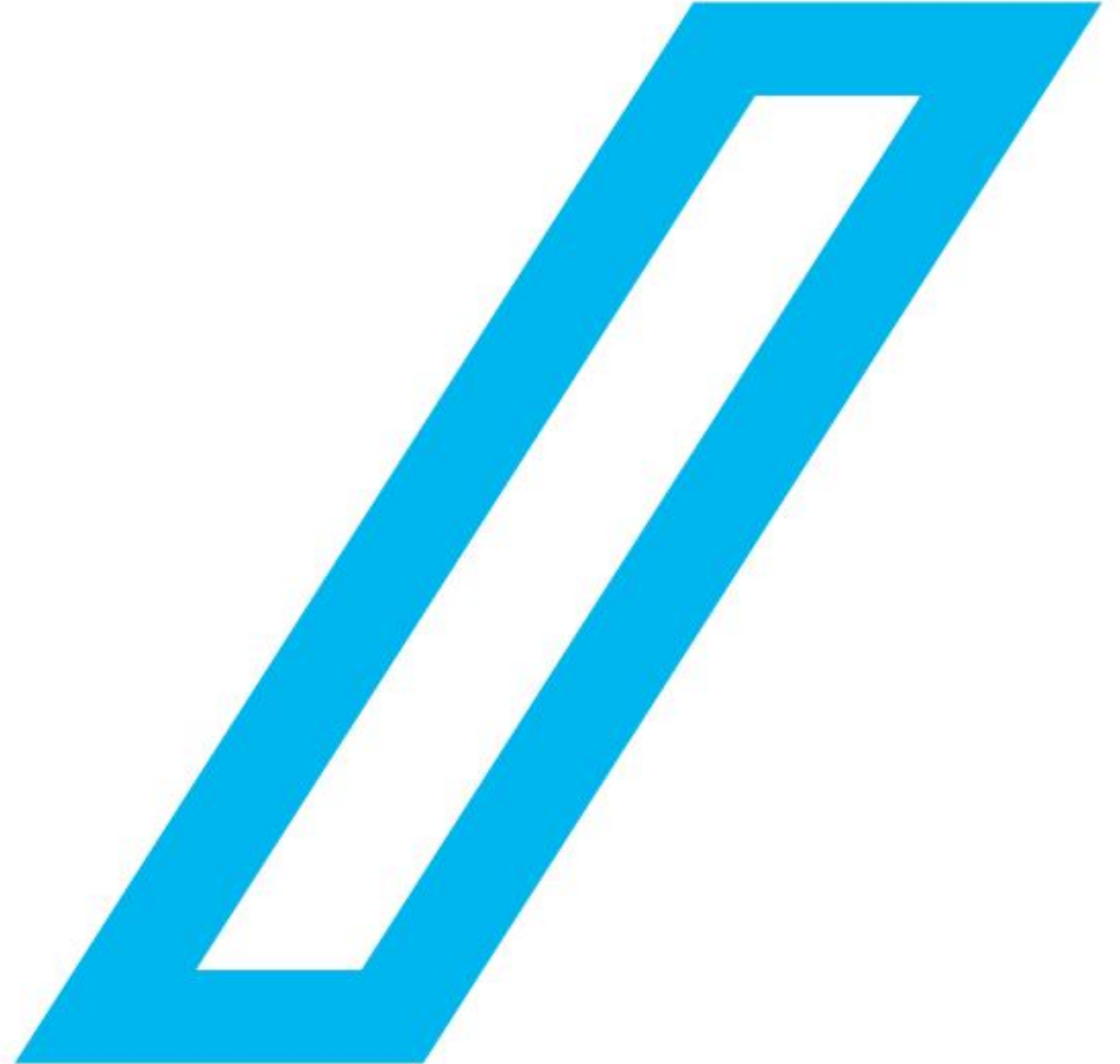


PID Control

Lecturer : Seungmok Song



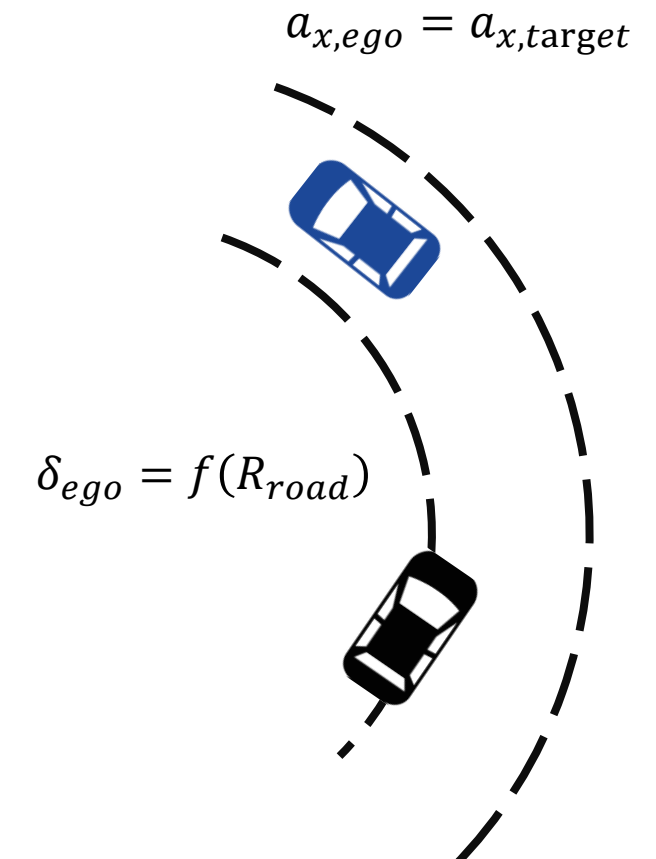
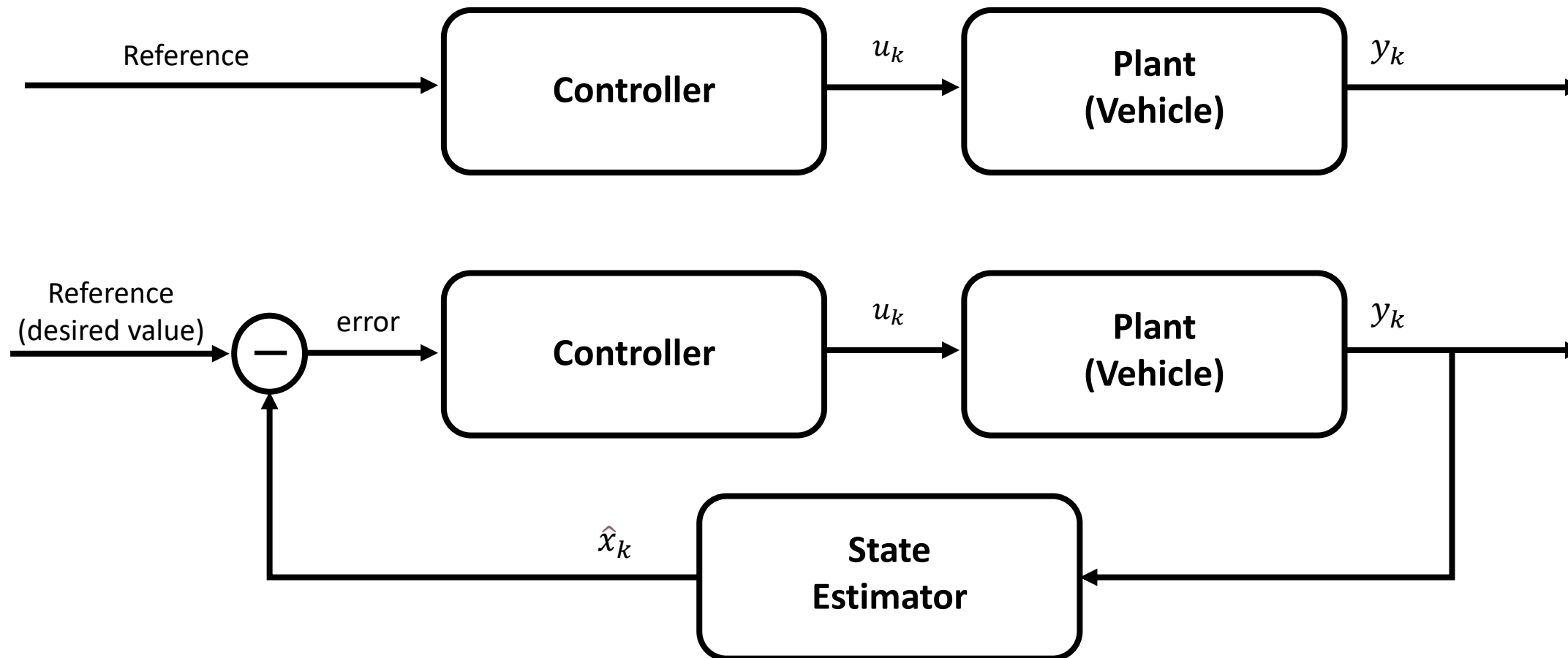
Contents

