Control Methods

YORK COLLEGE & UNIVERSITY CENTRE

Refresher

- **Error:** The difference between the desired value (setpoint) and the actual measured value of a system.
- **Open-loop control:** A control system that operates without feedback, so it does not correct its output if conditions change.
- **Closed-loop control:** A control system that uses feedback to continuously compare output to the setpoint and adjust its response to reduce error.
- **Steady-state error:** The remaining difference between the setpoint and the system output after the system has settled and is no longer changing.

Bang-Bang Control (On–Off Control)

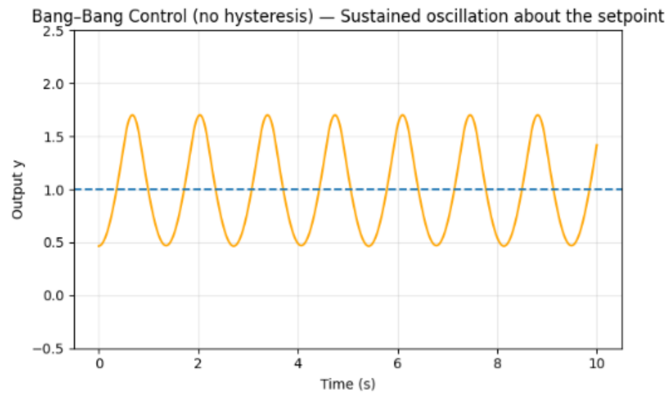
- Bang-bang control is a **very simple** control method where the output is either fully on or fully off with no **intermediate levels**

- The controller switches state depending on whether the measured value is **above or below the setpoint**

- This often causes **oscillation around the setpoint** as seen on the graph

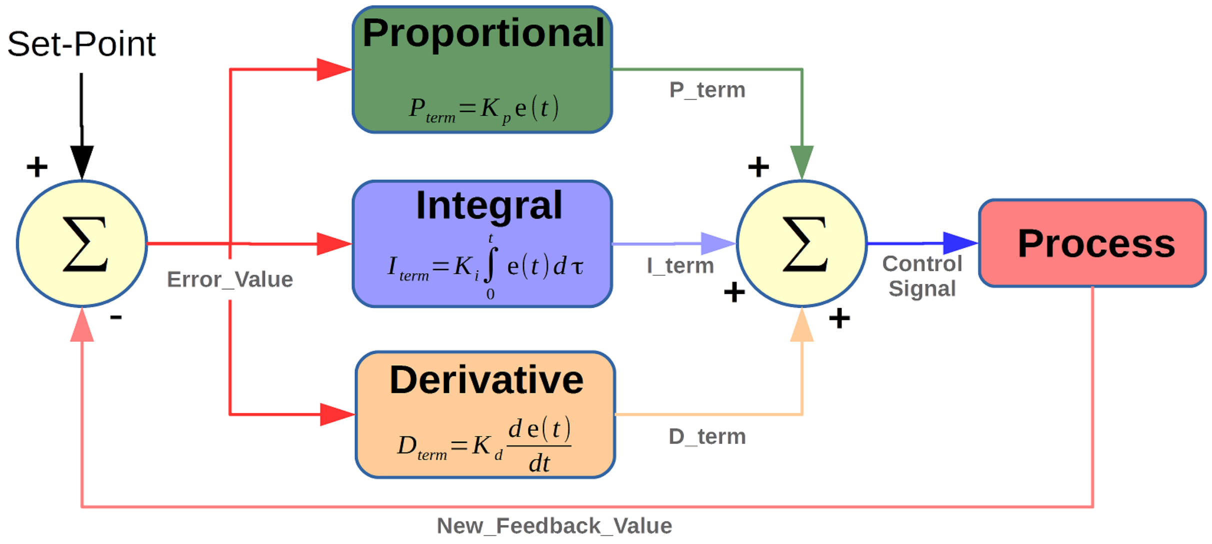
- Bang-bang control is **cheap**, **robust**, and **easy to implement**, but not very precise.

- Common applications include **thermostats**, **basic heaters**, **refrigerators**, and **float switches**.



What is a PID Control

- PID control is a **closed loop** feedback mechanism commonly used in automation and robotics
- The system continuously adjusts the output to optimise the system, respond to changes and minimise error
- Calculates an output based on the present, past, and predicted future error.

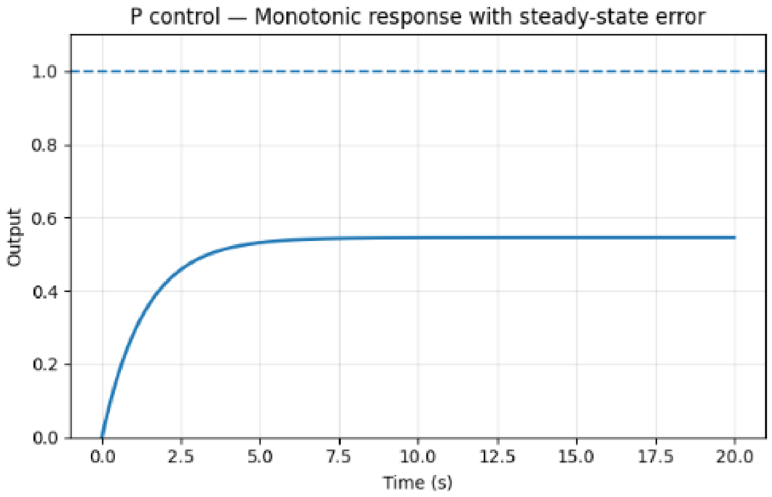


What does PID stand for?

