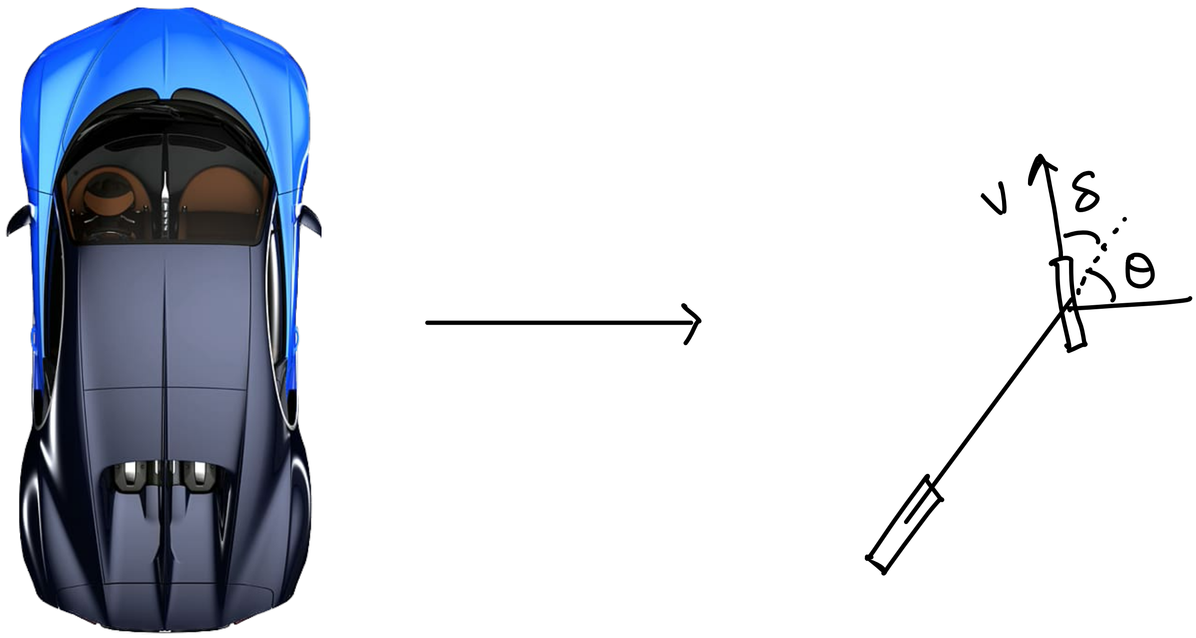
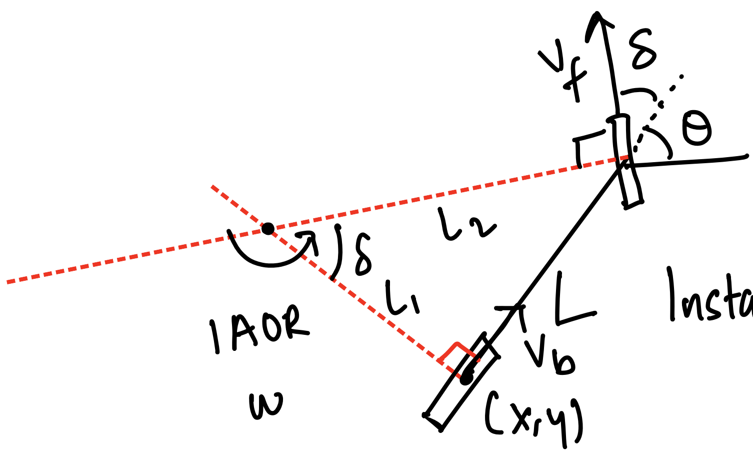
Control for 2D-Bicycle Model Car.

2D Bicycle Model



A simple two wheel - bicycle model for car kinematics



Instantaneous Axis of Rotation

Rigid Body Constraint

$$wl_1 = v_b$$

$$wl_2 = v_f$$

$$v_f \cdot \cos\delta = v_b$$

$$l_1 = l_2 \cos\delta$$

Assumption :→ wheels have no slip.