1. Introduction
2. P controller
3. PD controller
4. PID controller
5. Beyond PID controller



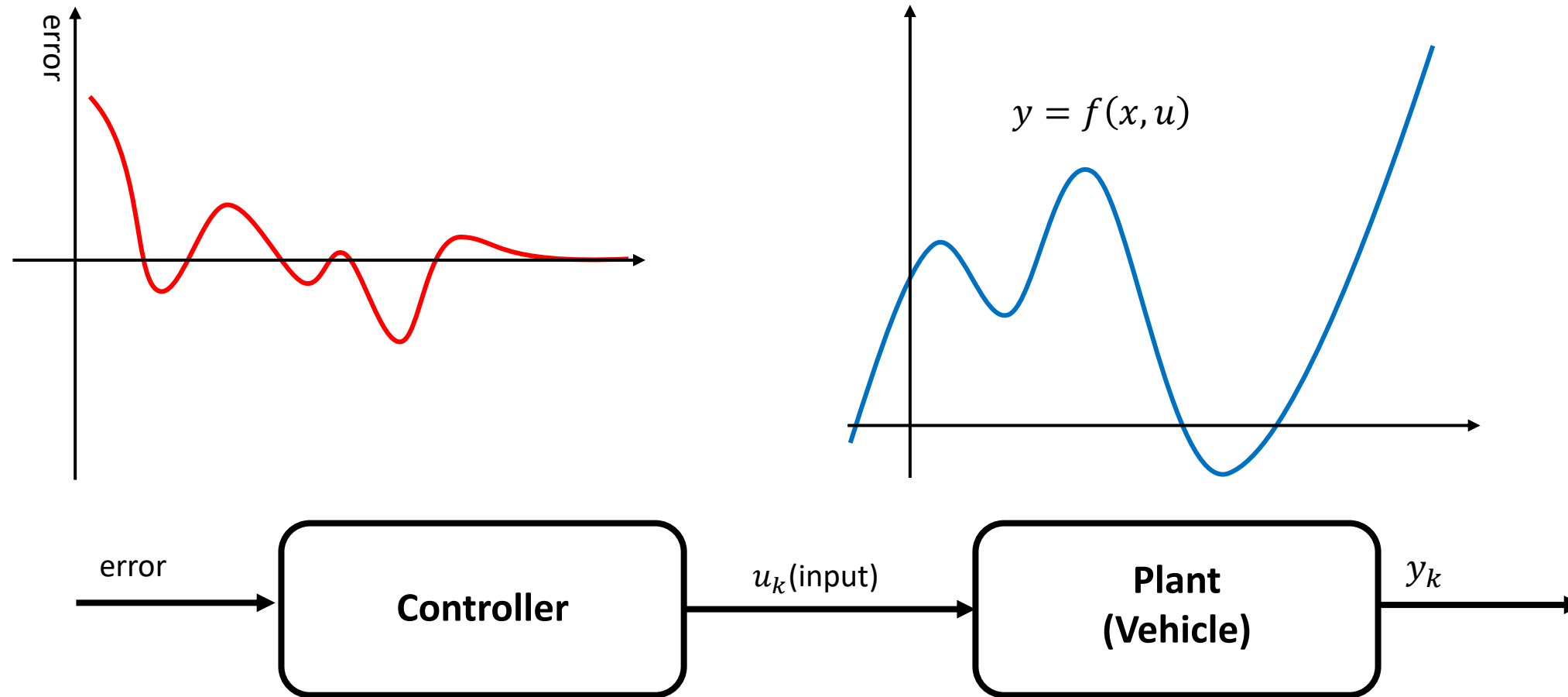
Introduction

- Feedback vs. Feedforward controller



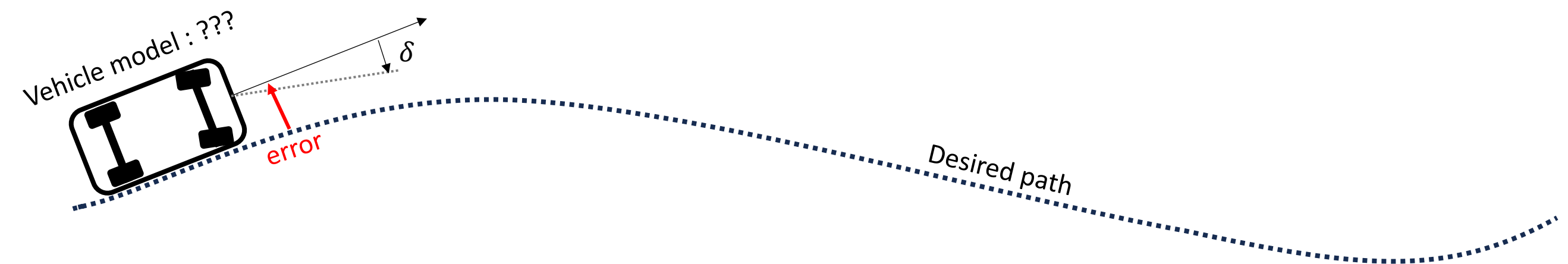
Introduction

- PID Controller
