Durham University

COMP52815 Robotics - Planning and Motion

Robotics –
Planning and Motion

COMP52815

Dr Junyan Hu & Prof Farshad Arvin

Email: junyan.hu@durham.ac.uk
Room: MCS 2060

Lecture: Learning Objectives

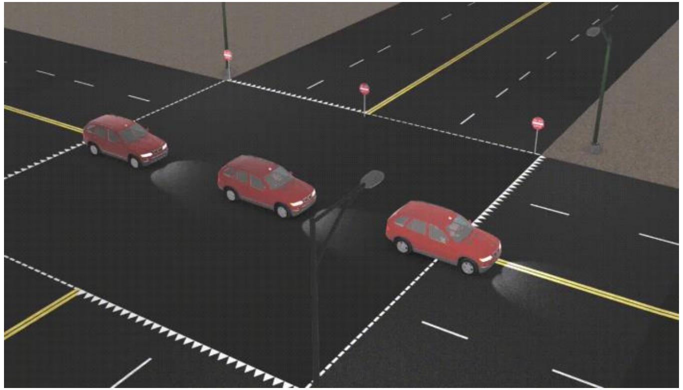
The aim of this lecture is to design a control system for dynamical systems.

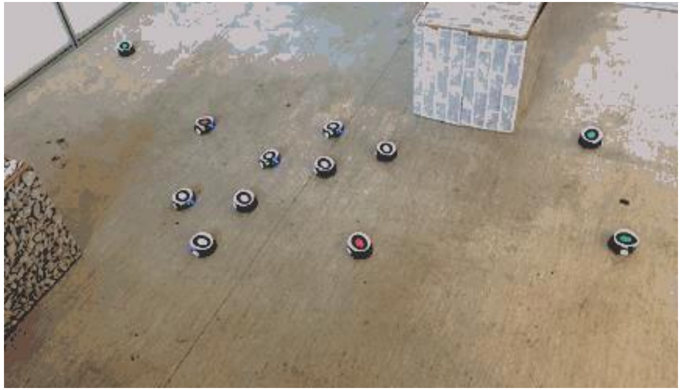
- Objectives:
  1. Feedback Systems
  2. Bang-Bang Control
  3. PID Control
  4. State-Space Representation
  5. Stability of the System

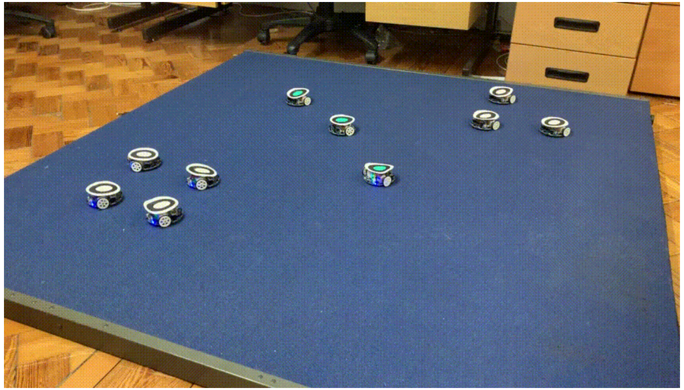
See also:

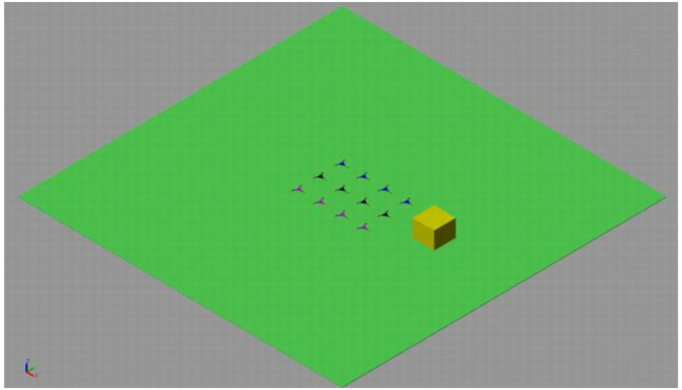
- Introduction to Mobile Robot Control, Spyros G. Tzafestas, 2014
- Feedback Systems, Karl Johan Aström, Richard M. Murray, 2009

Control of Mobile Robots









Durham University

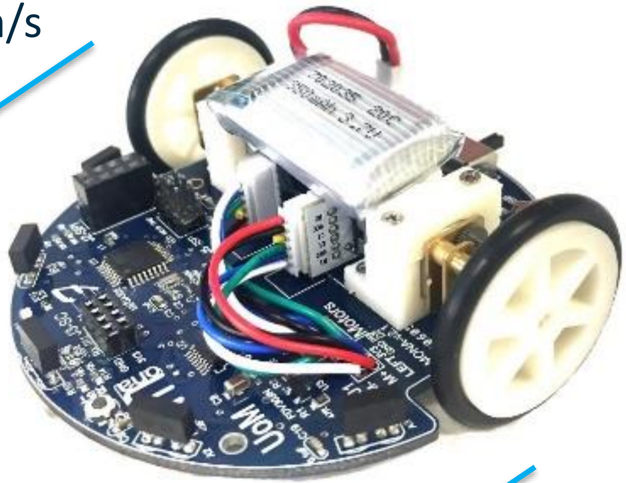
Robot Control

- **Robot control with (almost) no theory**
  - PID Controller
  - Differential drive robots

- **Control theory (State-space)**
  - Multiple inputs / Multiple outputs
  - Dynamics of internal states

Open-Loop Systems

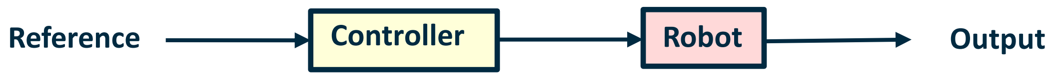
Right motor: 1 m/s



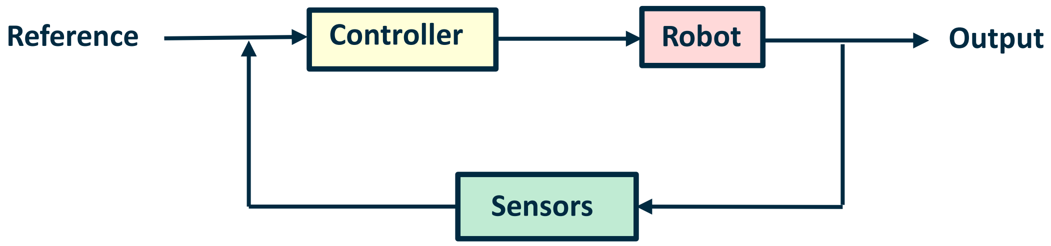
Left motor: 1 m/s

In reality, will the robot move in a straight line at 1 m/s?

Open-Loop VS Closed-Loop



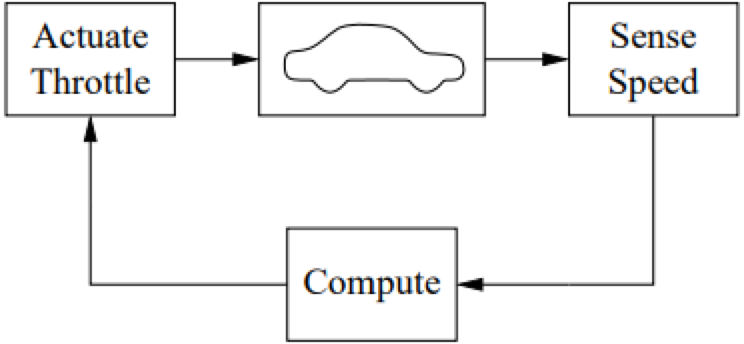
- Easy to implement
- Large tracking error
- Difficult to coordinate



- Accurate motion
- Possible to apply coordination algorithms
- Robust to disturbance
- More efforts in controller design and hardware implementation

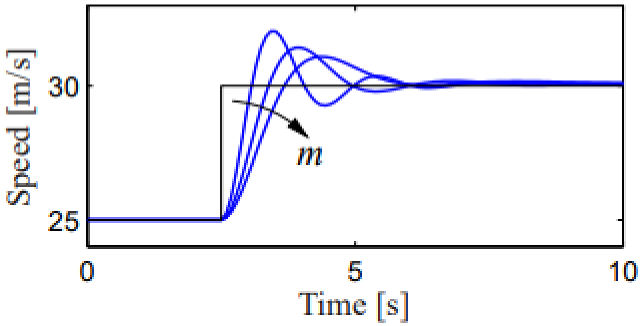
Durham University

Example



A feedback system for controlling the speed of a vehicle. In the block diagram, the speed of the vehicle is measured and compared to the desired speed within the "Compute" block. Based on the difference in the actual and desired speeds, the throttle (or brake) is used to modify the force applied to the vehicle by the engine, drivetrain and wheels

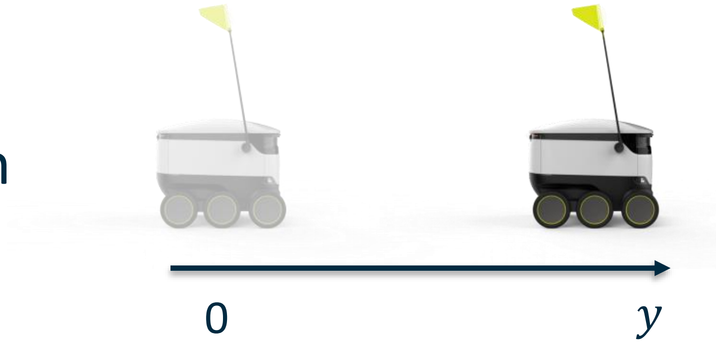
Example



The figure shows the response of the control system to a commanded change in speed from 25 m/s to 30 m/s. The three different curves correspond to differing masses of the vehicle, between 1000 and 3000 kg

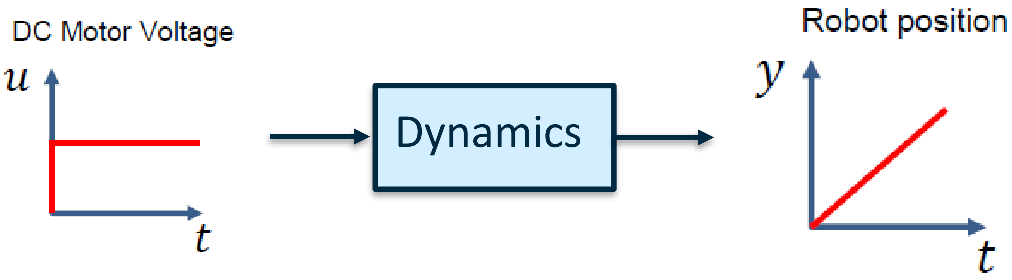
Durham University

Simple control system

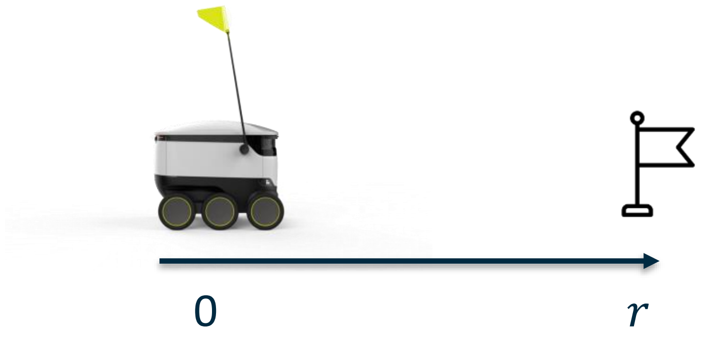


Mobile robot with 1-dimensional motion

- Single Input Single Output (SISO) system
- **Input** $[u]$: DC Motor voltage
- **Output** $[y]$: robot position