- Proportional – Looking at the present
- Integral – Looking at the past
- Derivative – Looking at the future

Proportional Control (P)

- Proportional control means that we adjust the output depending on how large our current error is
- **The bigger our error is, the bigger our correction is**
- **Increasing** proportional gain makes the system **respond faster.**
- Too much proportional gain can cause **oscillation** or **instability**.
- Proportional control alone often leaves a **steady-state error**.
- It is simple, fast, and commonly used as the base of PID control systems.



Proportional Control (P) Equation

$$u(t) = K_p e(t)$$

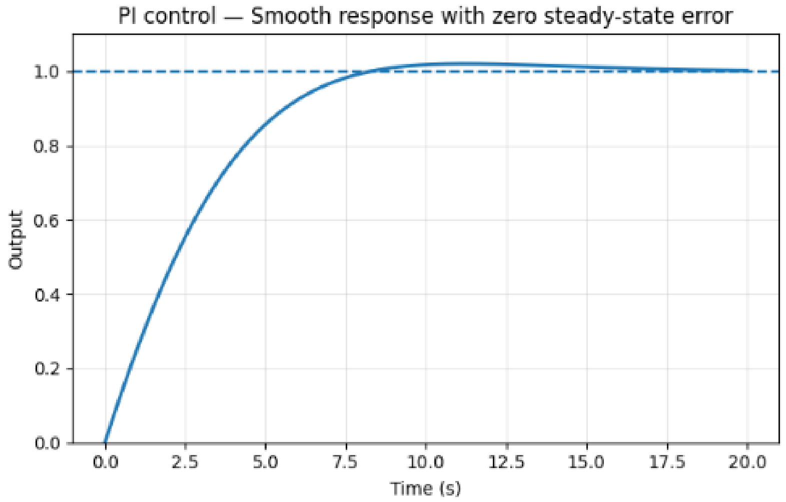
Proportional Output

Proportional Value (gain)

Error Value (setpoint - current)

Integral Control (I)

- Integral control responds to the accumulated error over time, not just the current error.
- **It increases the output while error persists, until the error is removed.**
- Integral control is used to **eliminate steady-state error** left by proportional control.
- Too much integral action can cause **overshoot** and **slow system response.**
- It can also lead to **integral wind-up** if the system cannot respond quickly enough.
- Integral control improves accuracy but must be carefully tuned.



Proportional Control (P) Equation

$$I(t) = K_i \int_0^t e(\tau)d\tau.$$

Integral Output

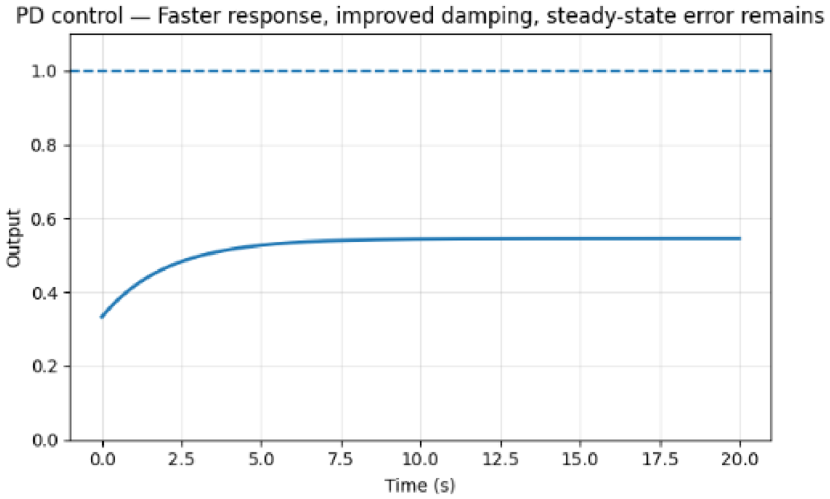
Integral Value

Error Value (setpoint - current)

Change in time

Derivative Control (D)

- Derivative control responds to the **rate of change of the error** rather than the error itself.
- **It predicts future system behaviour by observing how quickly the error is changing.**
- Derivative action helps **reduce overshoot and oscillation**.
- It improves system **stability and damping.**
- Derivative control is **sensitive to noise** in sensor signals.
- It is rarely used on its own and is typically combined with P and I in PID control.