 - We don't have to care about the system model



Introduction

- Path tracking example

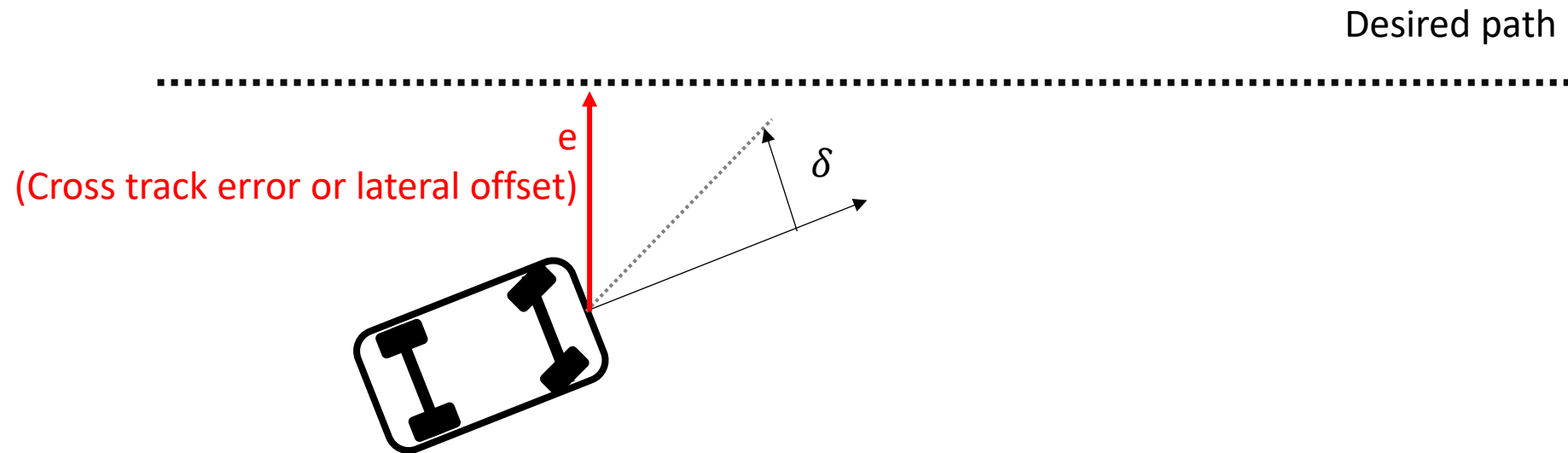


P Controller

- Error 에 비례한 control input 생성

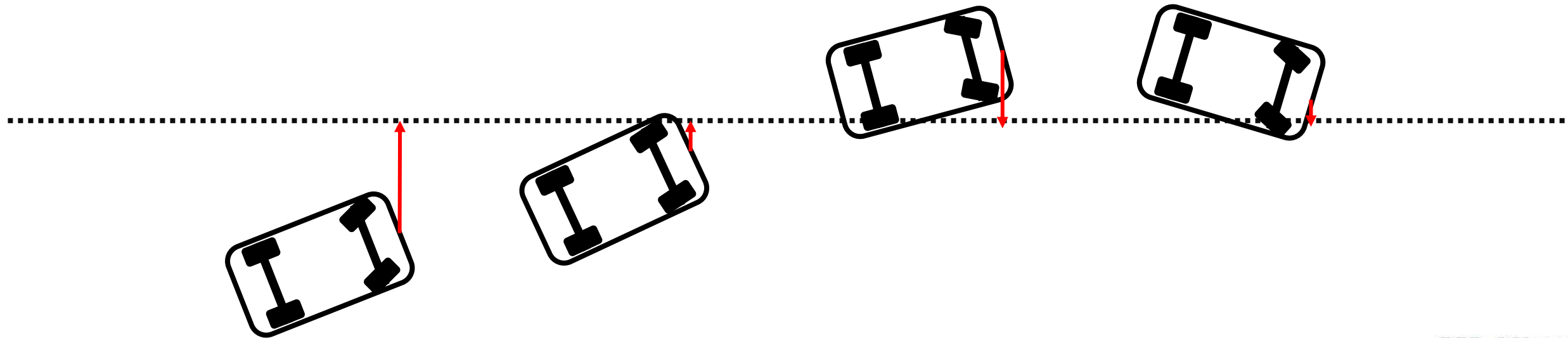
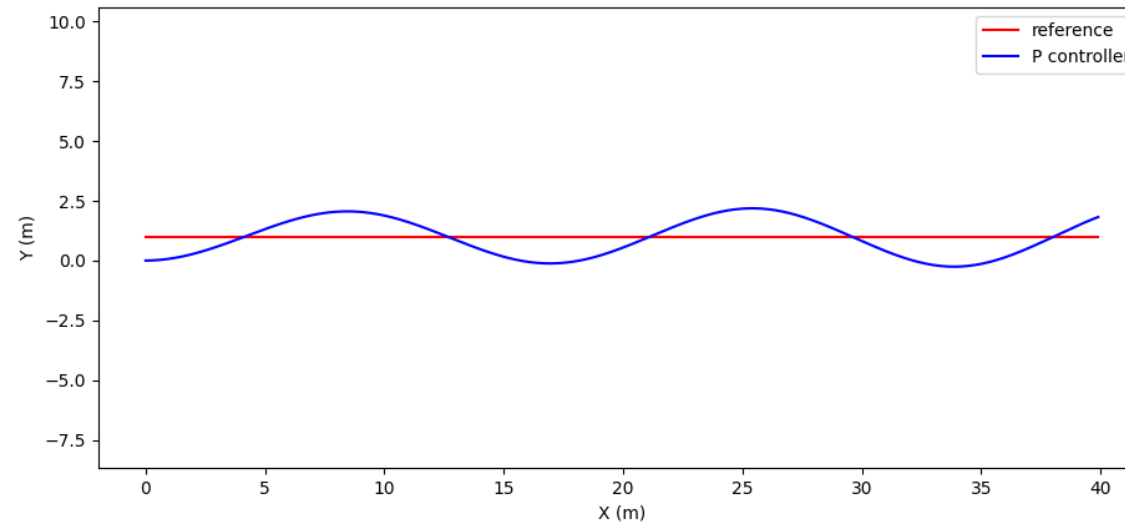
$$\text{input}(\delta) \propto \text{error}(e)$$