State of the car : (x, y), θ, δ

(x, y): position, θ: orientation, δ: steering

$\tan \delta = \frac{L}{l_1}$

Control : v_b, $\dot{\delta}$

v_b: speed of the rear wheel

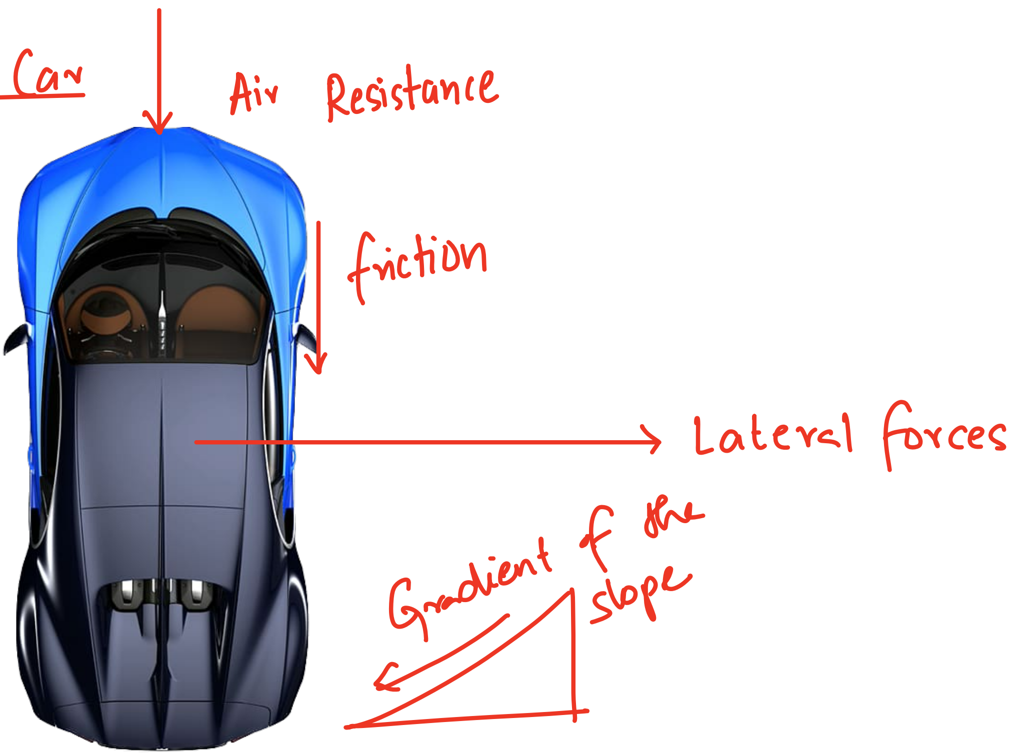
$\dot{\delta}$: rate of change of steering angle

Constraints :