Proportional Control (P) Equation

$$D = K_d \frac{de(t)}{dt}.$$

Derivative Output

Derivative Value

Change in time

Change in Error Value (setpoint - current)

PID Equation

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)\,d\tau + K_d \frac{de(t)}{dt}$$

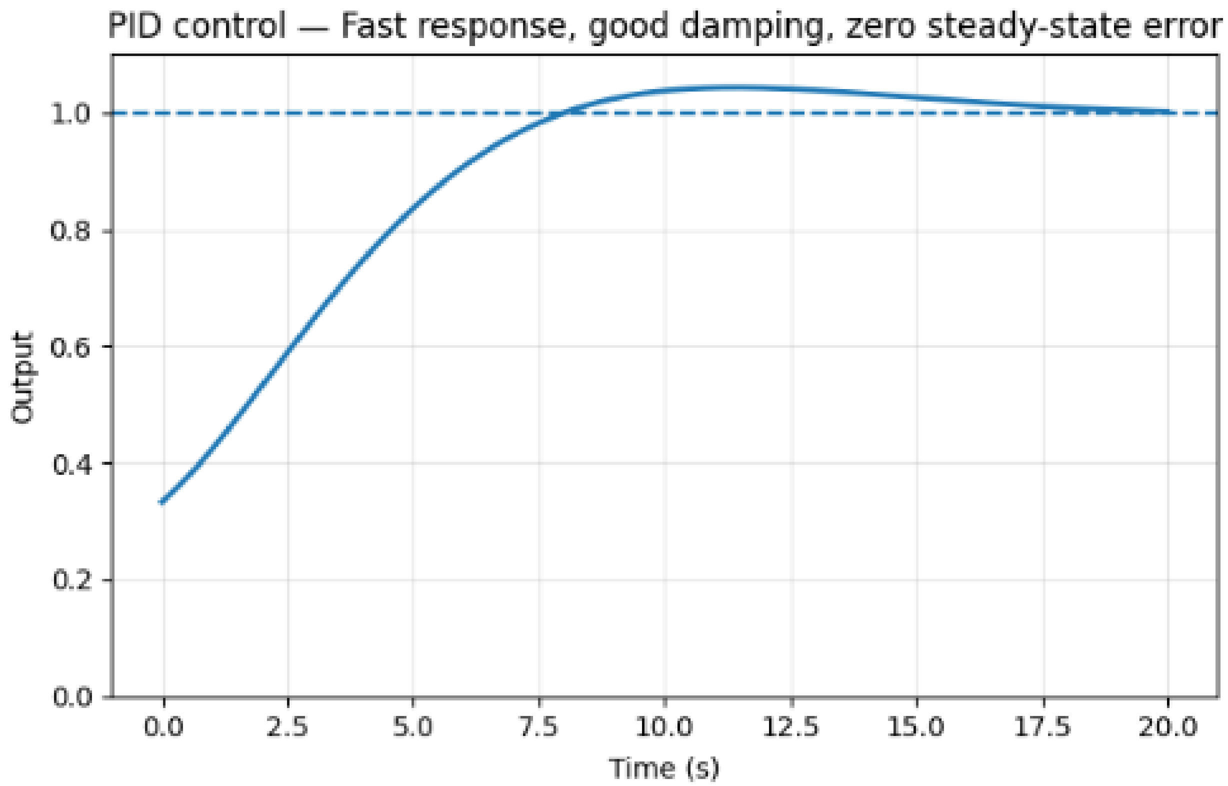
System output

Proportional Equation

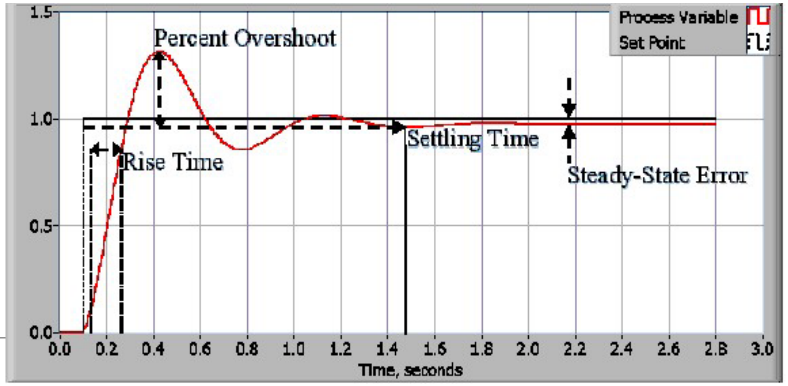
Integral Equation

Derivative Equation

PID Response Graph



Tuning PID



- **PID tuning** is the process of **adjusting the P, I, and D** gains to achieve stable and accurate control.
- The aim is to **minimise steady-state error**, **overshoot**, and **settling time**.
- Increasing **P** makes the system **respond faster** but can cause **oscillation**.
- Increasing **I** removes **steady-state error** but can increase **overshoot** and **slow response**.
- Increasing **D** reduces **overshoot** and improves **stability** but can amplify noise.
- Tuning is usually done by **trial and error**, starting with P, then adding I, then D.
- Poor tuning can result in **oscillation**, **slow response**, or **instability**.