$$\delta_p = K_p e$$



P Controller

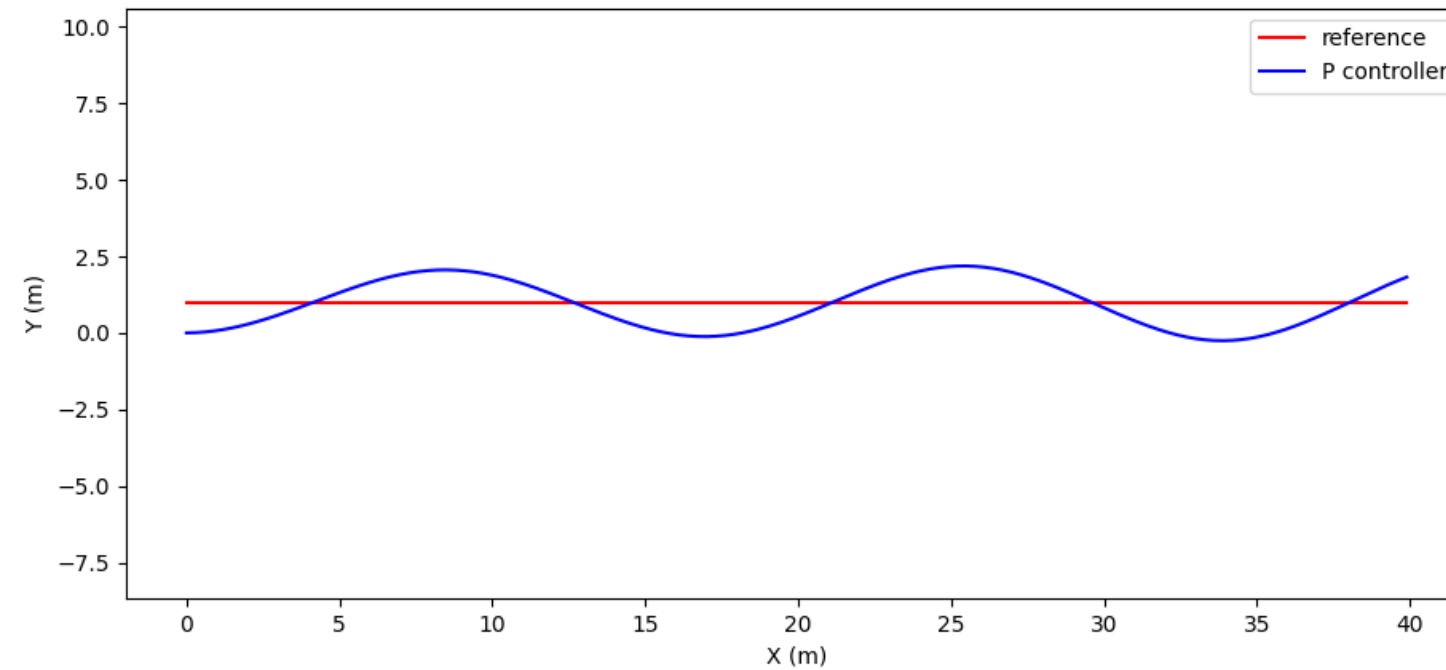
- Result
 - Overshoot



PD Controller

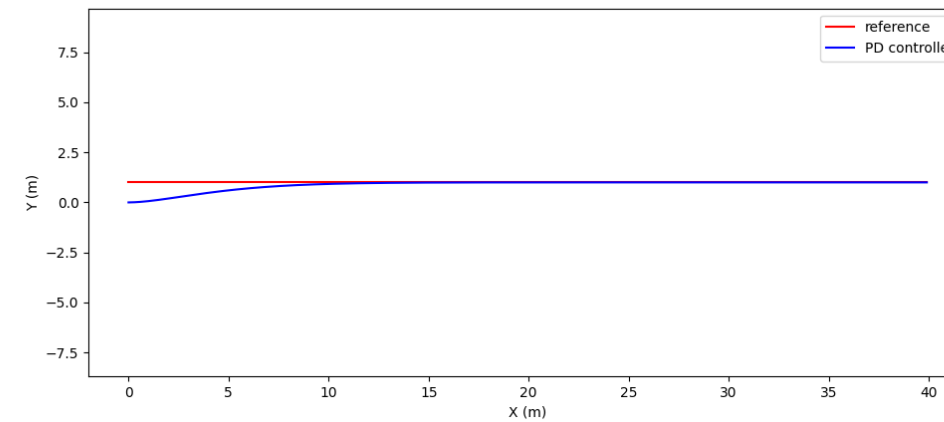
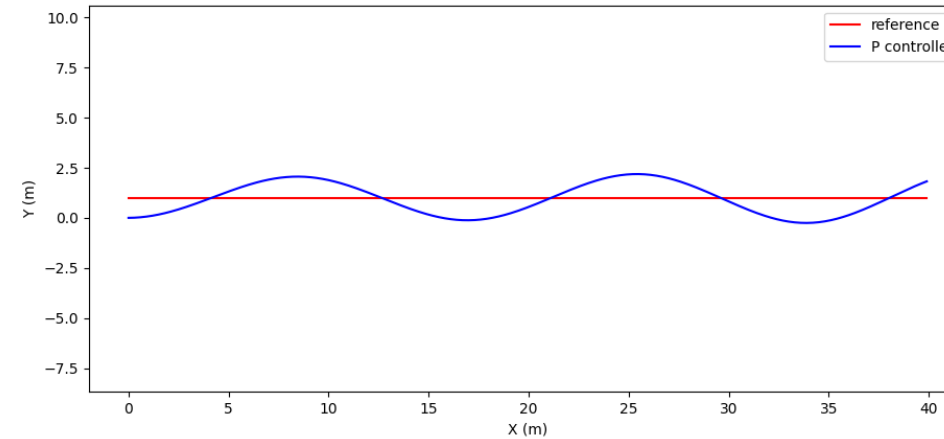
- Derivative term
 - Reducing overshoot

