

## MPC Path Planner

**This memo is only an idea sketch; it does not include detailed discussion.**

At current time, solve following optimization problem.  $N$  is a horizon length,  $m$  is the number of detected static and dynamic(moving) obstacles. For the simple notation, let initial time be 1 for all time windows. The pair  $z_k := (x_k, y_k)$  is a position of vehicle at time  $k$ .  $\tau_k^i$  is the position of the  $i^{th}$  obstacle at time  $k$ .  $T$  is a goal position that the ego vehicle wishes to arrive in the future. Fig.1 shows a concept of this approach.

### Modeling

Both ego vehicle and dynamic obstacles are unicycle model.

$$\begin{aligned}x_{k+1} &= x_k + v_k \Delta \cos \theta_k (+\Delta^2 \alpha_k \cos \theta_k), \\y_{k+1} &= y_k + v_k \Delta \sin \theta_k (+\Delta^2 \alpha_k \sin \theta_k), \\ \theta_{k+1} &= \theta_k + \Delta \omega_k, \\ v_{k+1} &= v_k + \Delta \alpha_k\end{aligned}\tag{1}$$

Inputs are yaw rate  $\omega$  and acceleration  $\alpha$ . High order terms () can be omitted for simplicity.

### MPC formulation

Inputs:

$$u = \left\{ \begin{bmatrix} \alpha_1 \\ \omega_1 \end{bmatrix} \cdots \begin{bmatrix} \alpha_N \\ \omega_N \end{bmatrix} \right\} \in R^{2 \times N}\tag{2}$$

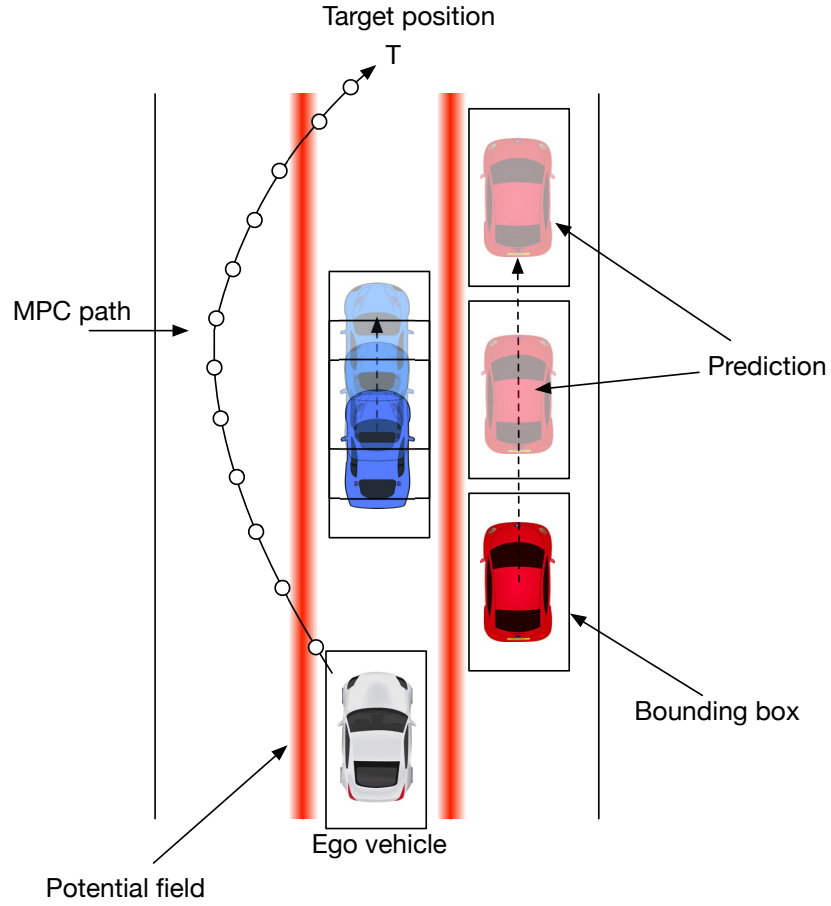


Figure 1: MPC path planning concept

Optimization problem:

$$\begin{aligned}
& \min_{u \in U} \sum_{k=1}^N w_{goal} D_k^2(z_k) + w_{acc} \|\ddot{z}_k\|^2 + w_{jerk} \|\ddot{\dot{z}}_k\|^2 + w_{yaw} \dot{\theta}^2 + w_{lane} L(x_k, l_k, r_k) \\
& s.t. \quad \text{Dynamics for ego and obstacle vehicles,} \\
& \quad - \kappa_{max} \leq \kappa_k \leq \kappa_{max}, \\
& \quad - a_{max} \leq \|\ddot{z}_k\| \leq a_{max}, \\
& \quad - \alpha_{max} \leq \|\dot{\theta}_k\| \leq \alpha_{max}, \\
& \quad \|z_k - \tau_k^i\|_g \geq \epsilon, \quad i \in \{1, 2, \dots, m\}
\end{aligned} \tag{3}$$

where

$$\kappa_k = \frac{\dot{x}_k \ddot{y}_k - \dot{y}_k \ddot{x}_k}{(\dot{x}_k^2 + \dot{y}_k^2)^{\frac{1}{3}}} \quad (4)$$

is a curvature of trajectory at time  $k$ .

$$D_k(z_k) = \|z_k - T\| \quad (5)$$

is a distance to the goal position.