$v < v_{max}$

$\delta < |\delta_{max}|$

Dynamics of the Car



Steering Model

$$\delta < |\delta_{max}|$$

$$\underline{\delta_s} = K_s \underline{\delta}$$

tyre angle

Steering angle

Throttle / Brake

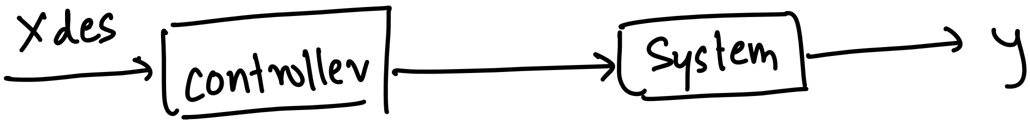
throttle control

$$\delta_t = \underline{K_t a} \quad \text{acceleration}$$

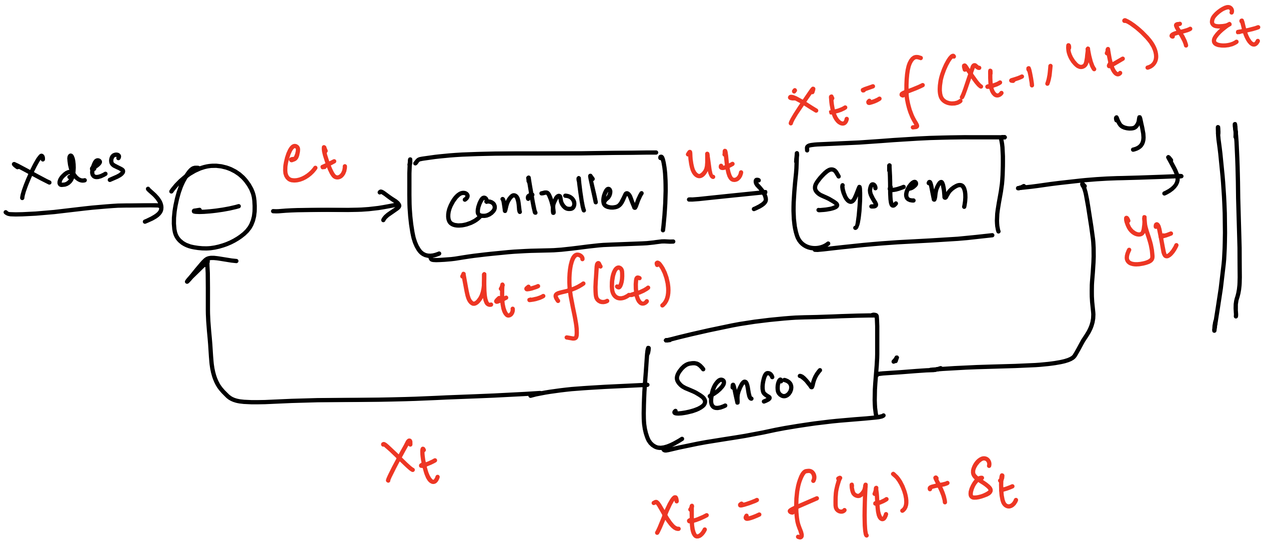
$$\delta_b = -K_b a$$

$$a < a_{max}$$

Feed back Control



|| Open loop



|| Closed loop Feedback

PID Controller

Position Control vs Trajectory Control

$\Downarrow$

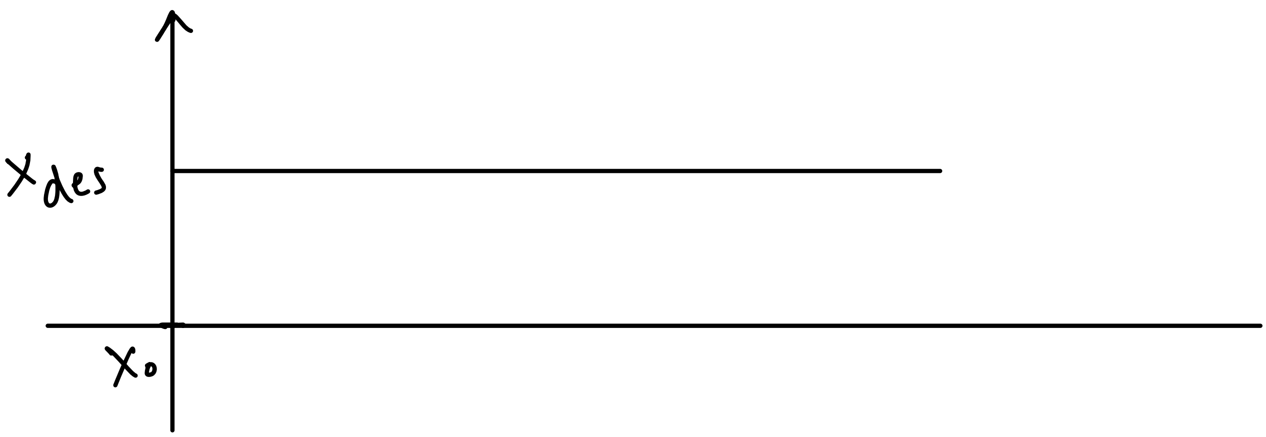
$X_{des} \rightarrow$ goal

$u \rightarrow$ control

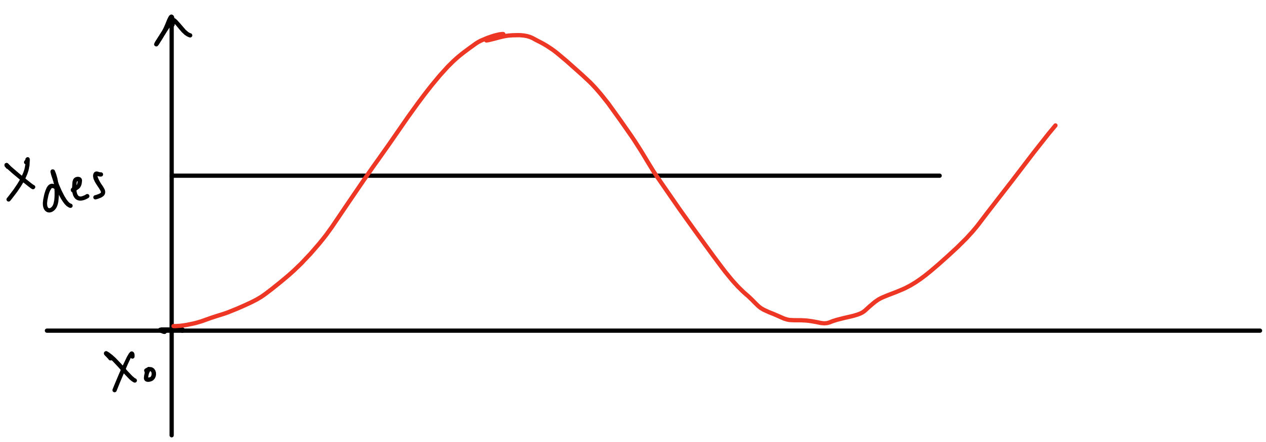
$z \rightarrow$ measurements

Simple Case : Point mass control

$x \rightarrow$ state
$\dot{x} \rightarrow$ control



