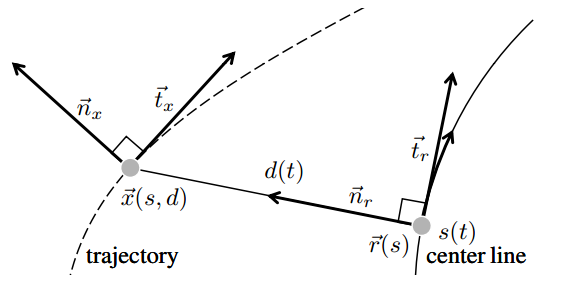


Figure 4. Frenet Coordination

The movement in both directions can be expressed by expressions. The result of the conversion is to obtain the position, velocity and acceleration of the starting point in those two directions. The conversion process between Cartesian coordinates and Frenet coordinate system is as follows:

A point in the Frenet coordination can be described as , in Cartesian coordination would be . Here is a brief introduction of these parameters:

From Cartesian coordination to Frenet coordination:

From Frenet coordination to Cartesian coordination, using matched point as a base, the equation would be:

1. Next, the lateral and longitudinal trajectories need to be generated, and the method applied is random sampling. For the longitudinal direction, two time samples will be taken, t=1s and 2s. According to the current speed of the vehicle, a speed range of the two samples can be obtained. Then divide them evenly to get a series of speed values. When considering the low limit of speed range, we tend to reduce the amount of deceleration to ensure a quick lap time. For the other direction, the time samples are used for correction, after a time gap of t, the vehicle should go back to the path. This period of time is divided into 5 pieces evenly from 5s-9s.

After acquiring the sampled data, we need to further calculate the parameters of the trajectory. In order to facilitate the acquisition of the three attributes of distance, velocity, and acceleration, polynomial expression is used to describe the velocity information. With this expression, along with current state of the vehicle, distance and acceleration can be figured out by integration and derivation. The detailed process is as follows:

For horizontal direction, when t = T, the vehicle should be in the reference line, and lateral speed should be reduced to 0. Then the equation should be like:

For vertical direction, we have 3 data, velocity in 0, 1, 2s. Combing those three, equation can be formed like:

Solving those equations and speed can be precisely formed.

1. After obtaining the two sub-trajectories, we will perform permutation and combination and evaluate all the trajectories. A Cost function is defined considering indicators such as average speed, change in the speed, offset value, steering speed, etc. After all the trajectories are evaluated, they will be sorted for latter use.

6. In the trajectory set, we select trajectory with the least value of Cost every round. Check whether the speed is too high, the steering is too fast, or collision with the pile buckets occurs. If it meets all the requirements, we would perform another transformation from Frenet coordination to Cartesian coordination as mentioned before. And the trajectory information is integrated in a message file and published.