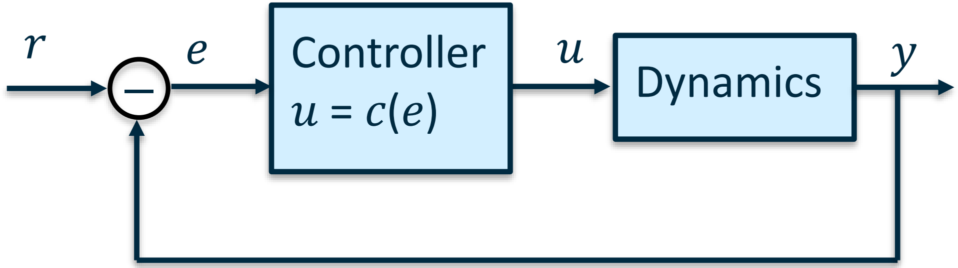


Simple control system



Move robot to position r

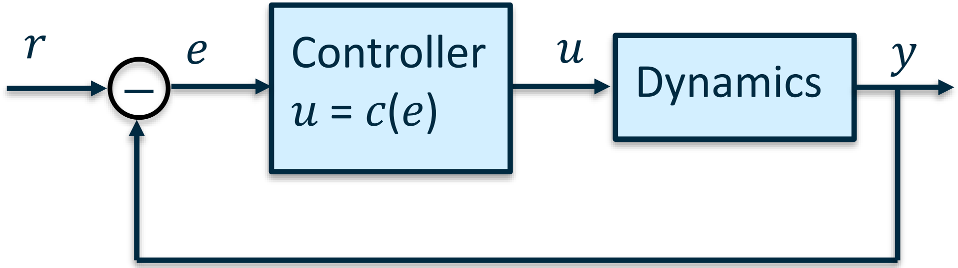
- **Reference** $[r]$: The desired value for the output
- **Error** $[e = r - y]$: Difference between desired and actual output.
- **Input** $[u = c(e)]$: Reacts to the error.



Durham University

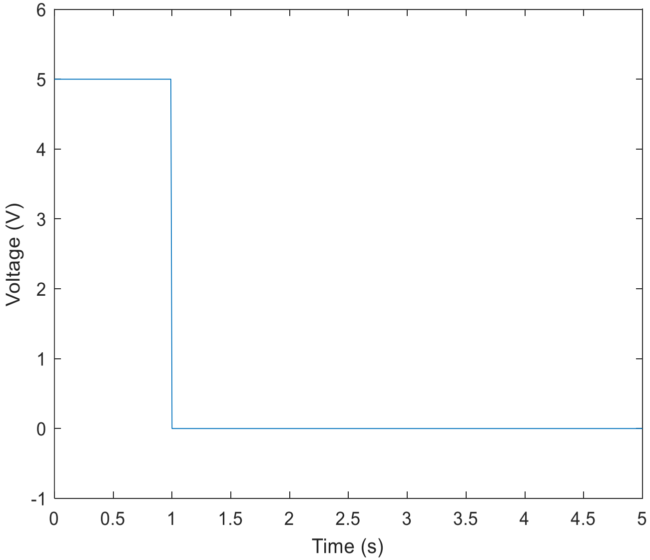
Bang-Bang Control

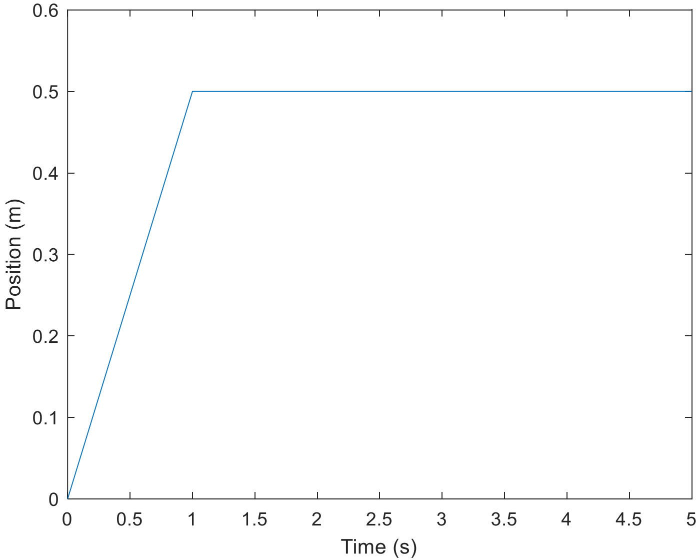
$$c(e) = \begin{cases} u = u_{max}, & e > \varepsilon \\ u = -u_{max}, & e < -\varepsilon \\ u = 0, & |e| \le \varepsilon \end{cases}$$



Simulation

$$u_{max} = 5\text{V} \quad r = 0.5\text{m} \quad \varepsilon = 1\text{mm}$$

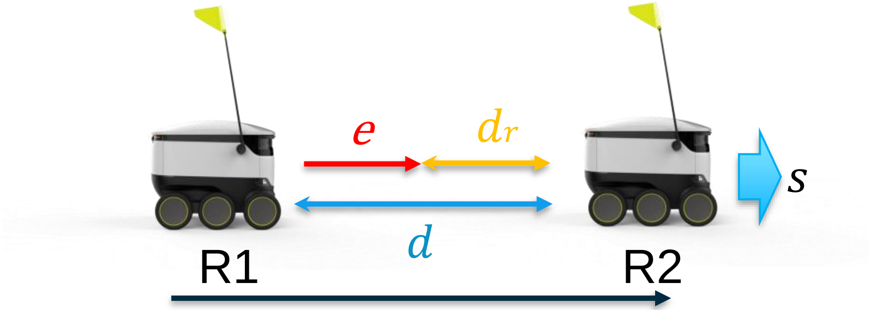




Following Another Robot

Control R1 to keep a constant distance d_r from R2.

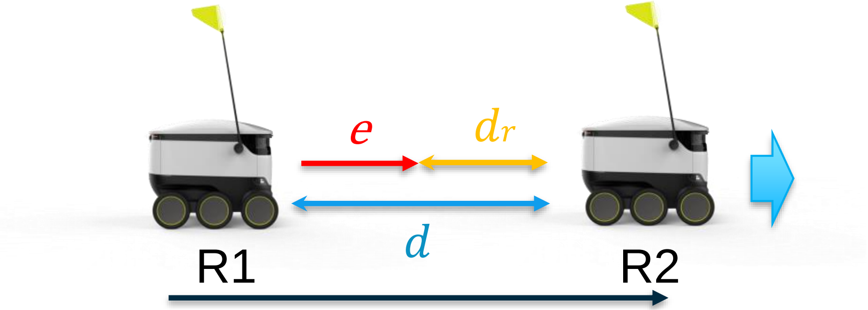
R2 moves at a constant speed s

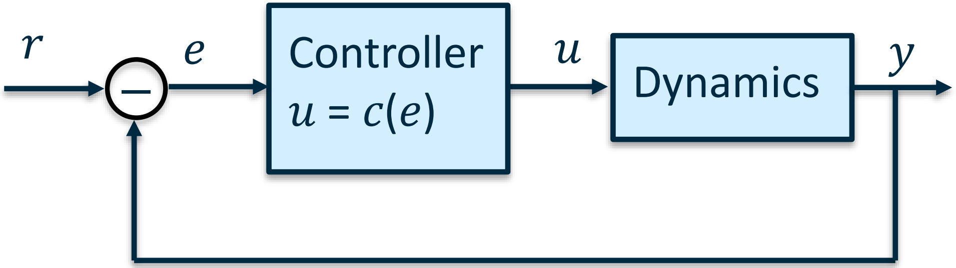


- **Input** $[u]$: DC Motor Voltage of R1
- **Output** $[y]$: position of R1
- **Error** $[e = d - d_r]$: distance to the desired position

Bang-Bang Control

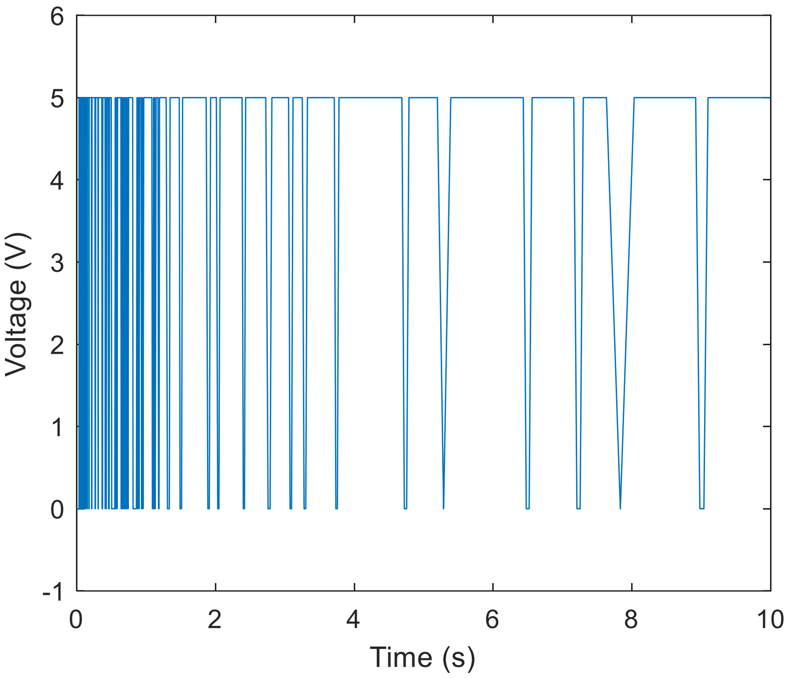
$$c(e) = \begin{cases} u = u_{max}, & e > \varepsilon \\ u = -u_{max}, & e < -\varepsilon \\ u = 0, & |e| \leq \varepsilon \end{cases}$$

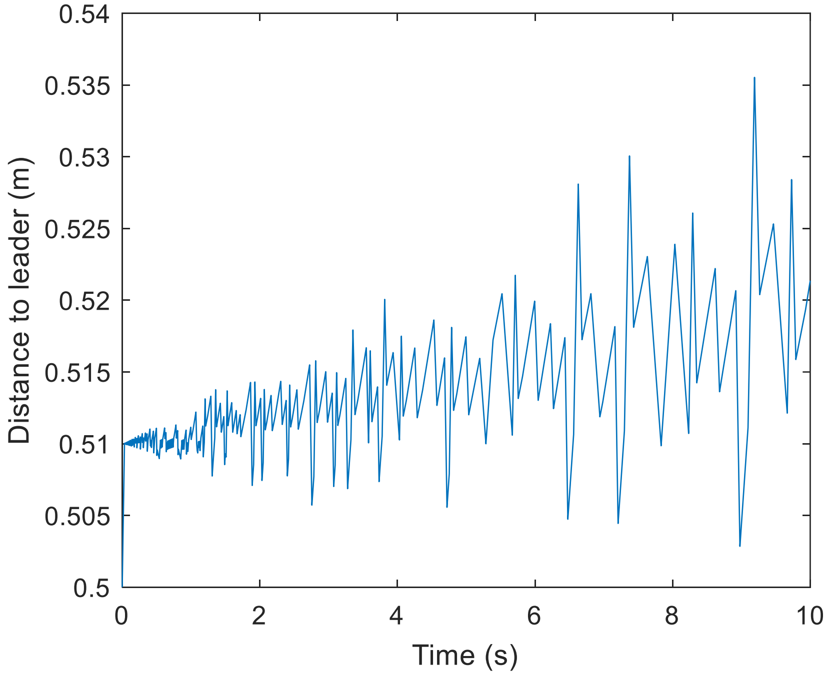




Simulation

$u_{max} = 5\text{V} \qquad d_r = 0.5\text{m} \quad \varepsilon = 1\text{mm} \qquad s = 0.3\text{m/s}$