$$\delta_{pd} = K_p e + K_d \frac{d}{dt}(e)$$



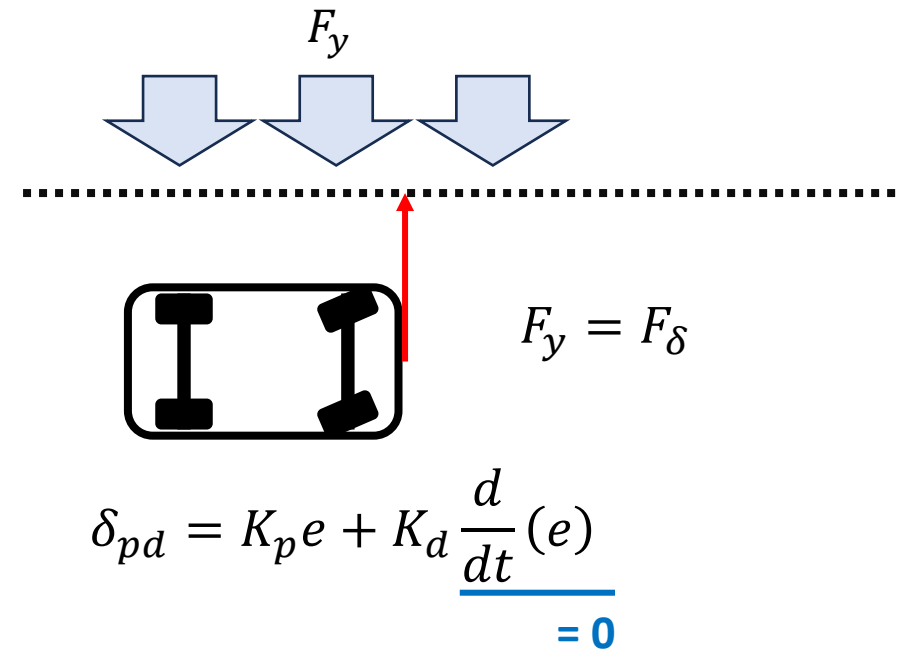
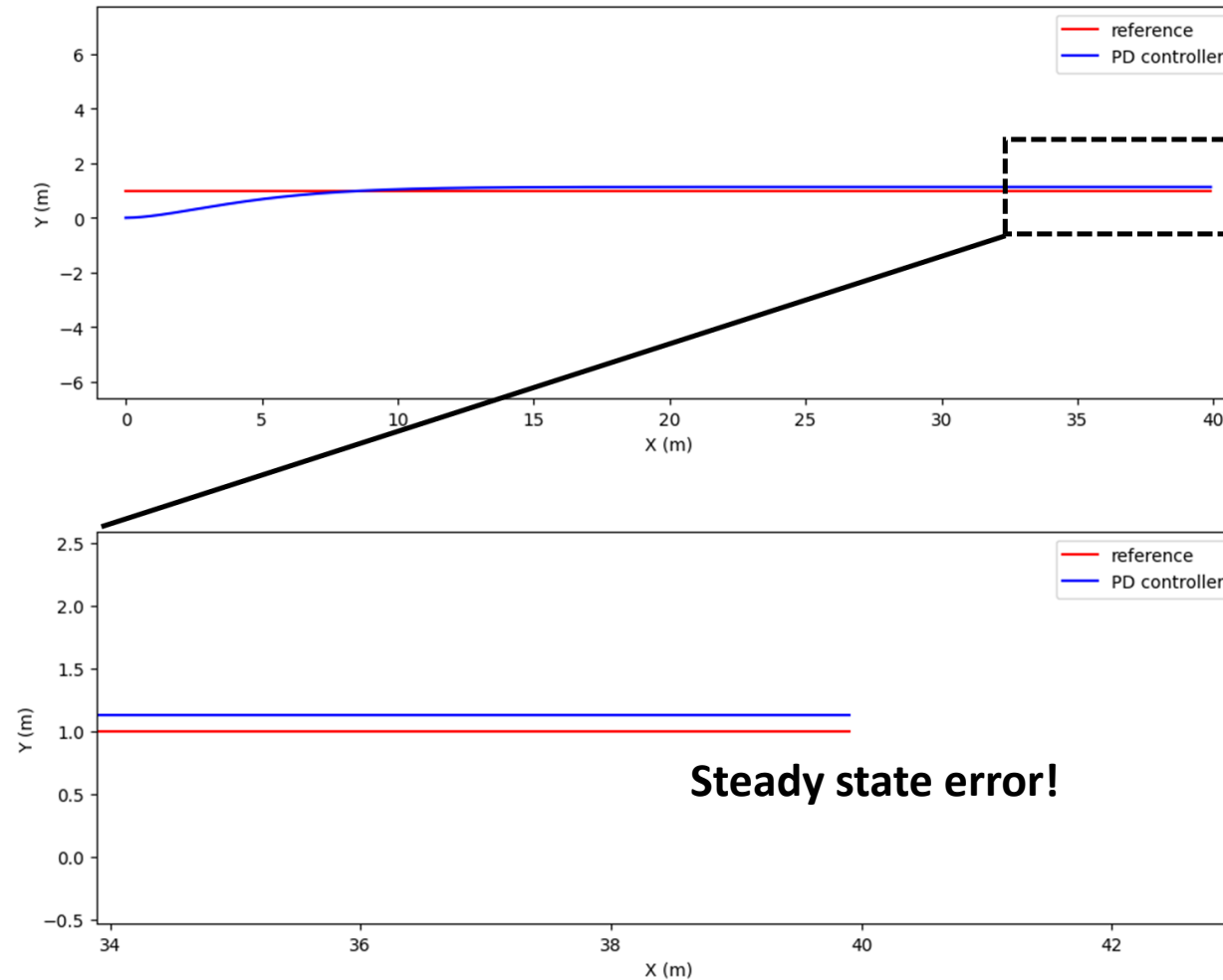
PD Controller

- Result



PD Controller

- Result
 - Bias

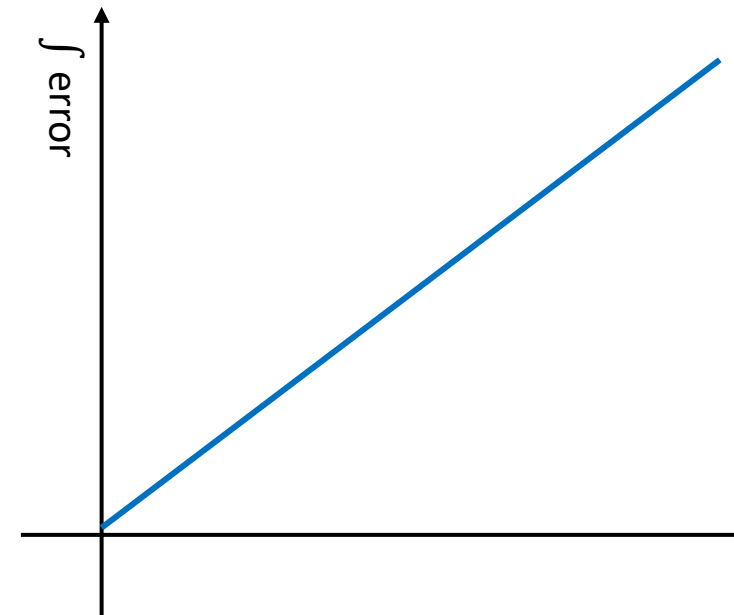
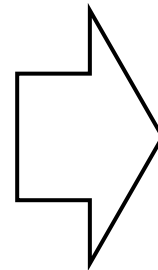
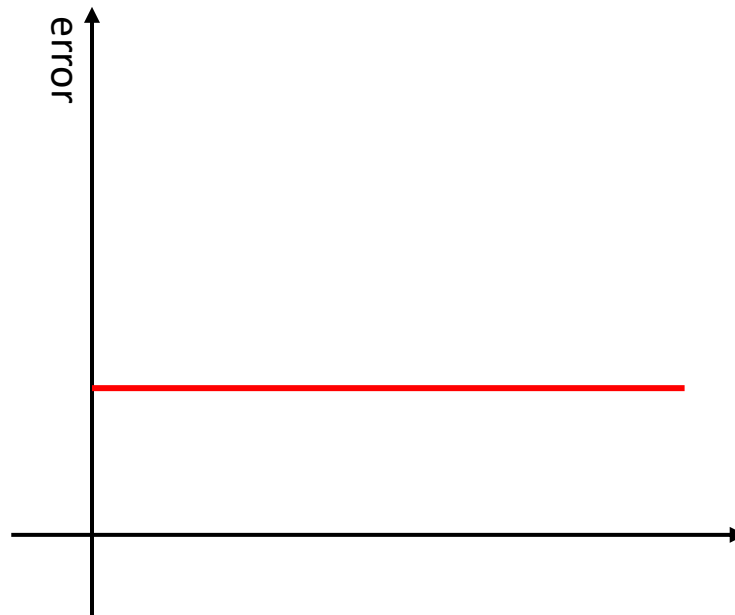