P - Control $\rightarrow$ $K_p(x_{des} - x_t) = u_t$



$$u_t = a = \ddot{x}$$

$$\Rightarrow \ddot{x} = K_p(x_{des} - x)$$

$$x_{des} - x = \ddot{x}/K \Leftarrow \frac{\delta^2 x}{dt^2} = K_p x_{des} - K_p x \rightarrow 1$$

Assuming, solution $\Rightarrow$ $x = (x_0 - x_{des})\cos(\omega t) \rightarrow$ 3

?

$$\dot{x} = -(x_0 - x_{des})\sin(\omega t)\,\omega$$

$$\ddot{x} = (x_{des} - x_0)\cos(\omega t)\,\omega^2 \rightarrow 2$$

from 1 and 3

$$\frac{\ddot{x}}{x} = -\omega^2$$

$$K_p = \omega^2 \cos(\omega t)$$

?

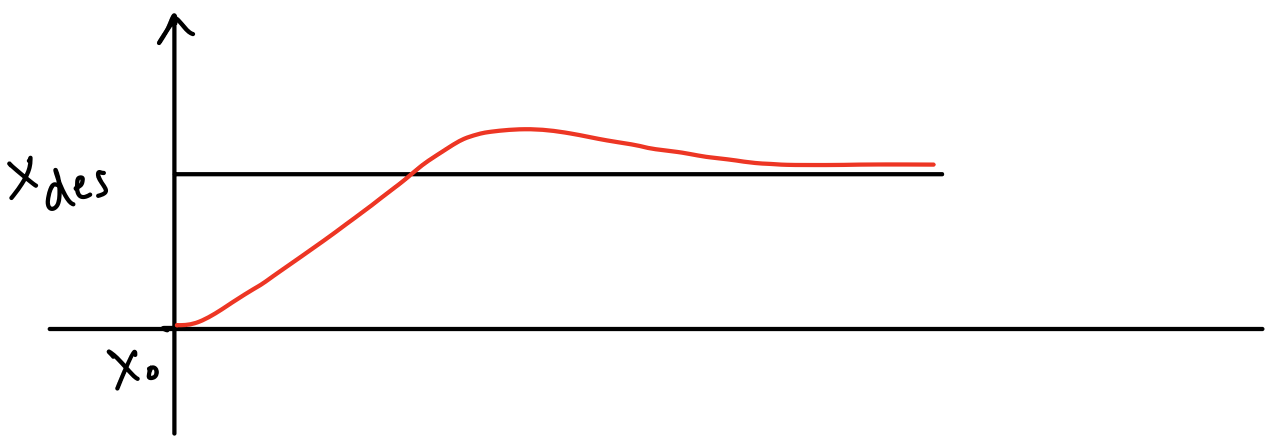
PD Control

$\rightarrow \quad u_t = K_p(x_{des} - x_t) + K_D(\dot{x}_{des} - \dot{x})$

We need Integral term to account for biases. Imagine a extra weight put on top of a drone while controlling height.