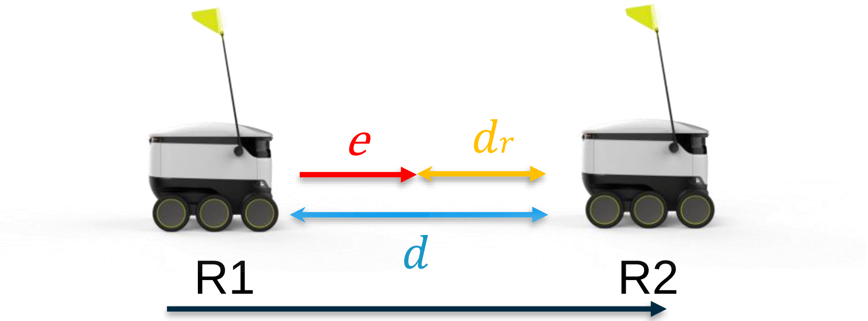


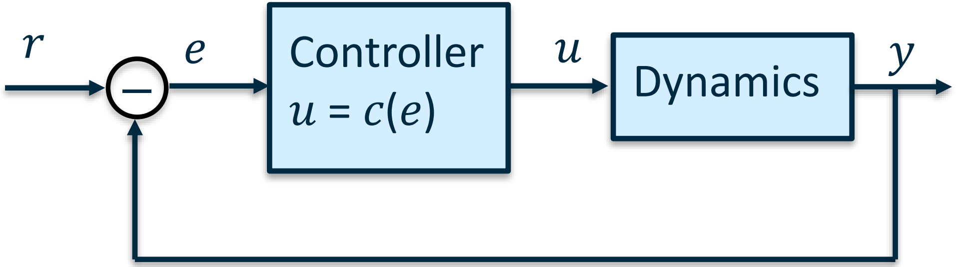


Can we make it smoother?

Proportional Control

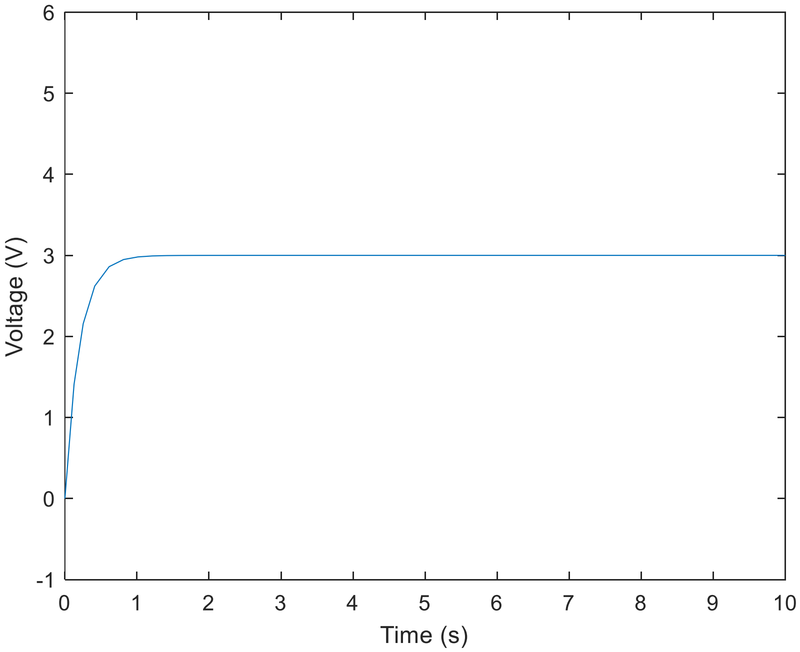
$$c(e) = K_p e$$

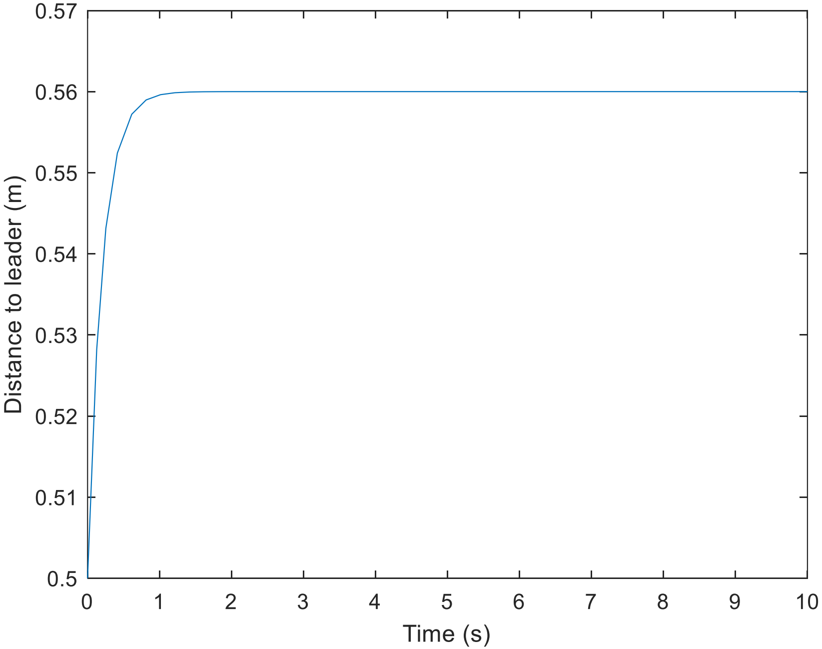




Proportional Control

$$K_p = 5$$



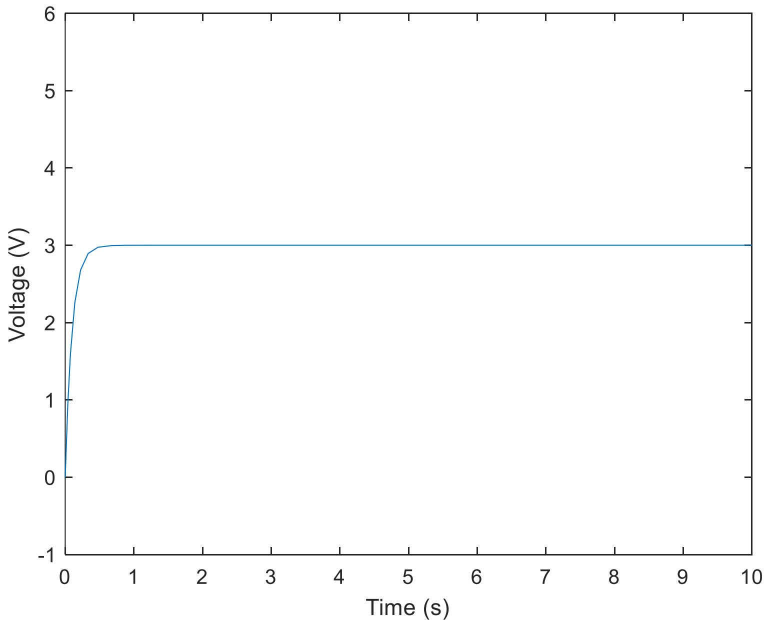


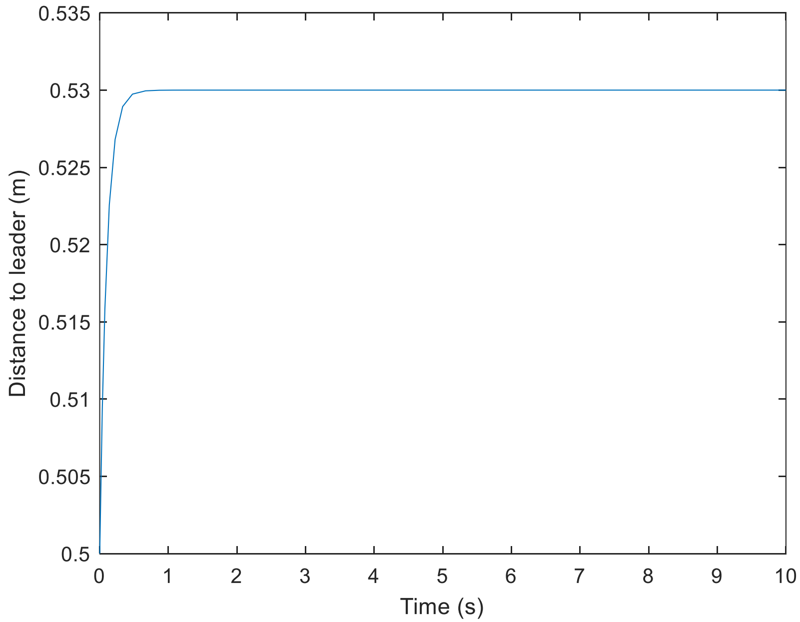
Durham University

The control is smoother now, but it does not converge to 0.5 !

Increase K_p

$$K_p = 10$$

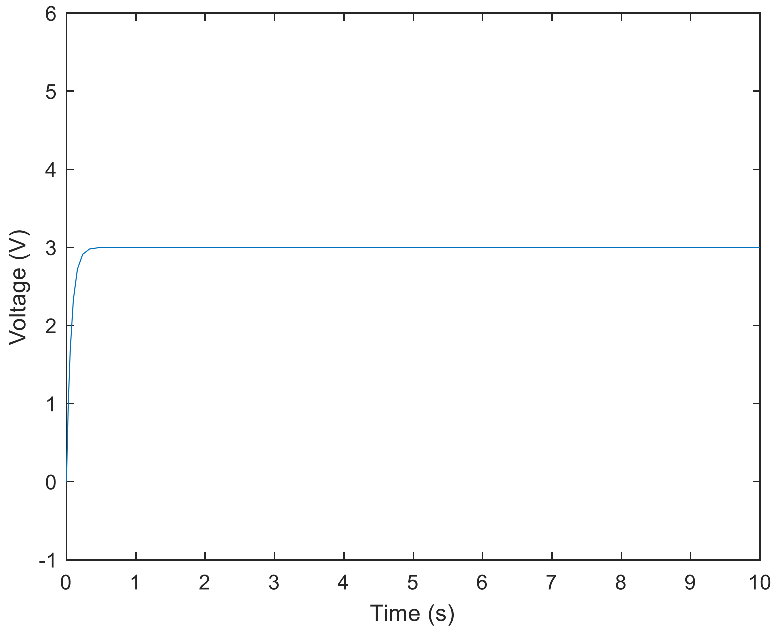


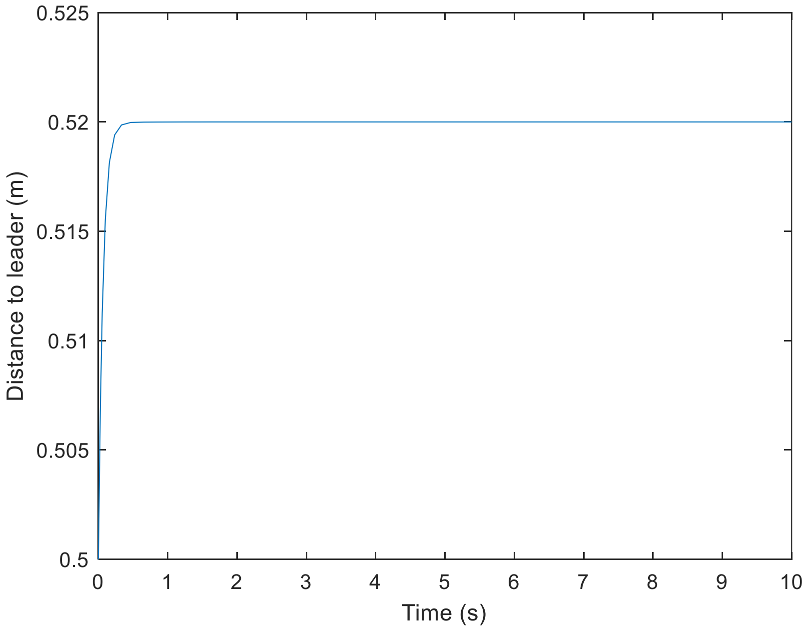


The output is now a little bit closer to 0.5 !