PID Controller

- Integral term
 - Reducing steady state error

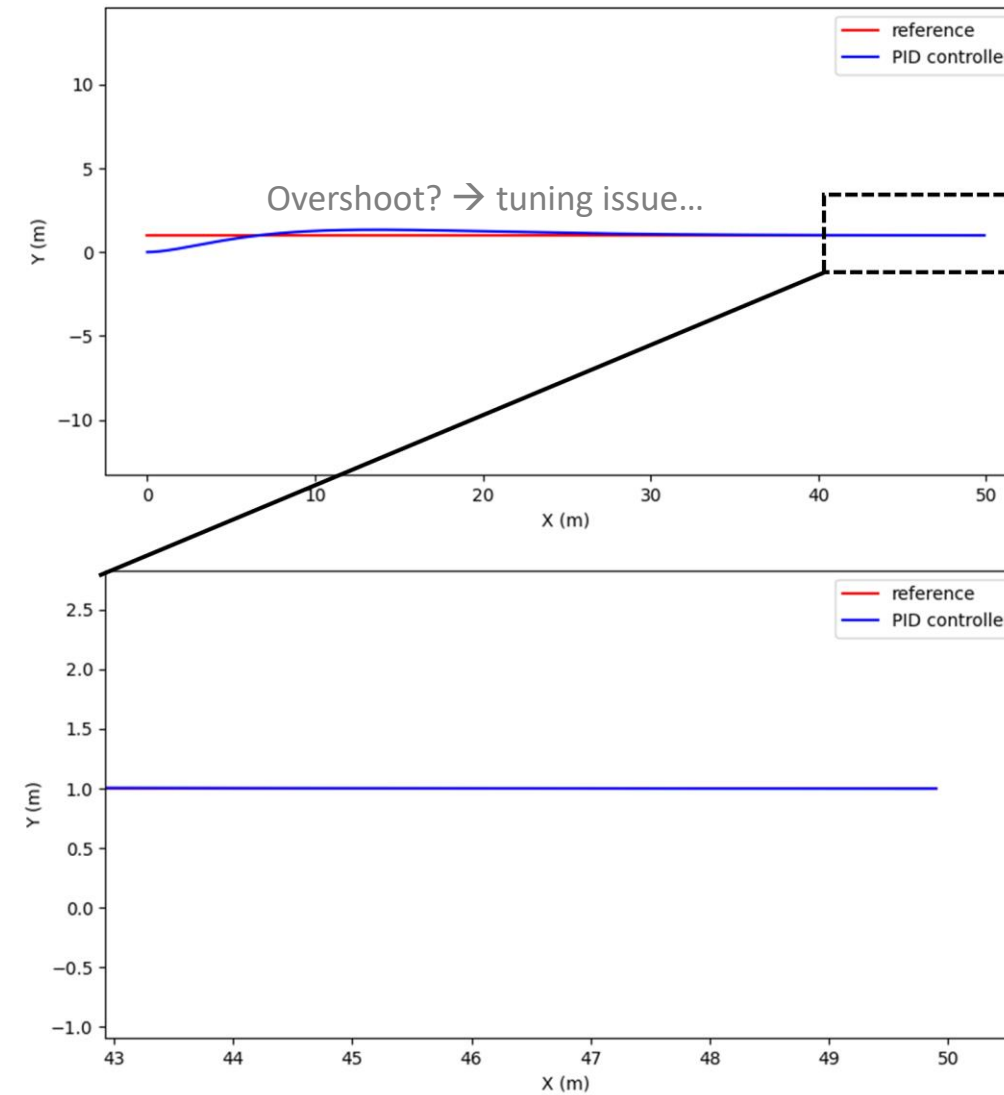
$$\delta_{pd} = K_p e + K_d \frac{d}{dt}(e) + K_i \int (e) dt$$

Error 가 존재하는 시간이 길어질 수록 커짐!



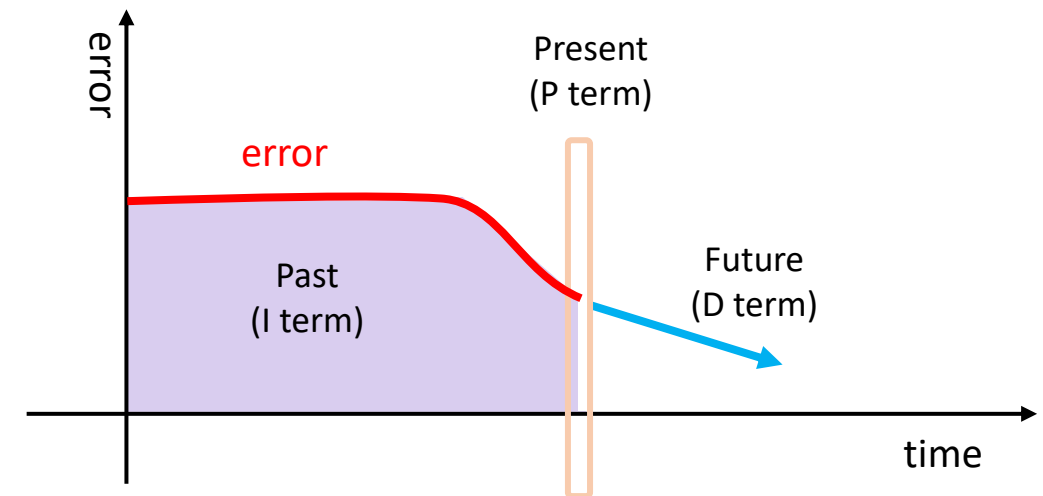
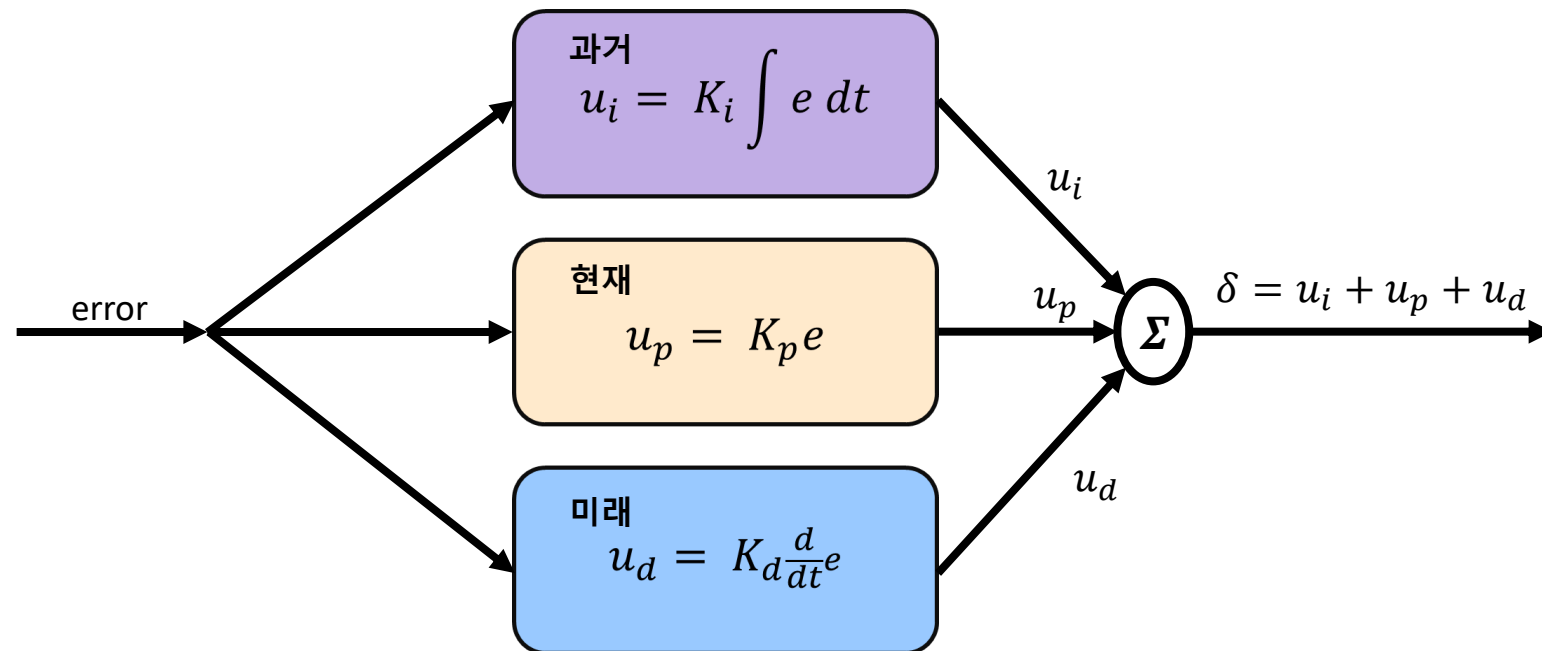
PID Controller