PID Control

$\rightarrow \quad u_t = K_p(x_{des} - x_t) + K_D(\dot{x}_{des} - \dot{x}) + K_I\left(\int_0^t (x_{des} - x_t)\,dt\right)$



* Some times → wind up error can cause undesired behaviours
* Imagine forcibly holding a drone by pulling it and leaving at once!

Trajectory Control vs Position Control

$\Downarrow$

Lateral + Longitudinal Control

velocity control $\Rightarrow$ follow a velocity profile

Longitudinal Control (PID)

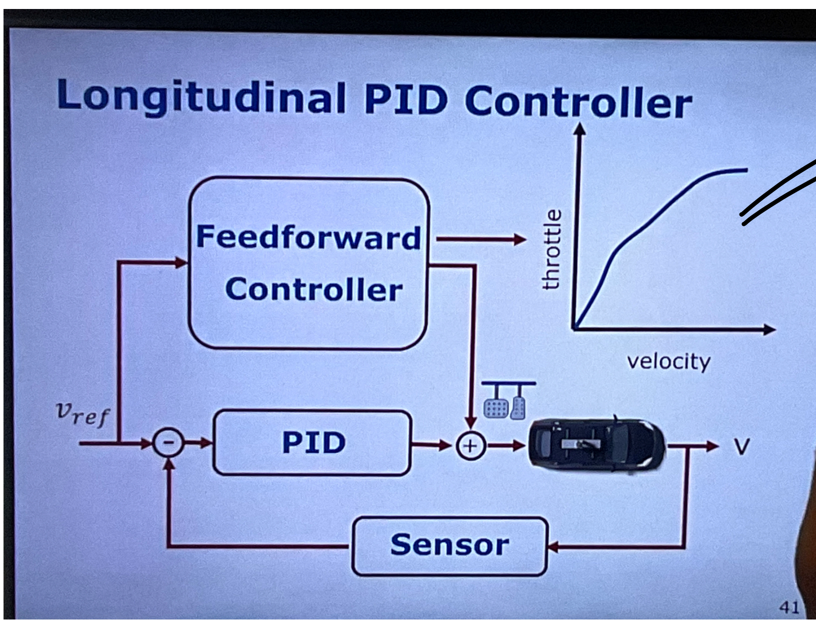
current velocity

$$\ddot{x}_{des} = K_p(\dot{x}_{des} - \dot{x}) + K_d \frac{d(\dot{x}_{des} - \dot{x})}{dt} + K_I \int_0^t (\dot{x}_{des} - \dot{x})\,dt$$

desired acceleration

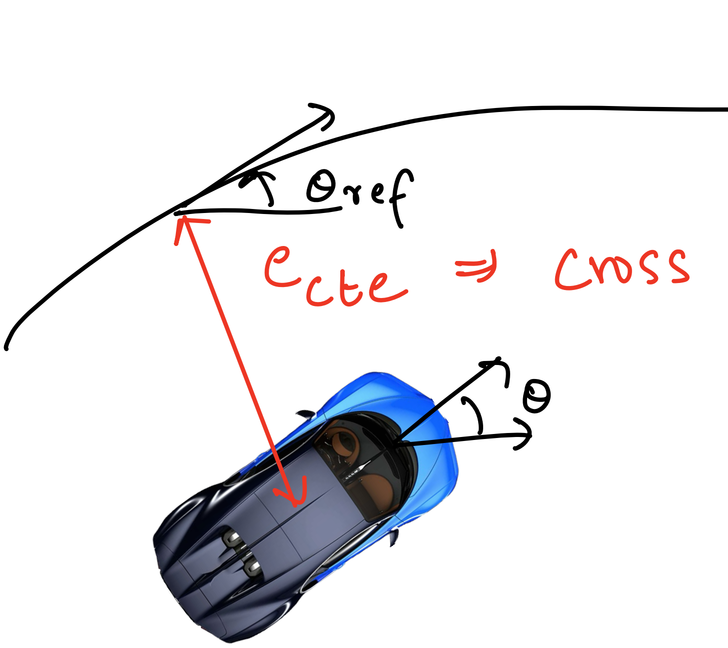
reference velocity

Feedforward Controller (Reaches Goal faster than PID alone)