Increase K_p!

$$K_p = 15$$



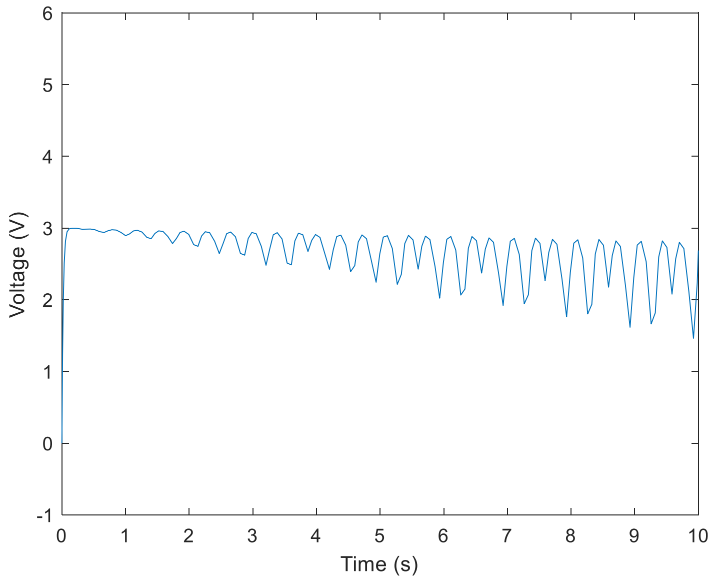


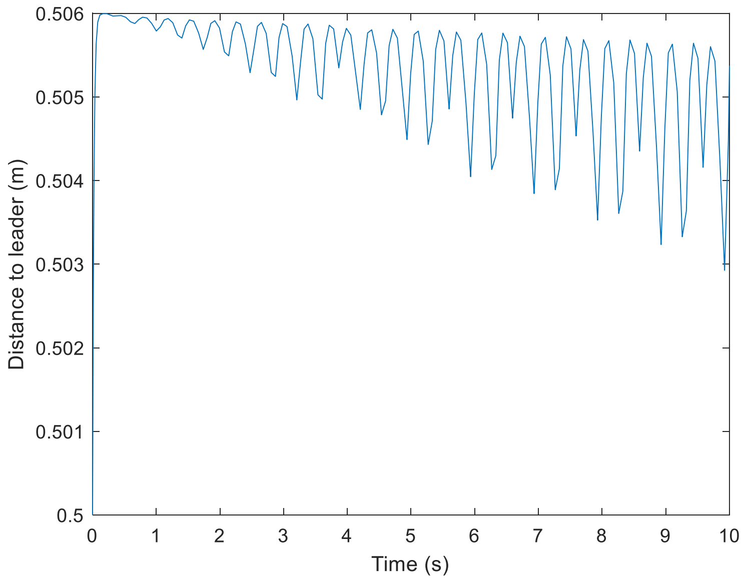
The distance is now further closer to 0.5 !

Durham University

Increase K_p!!!

$$K_p = 100$$

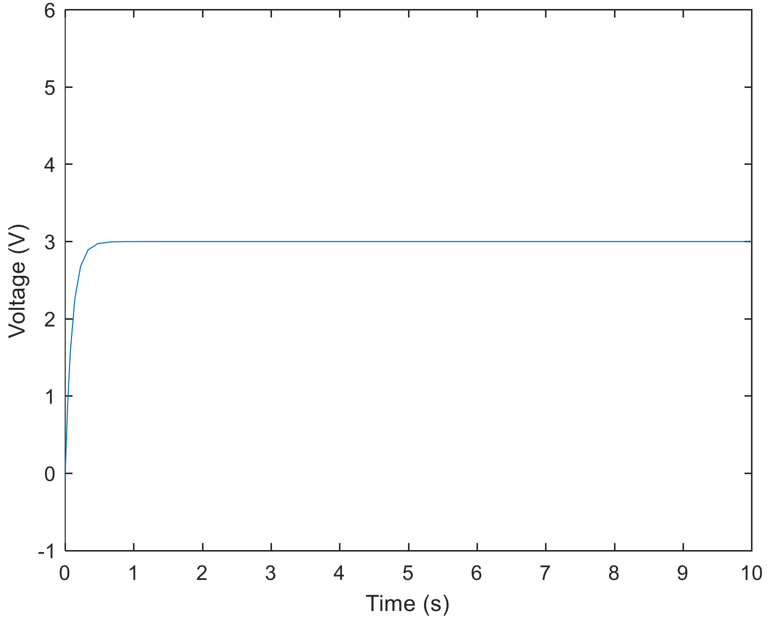


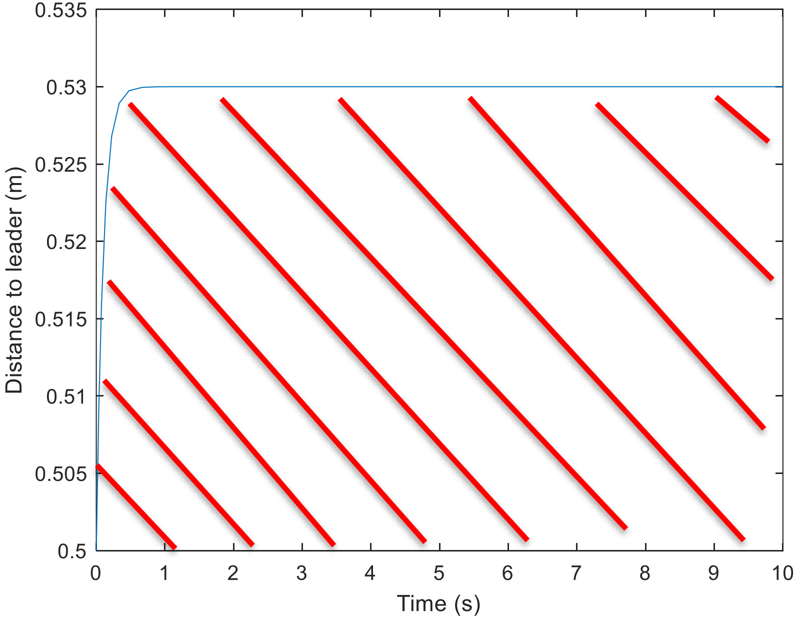


Oh no!

We need to do something different

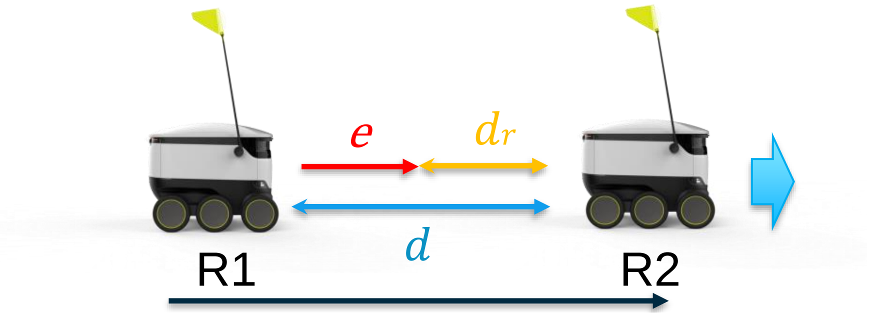
$$K_p = 10$$

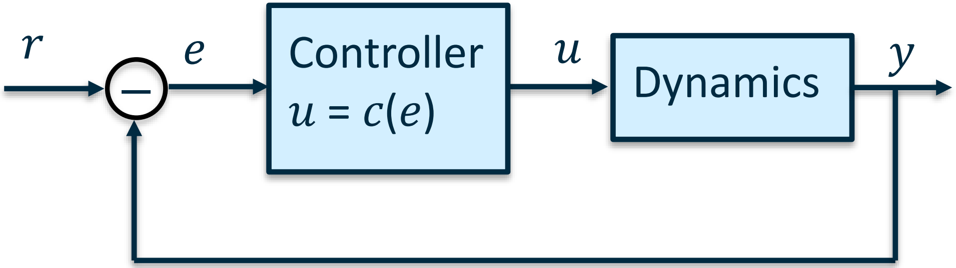




Proportional–Integral (PI) Control

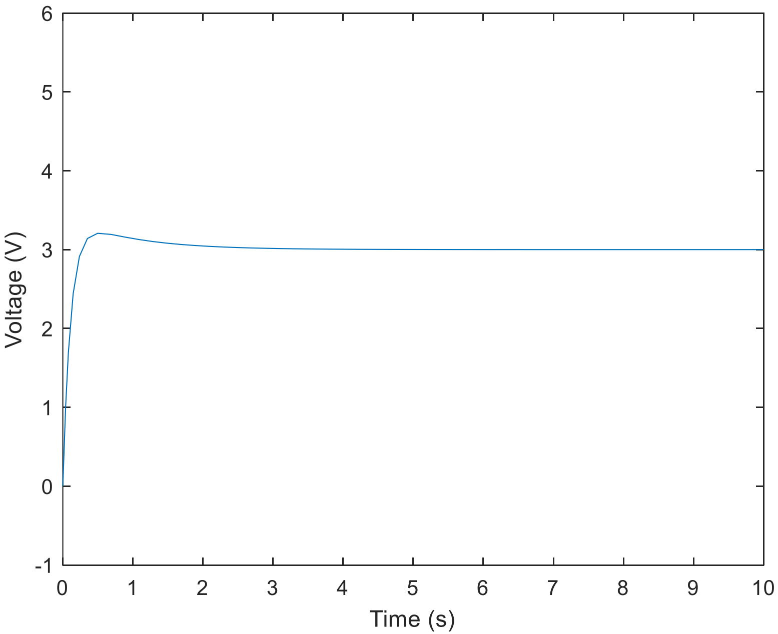
$$c(e) = K_p e(t) + K_i \int_0^t e(t)dt$$

