Maneuver Constraints Adopted by DynNPC

Anonymous Author(s)

1 Maneuver Constraints

According to the data provided by National Highway Traffic Safety Administration (NHTSA), in real-word driving scenarios, vehicles can perform various maneuvers including *decelerating*, *accelerating*, *passing*, *parking*, *lane changing*, *turning left*, *turning right* and *backing up*. Specifically, we utilize the maneuver specifications based on CRISCO to constrain the position of maneuver execution and other related aspects.

Decelerating. When the NPC vehicle N_k decides to decelerate at frame t_0, it must ensure a safe deceleration longitudinal distance to the Ego vehicle behind if they are in the same lane, and activate the brake lights, avoiding sudden braking and rear-end collision. The specification can be defined by Eq. 1,

$$D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0}) \geq threshold \wedge \ brakeSignal(p_{N_k}^{t_0}) = True \quad (1)$$

where $D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0})$ returns the distance between the NPC vehicle and the Ego vehicle at frame t_0 when they drives in the same lane, and $threshold$ is set to 20 meters by default. The function $brakeSignal(p_{N_k}^{t_0})$ returns the status of NPC vehicle's braking signal. Besides, the max decelerating rate is set to 8 m/s^2 as default.

Accelerating. When the NPC vehicle N_k decides to accelerate at frame t_0, if the NPC vehicle is behind the Ego vehicle in the same lane, the maximum speed of the NPC vehicle after acceleration must not exceed the speed of the Ego vehicle to avoid rear-ending the Ego vehicle. Besides, it should ensure that the speed after acceleration does not exceed the speed limit of the current road section. The specification can be defined by Eq. 2,

$$\begin{gathered}\left(D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0}) \leq 0 \Rightarrow MaxSpeed \leq v_E^{t_1}\right) \\ \wedge\ MaxSpeed \leq \text{speedLimit}\end{gathered} \quad (2)$$

where $D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0})$ returns the distance between the NPC vehicle and the Ego vehicle at frame t_0 when they drives in the same lane, and speedLimit is extracted from the map. Besides, the max decelerating rate is set to 8 m/s^2 as default.

Passing. When the NPC vehicle N_k decides to pass the Ego vehicle at frame t_0, it must ensure that it have a sufficient lateral and longitudinal distance before merging back into the original lane. Additionally, the maneuver must not be performed in no-passing zones (e.g., solid-line areas) and should comply with the speed limit. The specification can be defined by Eq. 3,

$$\text{isAdjacentLane}(p_{N_k}^{t_0}, p_E^{t_0}) = \text{True} \ \wedge \ v_{N_k}^{t_0} < \text{speedLimit} \quad (3)$$

Parking. When the NPC vehicle chooses to decelerate to stop, it must ensure that the parking maneuver is performed in a designated parking area, at a complete stop at frame t_n. The specification can be defined by Eq. 4,

$$prohibitParkingZone(p_{N_k}^{t_n}) = False \wedge \ v_{N_k}^{t_1} = 0 \quad (4)$$

where $prohibitParkingZone(p_{N_k}^{t_n})$ returns whether the NPC vehicle is stopping in prohibited areas (e.g., intersections or pedestrian crossings).

Lane Changing. When the NPC vehicle N_k is going to take lane changing maneuver to the left at frame t_0, it must ensure a safe lane changing distance from the following Ego vehicle, activate the left turn signal when changing lanes, and avoid performing the maneuver within a solid-line area. The specification can be defined by Eq. 5,

$$\begin{gathered}D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0}) \geq threshold \\ \wedge\ leftSignal(p_{N_k}^{t_0}) = True \wedge \ isSolid(p_{N_k}^{t_0}) = False\end{gathered} \quad (5)$$

where $D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0})$ returns the longitudinal (*i.e.,* the driving direction) distance between the NPC vehicle and the Ego vehicle at frame t_0, and $threshold$ is set to 30 meters by default. The function $leftSignal(p_{N_k}^{t_0})$ returns the status of NPC vehicle' left turn signal, while $isSolid(p_{N_k}^{t_0})$ indicates whether the surrounding lane markings of the NPC vehicle at frame t_0 are solid lines.

Similarly, when the NPC vehicle N_k is going to take lane changing maneuver to the right at frame t_0, the specification can be defined by Eq. 6,

$$\begin{gathered}D_{N2E}(p_{N_k}^{t_0}, p_E^{t_0}) \geq threshold \\ \wedge\ rightSignal(p_{N_k}^{t_0}) = True \wedge \ isSolid(p_{N_k}^{t_0}) = False\end{gathered} \quad (6)$$

where $rightSignal(p_{N_k}^{t_0})$ checks if the right turn signal is turned on.

Turning Left. When the NPC vehicle N_k decides to turn left at an intersection at frame t_0, it must ensure that the maneuver is performed safely in the left turn lane, following traffic signals (see Speed Planning). Besides, it must activate the left turn signal before initiating the turn. The specification can be defined by Eq. 7,

$$\begin{gathered}isLeftTurnLane(p_{N_k}^{t_0}) = True \wedge \ leftSignal(p_{N_k}^{t_0}) = True \\ \wedge RedSignalObeying = True\end{gathered} \quad (7)$$

where $isLeftTurnLane(p_{N_k}^{t_0})$ ensures that the NPC vehicle only turns left in the left turn lane and $leftSignal(p_{N_k}^{t_0})$ ensures the activation of the left turn signal.

Turning Right. When the NPC vehicle N_k decides to turn right at an intersection at frame t_0, it must ensure that the maneuver is performed safely in the right turn lane. Right turns on red are allowed in most places, but it must activate the right turn signal before initiating the turn. The specification can be defined by Eq. 8,

$$isRightTurnLane(p_{N_k}^{t_0}) = True \wedge \ leftSignal(p_{N_k}^{t_0}) = True \quad (8)$$

where $isRightTurnLane(p_{N_k}^{t_0})$ ensures that the NPC vehicle only turns right in the right turn lane and $rightSignal(p_{N_k}^{t_0})$ ensures the activation of the right turn signal.

Backing up. We do not consider backing up maneuvers because it is the most dangerous maneuver in highways and is forbidden in the crossroad road. Besides, responding to a vehicle that taking backing up maneuver is out of the ODD of current ADSs.