Penalties for left and right lane segments are as follow.

$$L(x_k, l_k, r_k) = \frac{\exp(-\alpha(\|x_k - l_k\|^2 - s^2))}{1 + \exp(-\alpha(\|x_k - l_k\|^2 - s^2))} + \frac{\exp(-\alpha(\|x_k - r_k\|^2 - s^2))}{1 + \exp(-\alpha(\|x_k - r_k\|^2 - s^2))}, \quad (6)$$

where  $l_k$  and  $r_k$  are positions of left and the right lanes at time  $k$ , respectively. This could be defined as distances depending on FORD RNDP map data.  $\|z_k - \tau_k^i\|_g$  is the GJK distance between ego vehicle and obstacles.

$w_{goal}, w_{acc}, w_{jerk}, w_{yaw}, w_{lane}$  are weights.

## Goal Position

Goal should be set based on traffic information such as high way exit location, road merge and so on. Once discrete decision about the number of target lane is made, goal position is set as a large enough offset in longitudinal direction,  $\bar{y}$ . Suppose current time,  $q$  number of lanes are detected. Goal position  $T$  is defined as follow.

$$T := (q^*, y_1 + \bar{y}) \quad (7)$$

where

$$q^* = \arg \max_{r \in \{1, 2, \dots, q\}} Q(z_1, r) \quad (8)$$

Design of  $Q(z_1, r)$  is TBD. This problem will be solved much less frequently than MPC optimization problem.

## Geometric model of vehicles

Vehicles are modeled as bounding boxes as this shape is similar to most vehicles and it is computationally light. The width of box is 80 – 90% of lane width. The length (longitudinal direction of vehicle) of box is a linear function of ego vehicle. This provides larger safe distance to the obstacle vehicle in a high speed.

## Distance to the obstacle model

As vehicles are approximated as a box (or polygon), a method that computes the distance between polygon to polygon must be introduced. Well known GJL (Gilbert-Johnson-Keerthi) method that computes the minimum distance between ploygon is used.

Alternatively, we can use even more simpler method as described below:

**Result:** Distance between two boxes

Set  $dist = 0$ .

Let the number of points on the boundary of box be  $n$ .

Step 1. Obtain direction vector from the origin of the ego vehicle to the  $i^{th}$  box-approximated obstacle  $d_i$

Step 2. Determine which quadrant  $d_i$  located for both ego behicle and  $i^{th}$  obstacle.

For ego vehicle

Let number of points on that quadrant, on the boundary  $m$

**for**  $j \leftarrow 1$  **to**  $m$  **do**

    | Computes  $d_{i,j} = \|proj(d_i, m^{th})\|$

**end**

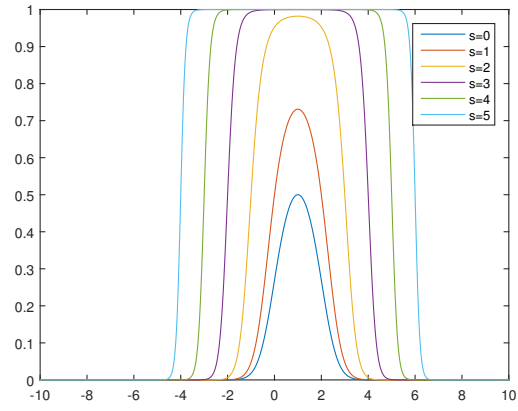
$dist_{ego} = \max\{dist, d_{i,j}\}$

For obstacle, perform similar procedure, and obtain  $dist_i$ .

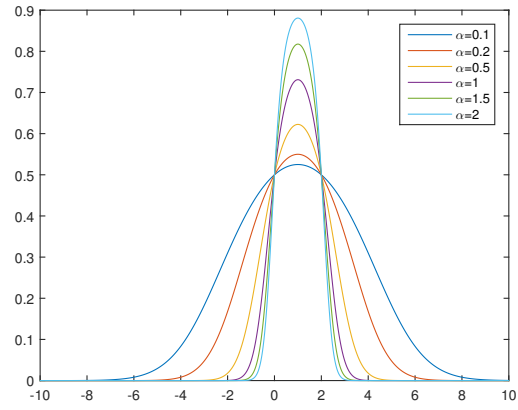
Compute  $dist = dist(Point_{ego}, Point_{obstacle}) - dist_{ego} - dist_i$ .

**Algorithm 1:** Distance computation algorithm

## Geomatic interpretation of potential field



(a) Parametrized by  $s$



(b) Parametrized by  $\alpha$

Figure 2: Penalty term parameterized by  $s$  and  $\alpha$