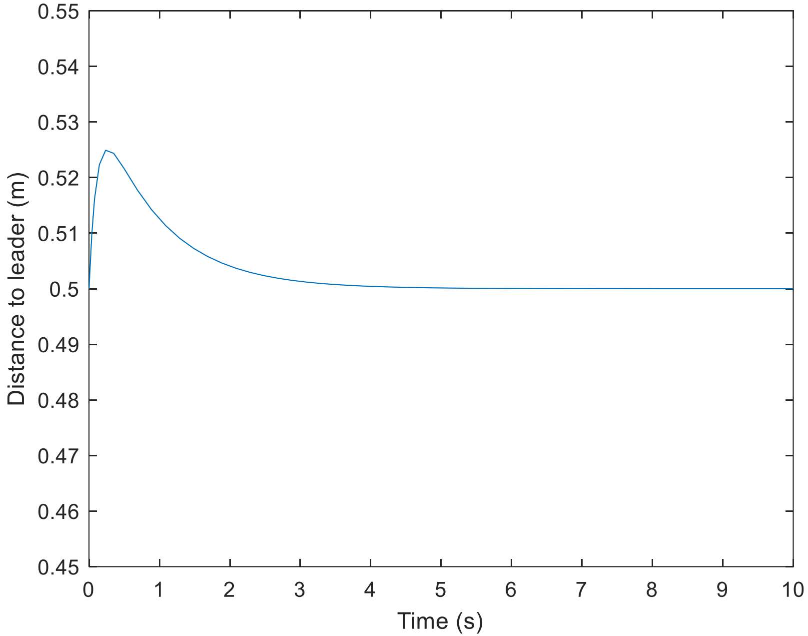




PI Control

$K_p = 10 \qquad K_i = 10$



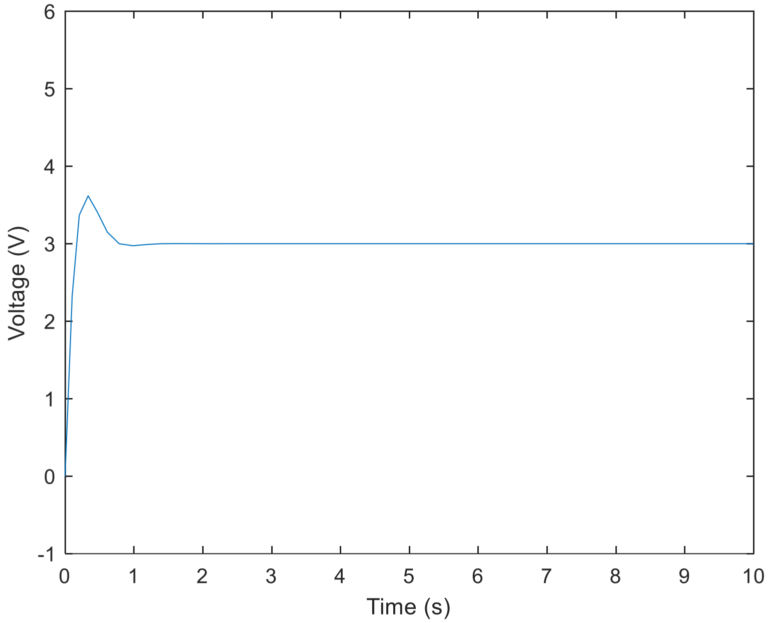


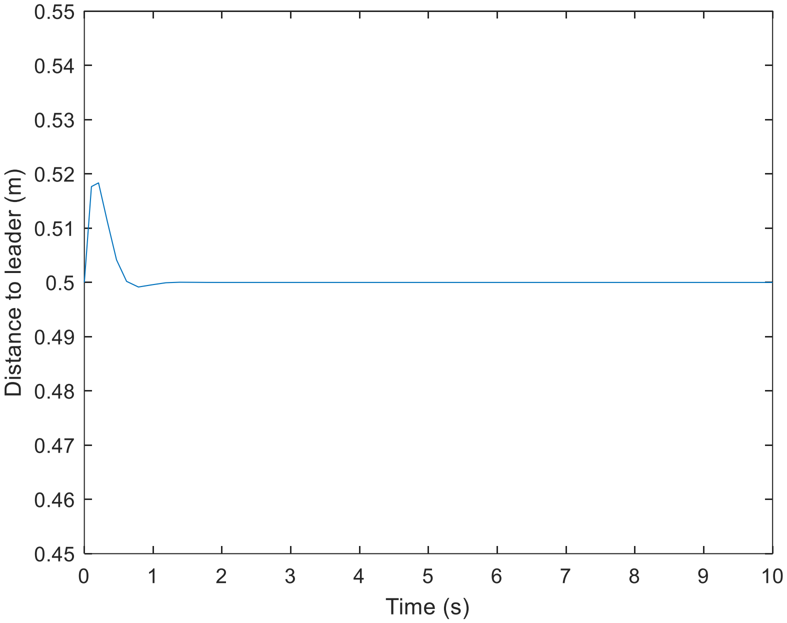
The output converges to 0.5, but it takes longer time

Durham University

PI Control

$K_p = 10 \qquad K_i = 50$



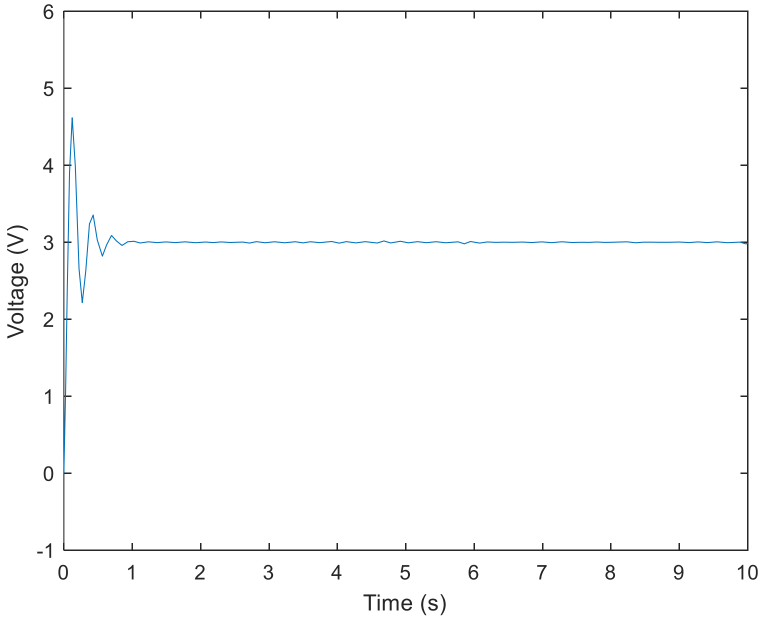


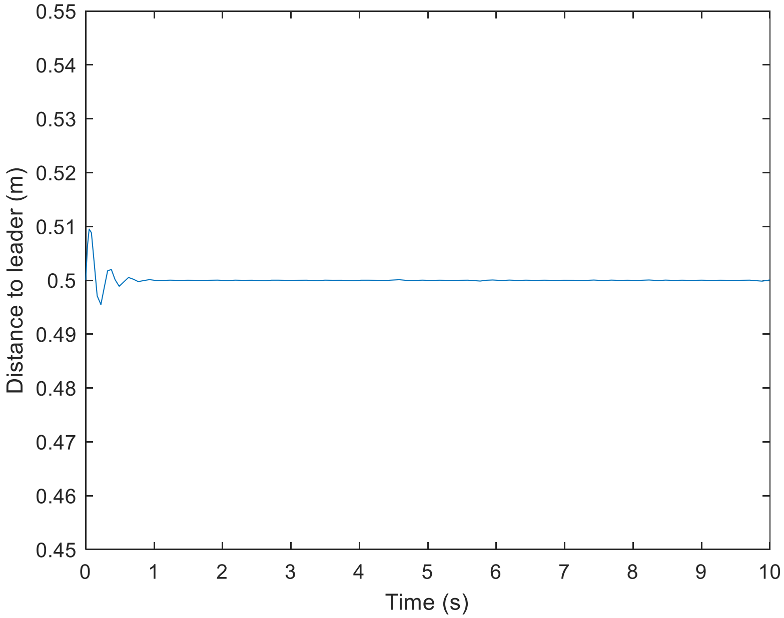
The settling time is shorter

Durham University

PI Control

$$K_p = 10 \qquad K_i = 500$$



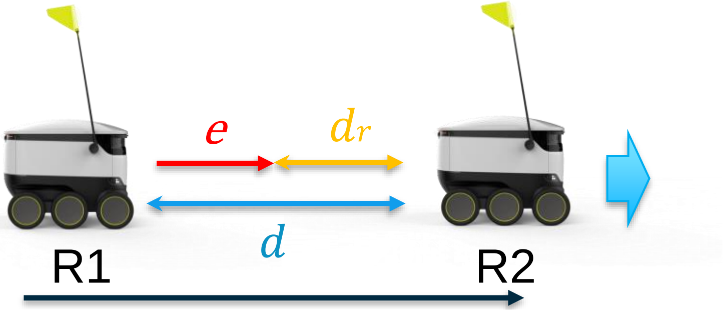


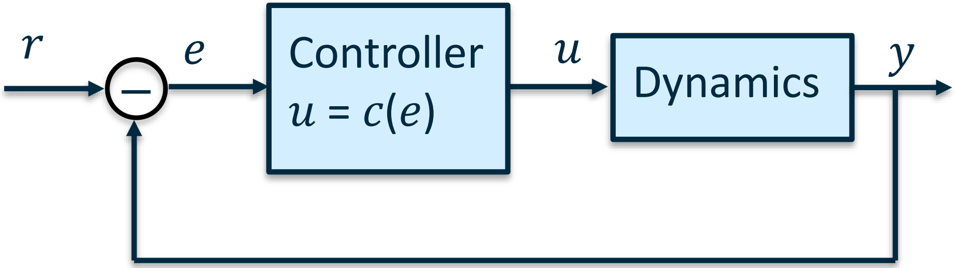
Oh no! It is oscillating again!

Durham University

Proportional-Integral-Derivative (PID) Control

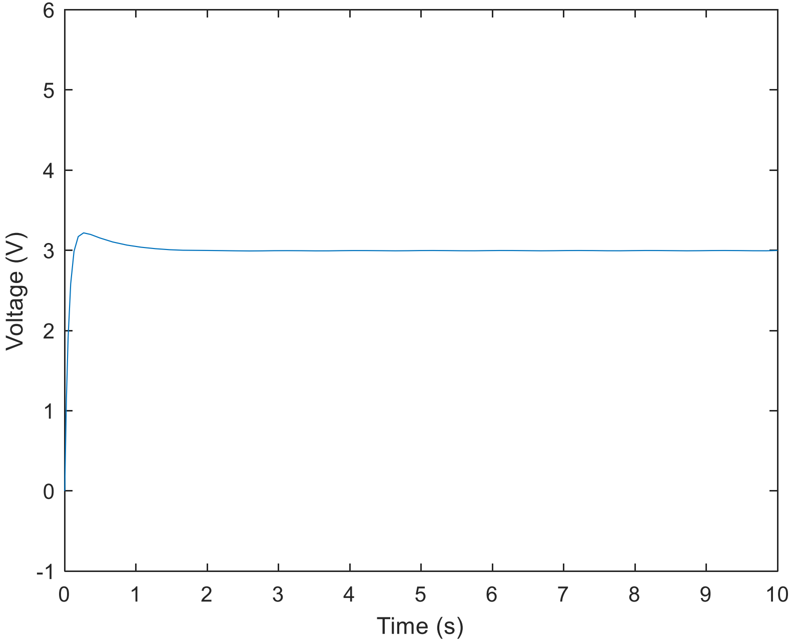
$$c(e) = K_p e(t) + K_i \int_0^t e(t)dt + K_d \frac{d}{dt} e(t)$$