If we have emperical data for the control vs output $\Rightarrow$ we may as well use a feedforward section to let PID only make small changes

Lateral Control



$e_{cte} \Rightarrow$ cross track error

Goals

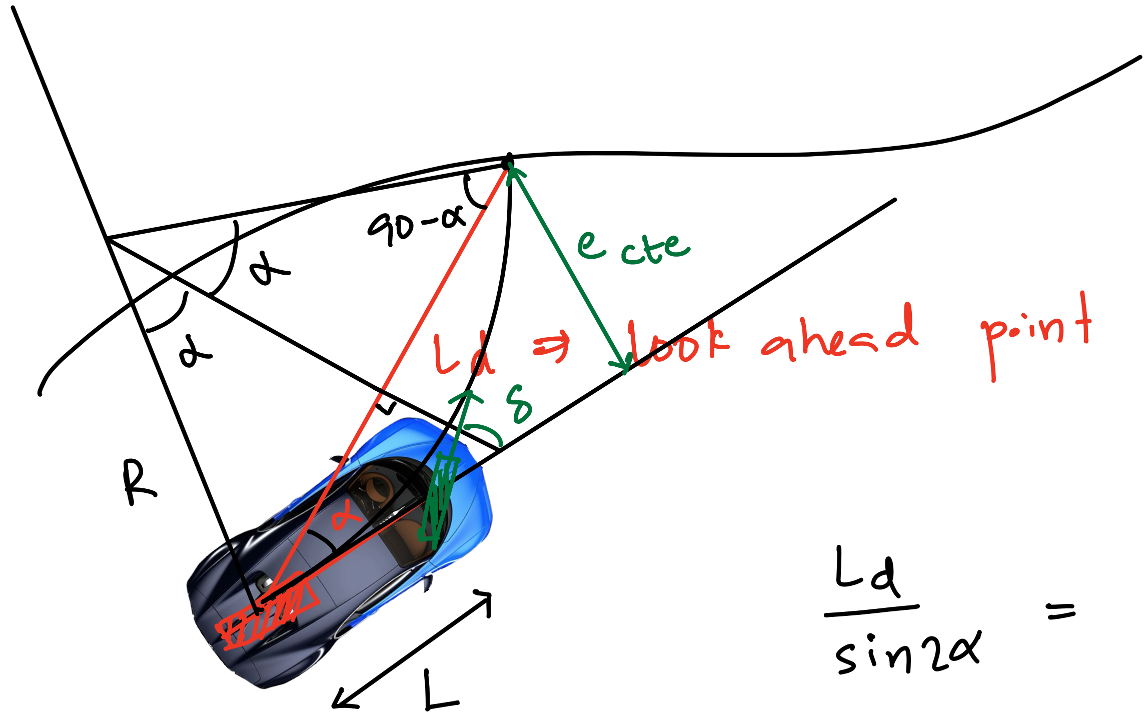
$e_{cte} \to 0$

$\theta_{ref} - \theta \to 0$

$$\dot{\delta}_{des} = \underbrace{-K_p e_{cte}}_{K_p(0 - e_{cte})} - K_d \frac{d e_{cte}}{dt} - K_I \int_0^t e_{cte}\, dt$$