- Result



PID Controller

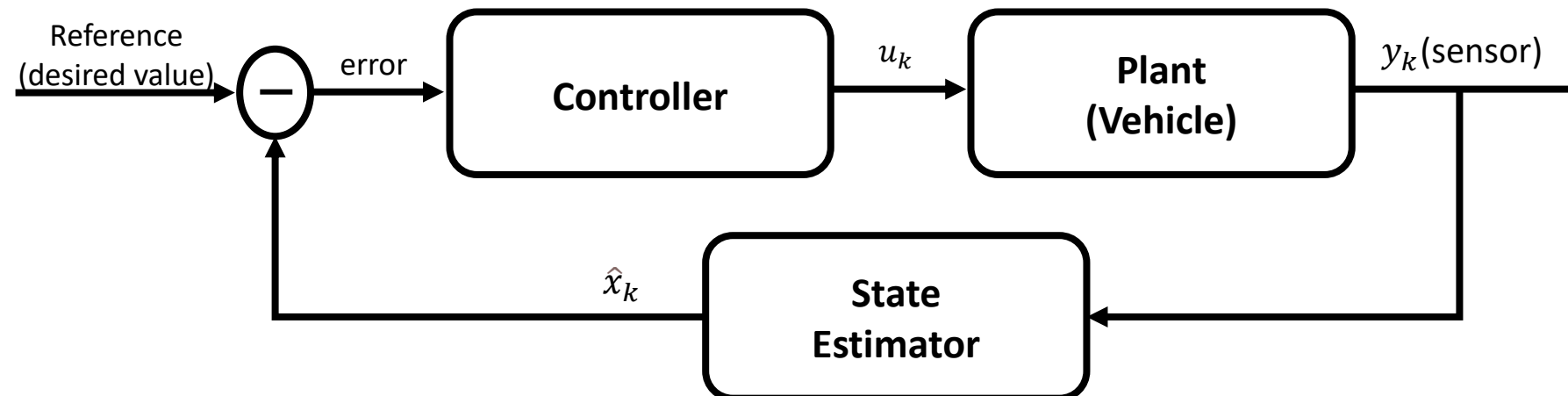
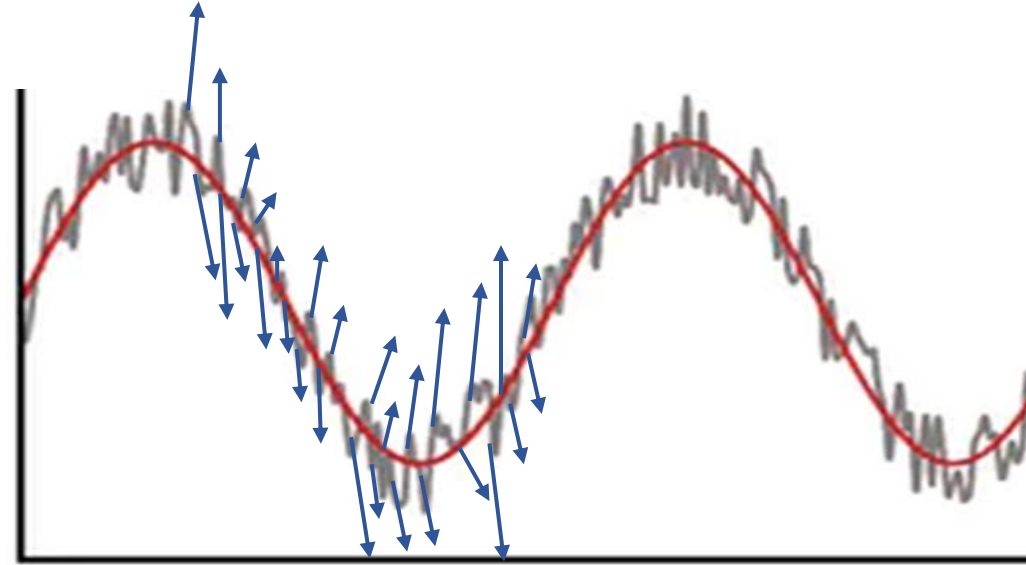
- Meanings of each controller



https://www.youtube.com/watch?v=1nJ79wX5EDM&ab_channel=%EB%A9%8D%EC%87%BC%EC%B8%A0

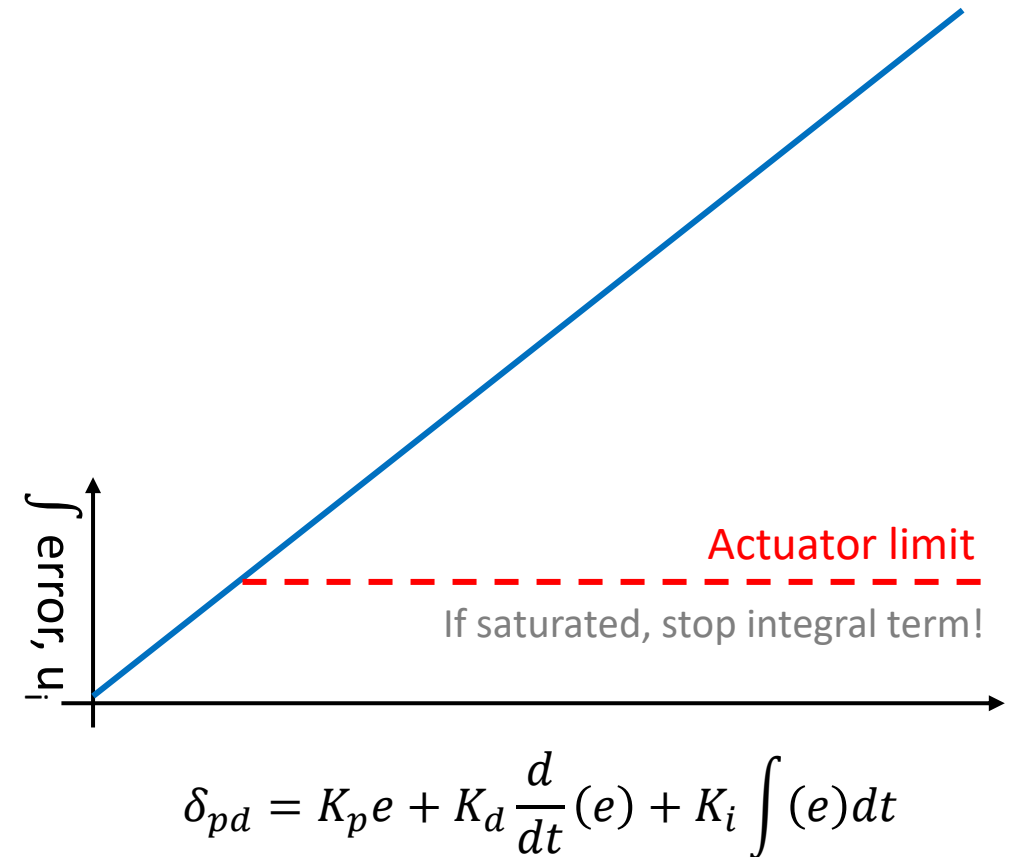
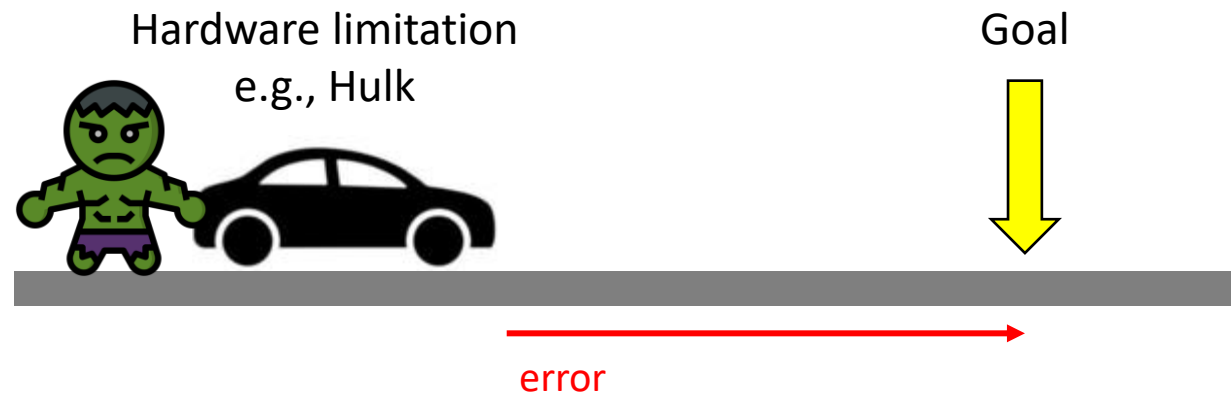
Beyond PID controller