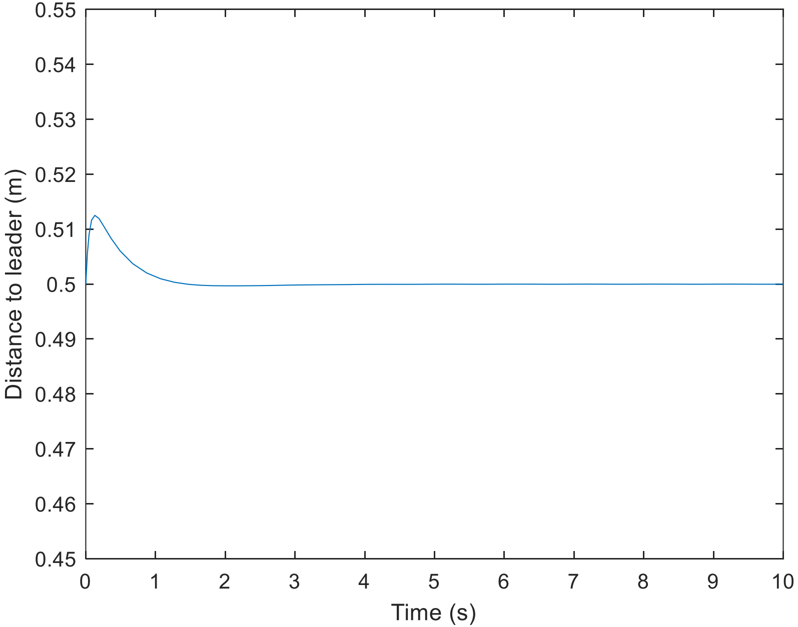




PID Control

$$K_p = 10 \qquad K_i = 50 \qquad K_d = 10$$





The performance looks good in simulation, but would this work well in a real system?

Durham University

Summary of Tuning Tendencies

Response	Rise Time	Overshoot	Settling Time	Steady-State Error
K_p	Decrease	Increase	Small change	Decrease
K_I	Decrease	Increase	Increase	Eliminate
K_D	Small change	Decrease	Decrease	No change

Durham University

Advantages of PID Control

1. Robustness: PID controllers are inherently robust. They can handle various disturbances and changes in the system, such as variations in load, setpoint changes, or changes in system parameters, and still maintain stable control.
2. Stability: Properly tuned PID controllers ensure system stability. They prevent the system from oscillating or becoming uncontrollable, which is crucial in many industrial applications to ensure safety and efficiency.
3. Ease of Implementation: PID controllers are relatively straightforward to implement, both in hardware and software. This simplicity makes them cost-effective and suitable for a wide range of applications.
4. Tuning Flexibility: While PID controllers require tuning to match the specific system, there are well-established methods for tuning PID parameters, such as the Ziegler-Nichols method.
5. Linear and Nonlinear Systems: PID controllers can be applied to linear and nonlinear systems.

Disadvantages of PID Control

1. Tuning Challenges: Tuning PID parameters can be a complex and time-consuming task. Finding the right set of parameters to ensure optimal performance can be challenging.
2. Integral Windup: In cases where the system experiences long periods of sustained error (e.g., saturation or integrator windup), the integral term can accumulate excessively, causing a large overshoot or instability.
3. Not Ideal for Dead Time Dominant Systems: Systems with significant dead time (delay between a control action and its effect on the process) can be challenging for PID control.
4. Limited Performance for Multivariable Systems: PID controllers are typically designed for single-input, single-output (SISO) systems. When dealing with complex, multivariable systems, multiple PID controllers may need to be coordinated.
5. Not Suitable for Some Highly Dynamic Systems: In systems with extremely fast dynamics or systems that require advanced control strategies, such as those in aerospace or high-speed manufacturing, PID control may not be sufficient to achieve the desired performance.

State-Space Representation

- **State** $[\boldsymbol{x}]$**:** A snapshot description of the system.
(e. g. mobile robot location, robotic arm joint configuration)

- **Input** $[\boldsymbol{u}]$**:** What we can do to modify the state.
(e. g. motor rotation)

- **Output** $[\boldsymbol{y}]$: What we can observe from the system.
(e. g. readings from GPS, distance sensors, cameras, etc)

- **Dynamics:** How the state evolves over time (laws of physics)

$$
\begin{bmatrix} u_1(t) \\ u_2(t) \\ \vdots \\ u_M(t) \end{bmatrix}
\longrightarrow
\boxed{
\begin{array}{ll}
\dfrac{d\boldsymbol{x}}{dt} = \dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u}) & \begin{bmatrix} x_1(t) \\ x_1(t) \\ \vdots \\ x_N(t) \end{bmatrix} \\
\boldsymbol{y}(t) = h(\boldsymbol{x}, \boldsymbol{u}) &
\end{array}
}
\longrightarrow
\begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_M(t) \end{bmatrix}
$$

Linear Time Invariant (LTI) systems

- Any system that can be represented in this shape is LTI:

$$\dot{x}(t) = Ax(t) + Bu(t)$$
$$\dot{y}(t) = Cx(t) + Du(t)$$

  Where A, B, C, D are constant matrices/vectors.

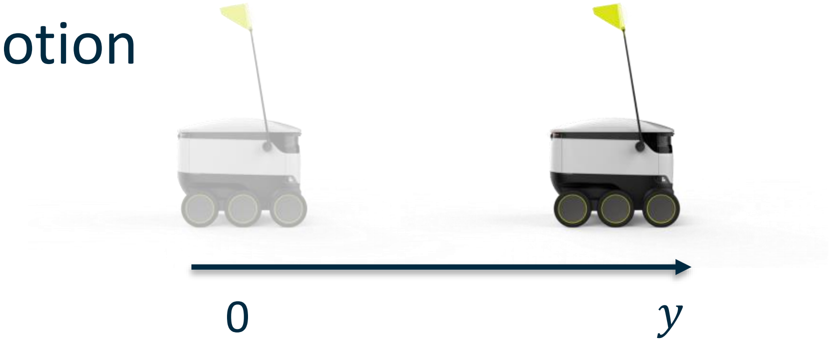
- Linearity:
  - If input $u_1(t)$produces output $y_1(t)$
  - and input $u_2(t)$ produces output $y_2(t)$
  - then input $a_1u_1(t) + a_2u_2(t)$produces output $a_1y_1(t) + a_2y_2(t)$
- Time invariance
  - If input $u(t)$ produces output $y(t)$
  - then input $u(t-T)$ produces output $y(t-T)$

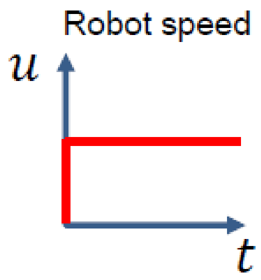
Durham University

Single-Integrator System

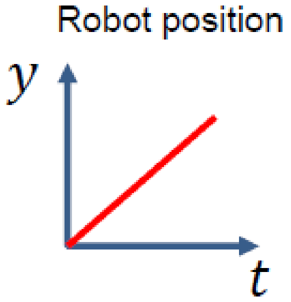
Mobile robot with 1-dimensional motion

- **State** [x]: robot position
- **Input** [u]: robot speed
- **Output** [y]: robot position





$$\dot{x} = u(t)$$
$$y(t) = x(t)$$



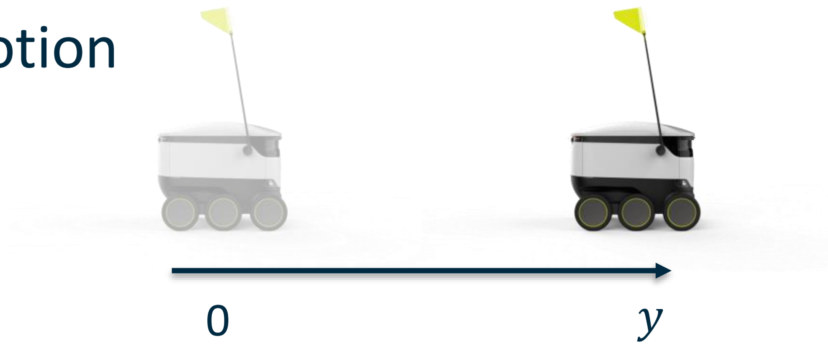
$$A = 0, B = 1, C = 1, D = 0$$

Durham University

Double-Integrator System

Mobile robot with 1-dimensional motion

- **State 1** $[x]$: robot position
- **State 2** $[v]$: robot velocity
- **Input** $[u]$: robot acceleration
- **Output** $[y]$: robot position



$$\dot{x} = v(t)$$
$$\dot{v} = u(t)$$
$$y(t) = x(t)$$

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v(t) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ u(t) \end{bmatrix}$$
$$y(t) = x(t)$$

$$\begin{bmatrix} \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} + 0 * u(t)$$

Output of the LTI System

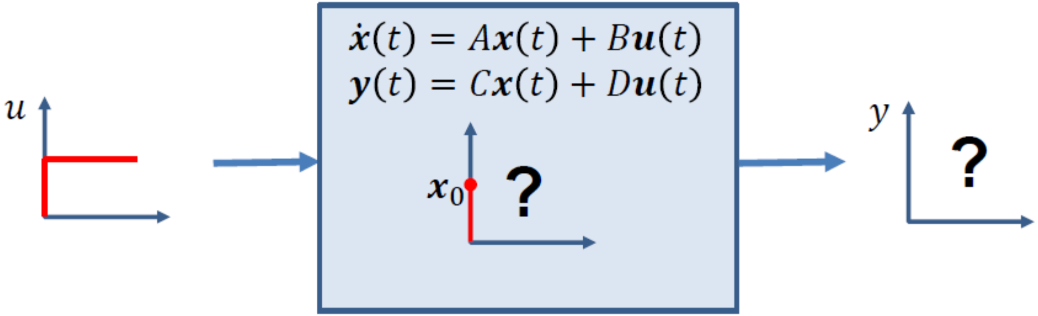
Predict (or simulate) the dynamics of an LTI system

Given

- A LTI system with known A, B, C, D
- An initial state with $x_0 = x(0)$
- A known input signal $u(t)$

Find

- How state $x(t)$ and output $y(t)$evolve over time



Durham University

Initial Condition Response

- Consider no control input

$$\dot{\boldsymbol{x}} = A\boldsymbol{x}$$

- Now, if $A = a$ is a scalar:

