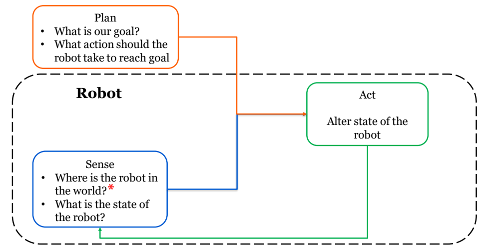
Software and Robotic Integration

Feedback Control

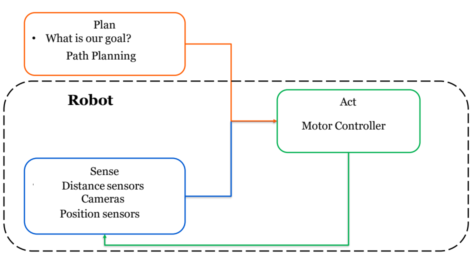
Rachel Sparks, Ph.D.
Rachel.sparks@kcl.ac.uk
Lecturer in Surgical & Interventional Engineering
School of Biomedical Engineering & Imaging Sciences

Robot Control Diagram

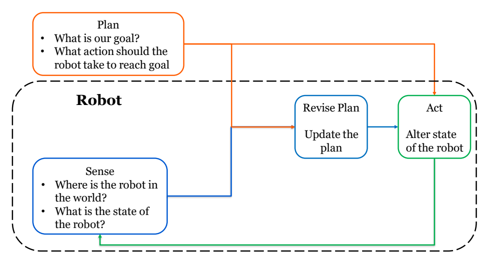


*Can be achieved through external sensors

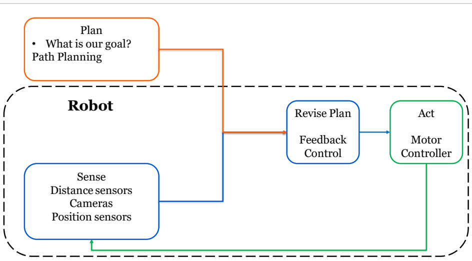
Robot Control Diagram



What is a Robot?



What is a Robot?



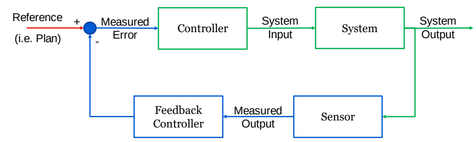
Feedback Control

The aim of feedback control is to ensure the goal state is reached with
- Minimal delay
- Minimum steady state error
- As quickly as possible
- Stability – i.e. will converge to a final solution

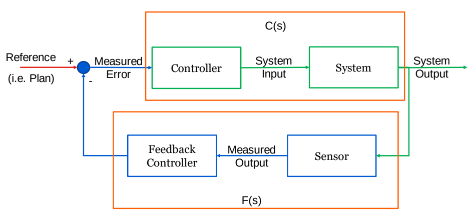
Feedback Control

Unlike in ROS where we can set a joint to a given state:
- Motors apply force to move a joint – friction, damping, or external forces can all cause a joint motion to deviate from the desired state
- Over time small errors tend to be multiplicative and cause drift that may need to be corrected
- The state of the world can be dynamic – i.e. the world may change relative position or another state in response to the movement in the robot that may require updating the plan.

Feedback Block Diagram



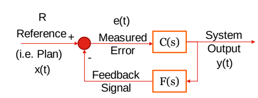
Feedback Block Diagram



Feedback Design

- The goal is to pick an appropriate feedback function F(s) to appropriately alter the forward function C(s) and converge to a reference state R (i.e. minimize measurement error)
- The entire system has a closed loop transfer function (Laplacian Space)

$$x(s) = \frac{C(s)}{1 + C(s)F(s)}$$



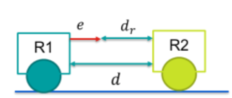
Feedback Algorithm Design

$$H(s) = \frac{C(s)}{1 + C(s)F(s)}$$

Care must be taken to ensure H(s) is stable
- Bounded input-bounded output (BIBO): for any input x(t) < B the output will also be y(t) < B
- The poles of H(s) must have real negative values (Laplacian Transform/Bode Plot)