- State estimator
 - Proper filter required!



Beyond PID controller

- Integrator anti-windup
 - Error 적분이 해소될 수 없을 때 적분 term 은 점점 커짐



Beyond PID controller

- Control engineering

- 제어공학에서는 Plant와 PID controller가 수식으로 주어지면 input에 대한 system의 response를 계산하는 방법도 배웁니다
- Laplace transform을 아는 것이 필수
- 한국어 lecture “제어공학 뽀개기 (99%의 확률로 내가 부서짐)”이 유튜브에 공개되어 있으니 제어공학이 궁금하신 분들은 한번 들어 보시는 것도 좋겠습니다.
- https://youtu.be/pVjKo_OVhU4 (강추)

$- D(s) = \frac{U(s)}{E(s)} = K_p + K_I/s$

✓ 1st-order system에 PI Controller 달고 Pole 내맘대로 움직일 수 있는지 확인해보자!

Block diagram: $R(s) \rightarrow \oplus \rightarrow \Sigma \rightarrow E(s) \rightarrow [K_p + \frac{K_I}{s}] \rightarrow U(s) \rightarrow [\frac{A}{(z s + 1)}] \rightarrow Y$. Feedback: $Y \rightarrow \ominus \rightarrow \Sigma$. Steady-state error: $\text{P: } s \cdot s \cdot E \rightarrow 0$.

CE: $1 + G(s)D(s) = 1 + (\frac{A}{zs+1})(K_p + \frac{K_I}{s}) = 0$ I: stable X

$\rightarrow z s^2 + (A K_p + 1) s + K_I A = 0$

$\rightarrow s^2 + \frac{(A K_p + 1)}{z} s + \frac{K_I A}{z} = 0$ $D(s)G(s) = \frac{(K_p s + K_I)A}{s(z s + 1)}$ system type 1

$\Rightarrow \omega_n = \sqrt{\frac{K_I A}{z}}, \quad \zeta = \frac{K_p A + 1}{2 z \omega_n}$

내맘대로 ω_n 와 ζ 를 조절하니까 ω_n 과 ζ 를 바꿀 수 있다.
즉, transient state의 time response 완벽하게 통제할 수 있음.

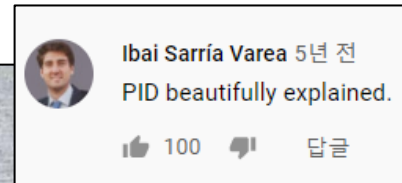
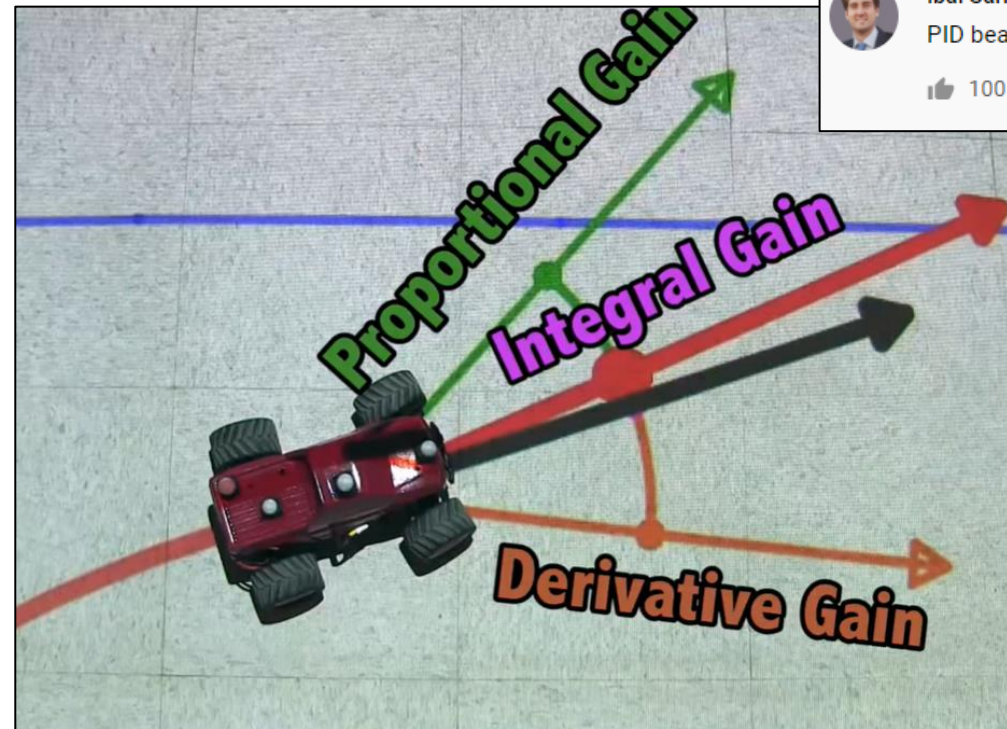
- PI Controller의 한계점
 - 2nd-order system의 transient state의 time response를

Beyond PID controller

- Controlling Self Driving Cars (Video)

- Aerospace Controls Lab (MIT)

<https://youtu.be/4Y7zG48uHRo>



Thank You