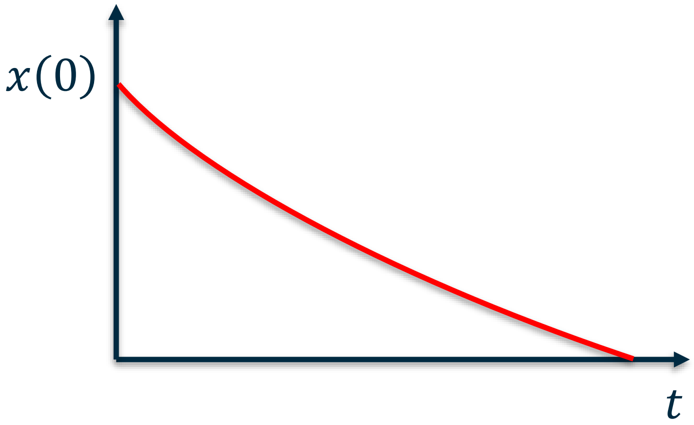
$$\dot{x} = ax$$

- The time response is given by

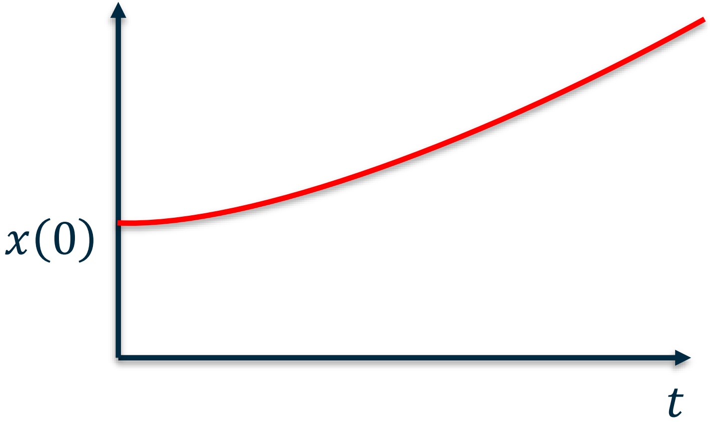
$$x(t) = e^{at}x(0)$$

Exponential

- If $a < 0$



- If $a > 0$



Matrix Exponential

- Similarly, if A is a matrix, the Taylor expansion of e^A is

$$e^A = I + A + \frac{A^2}{2!} + \frac{A^3}{3!} + \cdots = \sum_{k=0}^{\infty} \frac{A^k}{k!}$$

- Then we have

$$e^{At} = I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \cdots \quad = \quad \sum_{k=0}^{\infty} \frac{A^k t^k}{k!}$$

- Differentiating

$$\frac{d}{dt} e^{At} = A + A^2 t + \frac{A^3 t^2}{2!} + \cdots \quad = \quad A\left(I + At + \frac{A^2t^2}{2!} + \frac{A^3t^3}{3!} + \cdots\right)$$

$$= Ae^{At}$$

Durham University

Matrix Exponential

- Hence, we have

$$\dot{x}(t) = \frac{d(e^{At})}{dt} x_0 = A e^{At} x_0 = A x(t)$$

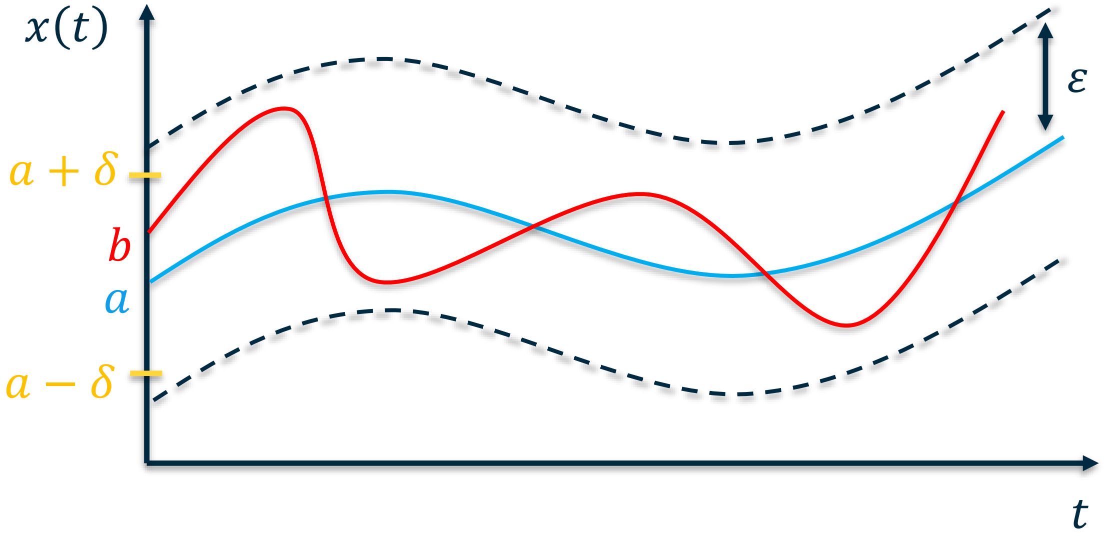
- The time response is given by

$$x(t) = e^{At} x(0)$$

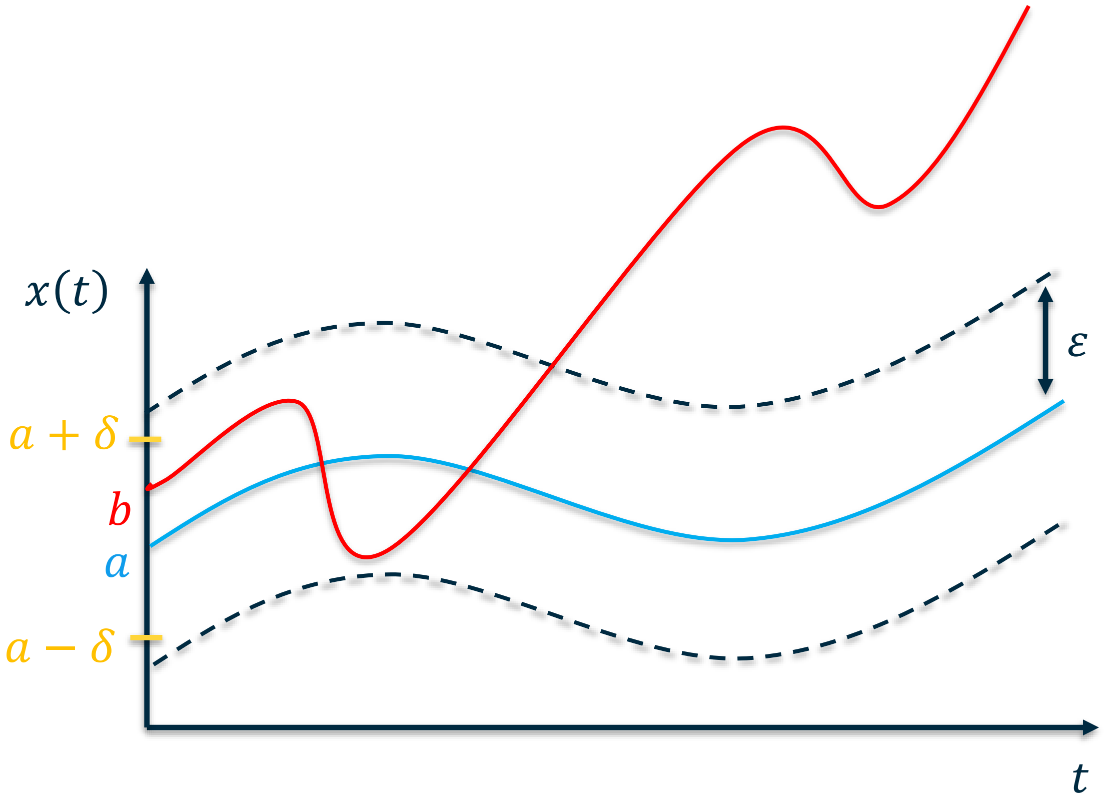
Lyapunov Stability

- Let $x(t;a)$ be a solution to $\dot{x} = f(x)$ with initial condition a
- A solution is stable in the sense of Lyapunov if other solutions that start near a stay close to $x(t;a)$
- For all $\varepsilon > 0$ is there exists $\delta > 0$ such that

$$|b-a| < \delta \Rightarrow |x(t;b) - x(t;a)| < \varepsilon \quad \forall t \geq 0$$



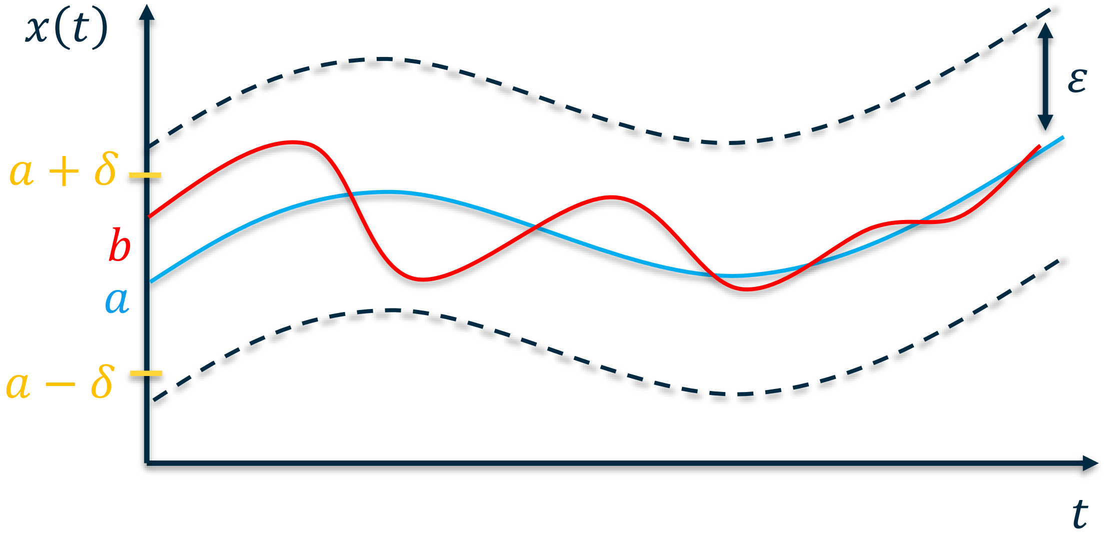
Unstable System



Asymptotic Stability

- When a system verifies the following:
    - It is Lyapunov stable
    - Additionally:

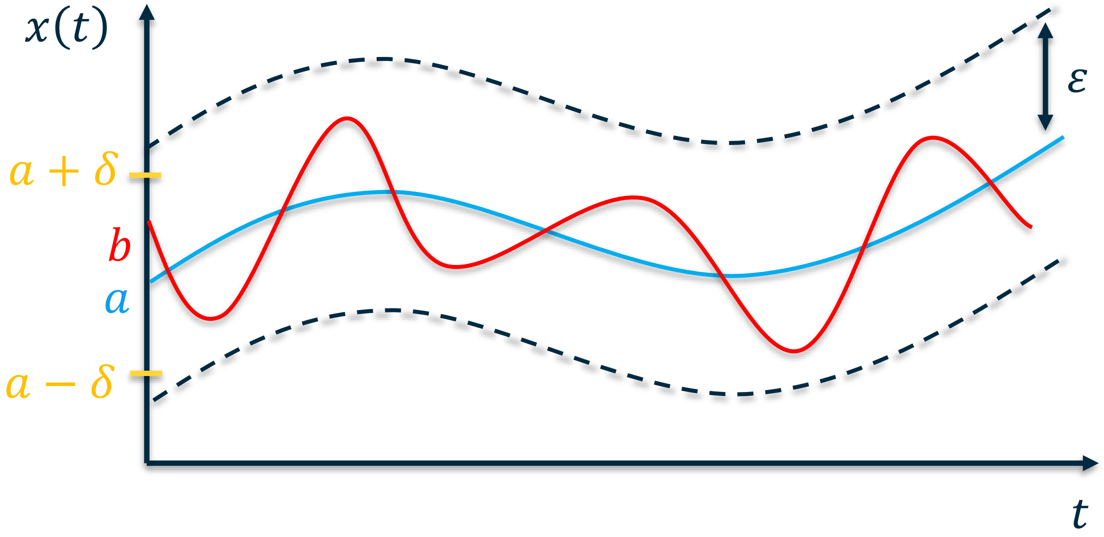
$$|b-a|<\delta \Rightarrow \lim_{t\to\infty}|x(t;b)-x(t;a)|=0$$



Durham University

Neutral Stability

- When a system verifies the following:
  - It is Lyapunov stable
  - It is not asymptotically stable



Durham University

Stability of the LTI System

$$\dot{x}(t) = Ax(t) + Bu(t)$$

- Let's say $u(t)$ is either known or depends on $x(t)$

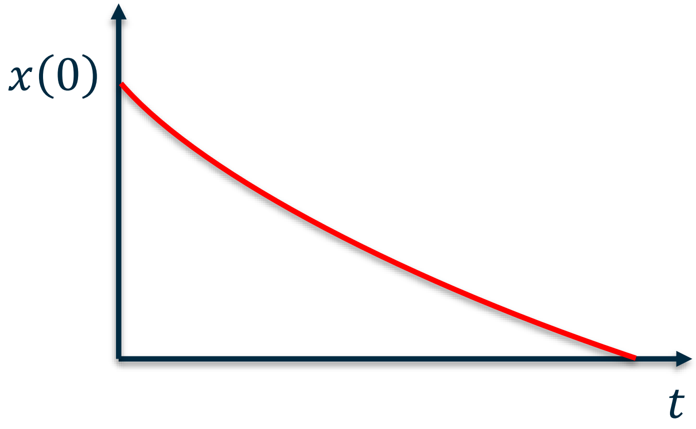
- Can we determine the stability of the system from A, B ?