Geometric Steering Control

pure pursuit control



$Ld \Rightarrow$ look ahead point distance

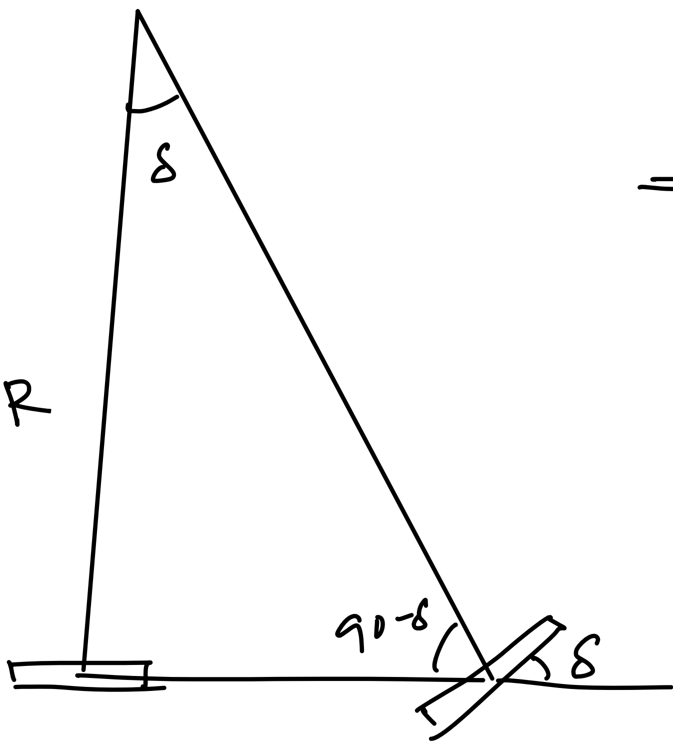
$$\sin\alpha = \frac{e_{cte}}{Ld}$$

$$\frac{Ld}{\sin 2\alpha} = \frac{R}{\cos\alpha}$$

$$Ld = 2\sin\alpha\, R$$

$$\kappa = 1/R = \frac{2\sin\alpha}{Ld} \rightarrow (1)$$

curvature



$$\tan\delta = \frac{L}{R} \rightarrow (2)$$

$$\Rightarrow \quad \delta = \tan^{-1}\left(\frac{2L\sin\alpha}{Ld}\right)$$

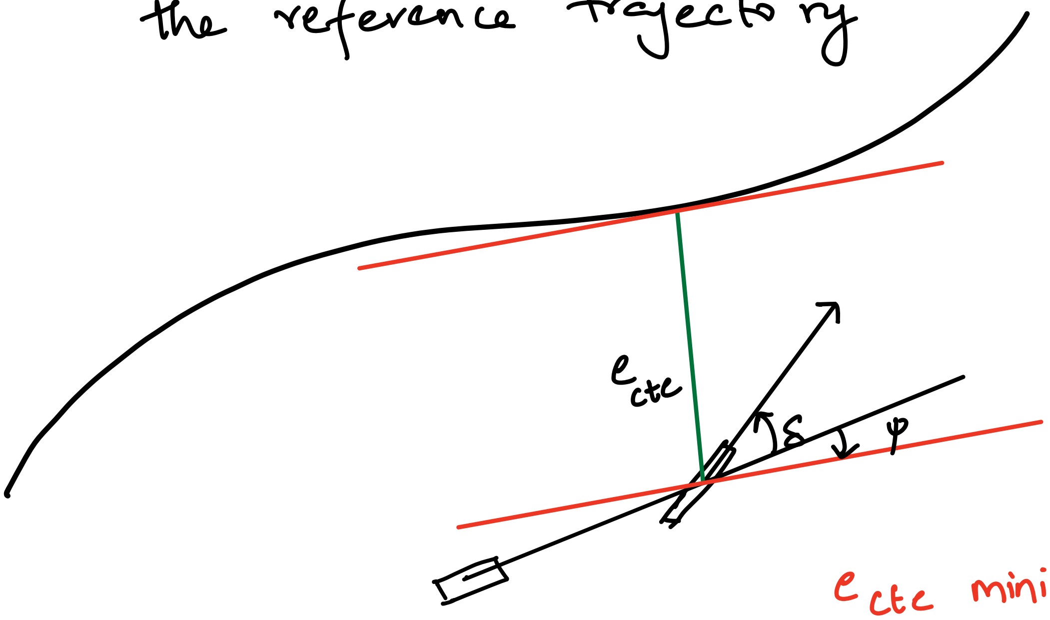
$$\in [-\pi/2, \pi/2]$$

One strategy :→ $L_d = k_{dd} V$ } more speed → longer merging distance

$$\Rightarrow \delta = \tan^{-1}\left(\frac{2L\sin\alpha}{k_{dd} V}\right)$$

Stanley Controller → tracking front wheel.

Reduce both error in heading and nearest point on the reference trajectory



$$\delta = \underbrace{\psi}_{\text{steering direction error minimizer}} + \overbrace{\tan^{-1}\left(\frac{k e_{cte}}{v}\right)}^{e_{cte}\text{ minimizer}}$$

$\delta \in [\delta_{min}, \delta_{max}]$