Scalar Exponential Response

- Assuming no input, and A is a scalar, we have

$$\dot{x} = ax \qquad\qquad x(t) = e^{at}x(0)$$

- If $a < 0$, the system is asymptotically stable

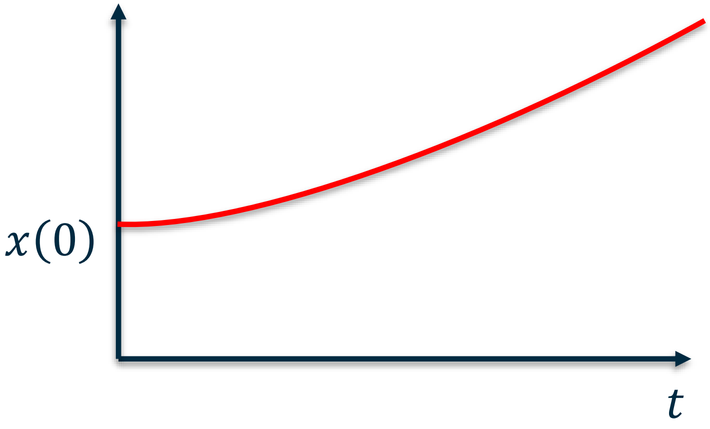


Scalar Exponential Response

- Assuming no input, and A is a scalar, we have

$$\dot{x} = ax \qquad\qquad x(t) = e^{at}x(0)$$

- If $a > 0$, the system is not stable

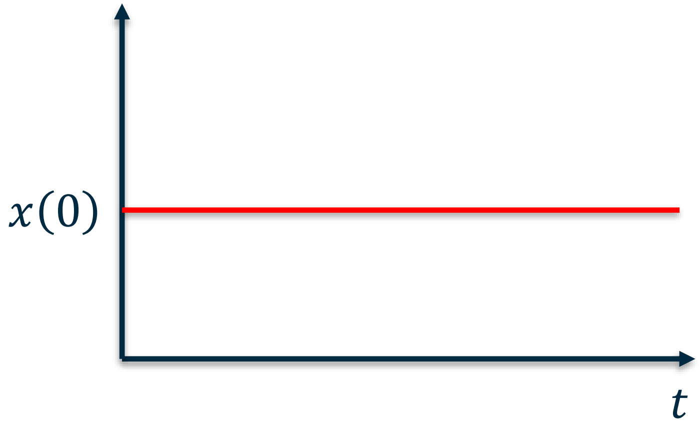


Scalar Exponential Response

- Assuming no input, and A is a scalar, we have

$$\dot{x} = ax \qquad\qquad x(t) = e^{at}x(0)$$

- If $a = 0$, the system is neutrally stable



Matrix Exponential Response

- If A is a matrix, a matrix A is diagonalisable if there is an invertible matrix T and a diagonal matrix Λ such that:

$$\Lambda = T^{-1}AT = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix}$$

- Choose a set of coordinates z for our state such that

$$Tz = x$$

- Then

$$T\dot{z} = \dot{x} = Ax \qquad \dot{z} = T^{-1}ATz = \Lambda z$$

- $\dot{z} = \Lambda z$ has the same stability properties as $\dot{x} = Ax$

Durham University

Matrix Exponential Response

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & \cdots & 0 \\ 0 & \lambda_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \lambda_n \end{bmatrix} \qquad e^{\Lambda t} = \begin{bmatrix} e^{\lambda_1 t} & 0 & \cdots & 0 \\ 0 & e^{\lambda_2 t} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{\lambda_n t} \end{bmatrix}$$

- The system is asymptotically stable if

$$\lambda_i < 0 \qquad \forall i \in \{1,2,\dots,n\}$$

- The system is not stable if

$$\exists \lambda_i > 0 \qquad i \in \{1,2,\dots,n\}$$

- The system is neutrally stable if

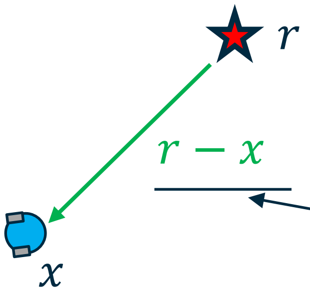
$$\exists \lambda_i = 0 \qquad i \in \{1,2,\dots,n\}$$

$$\lambda_i \leq 0 \qquad \forall i \in \{1,2,\dots,n\}$$

Durham University

Example: Target Tracking

- **The setup:**
  - Given a mobile robots who can only measure the relative displacement of its target (no global coordinates)



This is what robot i can measure

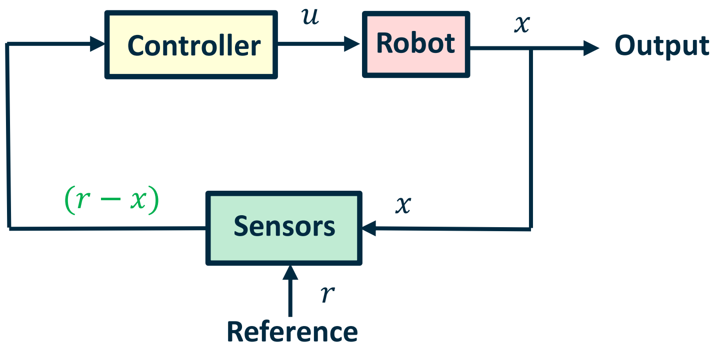
  - Problem: Have the robot meet at the desired position

Example: Target Tracking

- **Robot dynamics:**
  - $\dot{x} = u$

- **Controller design:**
  - $u = K(r - x)$

A positive constant



- **Condition:**
  - The target can always be detected by the robot using onboard/external sensors (e.g., a camera)

Example: Target Tracking

Theoretical guarantee

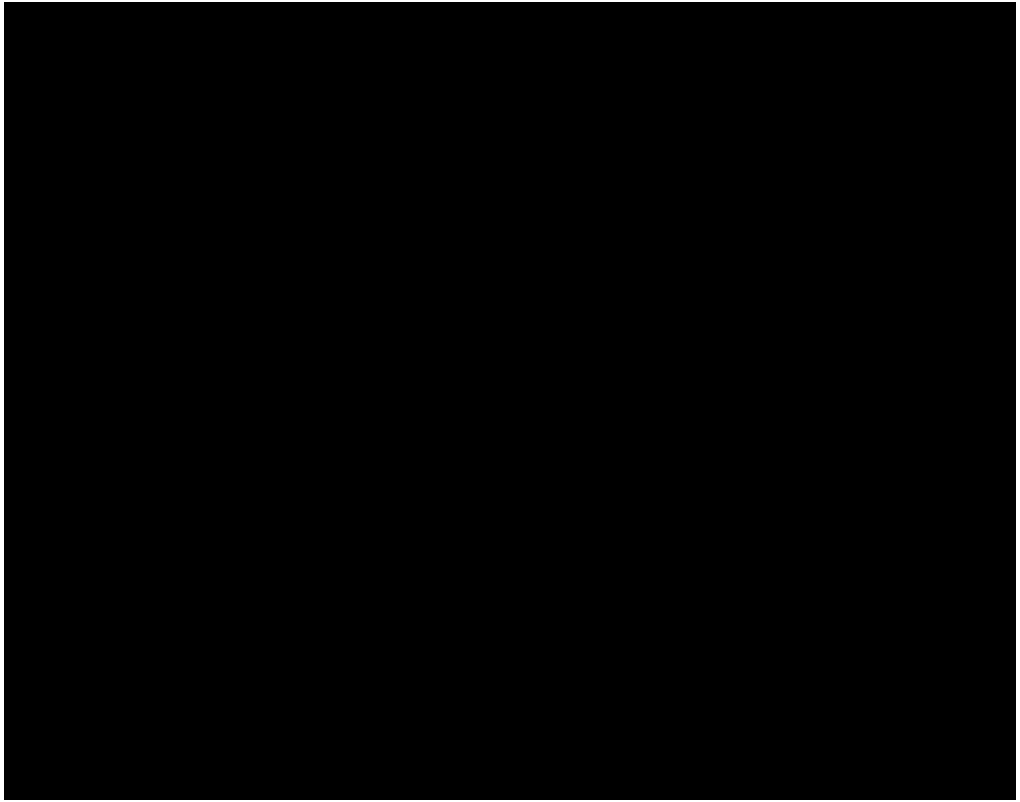
- Define tracking error signal $e = r - x$, we have

$$\dot{e} = -\dot{x} = -u = -Ke$$

- Hence, the error system is asymptotically stable

Example: Target Tracking

Command



Durham University

Stability of Nonlinear Systems

- We consider nonlinear time-invariant system $\dot{x} = f(x)$
- A point x_e is an equilibrium point of the system if $f(x_e) = 0$

The system is globally asymptotically stable if for every trajectory $x(t)$, we have $x(t) \to x_e$ as $t \to \infty$

Durham University

Positive Definite Functions

A function V is *positive definite* if

- $V(x) \geq 0$ for all x
- $V(x) = 0$ if and only if $x = 0$
- $V(x) \to \infty$ as $x \to \infty$

Example: $V(x) = x^T P x$, with $P = P^T$, is *positive definite* if and only if $P > 0$.

Durham University

Lyapunov Theory

Lyapunov theory is used to make conclusions about trajectories of a system $\dot{x} = f(x)$ without finding the trajectories (i.e., solving the differential equation)

a typical Lyapunov theorem has the form:

- **If** there exists a function $V(x)$ that satisfies some conditions on V and $\dot{V}$.

- **Then** trajectories of system satisfy some property

If such a function V exists we call it a Lyapunov function (that proves the property holds for the trajectories)

Durham University

Lyapunov Stability Theorem

Suppose there is a function V such that

- $V\ (x)$ is *positive definite*
- $\dot{V}(x) < 0$ for all $x \neq 0$, $\dot{V}(0) = 0$

Then, every trajectory of $\dot{x} = f(x)$ converges to zero as $t \rightarrow \infty$
(i.e., the system is globally asymptotically stable)

Lecture Summary

- Feedback Systems
- Bang-Bang Control
- PID Control
- State-Space Representation
- Stability of the System