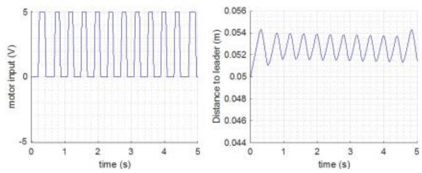
Follower Robot

The goal is to keep R1 at a fixed distance d_r from R2.
- R2 moves at a constant speed s
- Input[x(t) = u]: On-Off DC motor
- Output [$y(t)$ = d]:distance between R1 and R2



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

Follower Robot – Feed forward Only



Proportional Control

- Linear feedback control proportional to the measured error e(t)

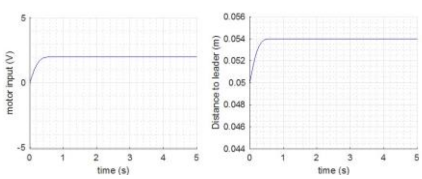
$$C_{out} = K_p e(t) + C_0$$

where K_p is the gain, C_{out} is the control output and c_0 is the control output when the reference R have been achieved

- Advantage: easy to control and tune – small K_p will ensure error remains bounded
- Disadvantages:
  - Slow to converge – as you near the goal the change in state gets smaller and smaller
  - Must carefully select K_p to reduce overshooting and ensure convergence

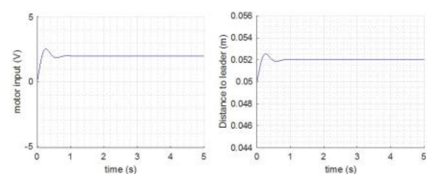
Follower Robot – Proportional Control

$K_p = 500$



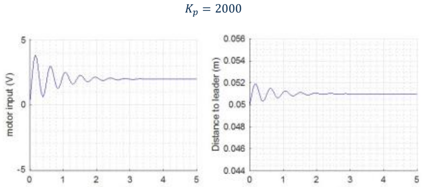
Follower Robot – Proportional Control

$K_p = 1000$



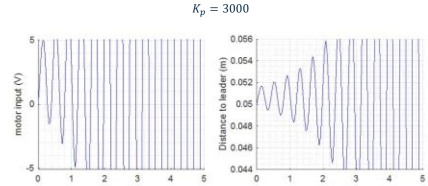
Follower Robot – Proportional Control

$K_p = 2000$



Follower Robot – Proportional Control

$K_p = 3000$

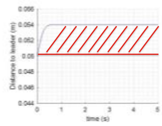


Integral Control

- Feedback control proportional to the integral of the measured error.

$$C_{out} = K_i \int_0^t e(\tau)\, d\tau + C_0$$

where K_i is the gain.



- Over time the feedback will grow
- Advantage: Large steps after sufficient time has passed,
- Disadvantage: Unstable over long time scales. $c_{out} \rightarrow \infty$ as $t \rightarrow \infty$

Proportional Integral Control

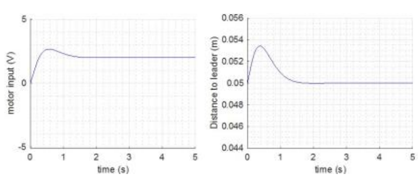
- Initially proportional control dominates, overtime integral control will dominate

$$C_{out} = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + C_0$$

- Advantages: converges to the desired solution
- Disadvantages: slow to converge, still prone to overshooting

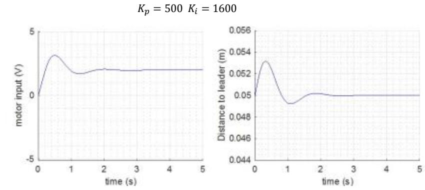
Follower Robot – PI Control

$K_p = 500 \;\; K_i = 800$



Follower Robot – PI Control

$K_p = 500 \;\; K_i = 1600$



Derivative Control

Responds to the change in error over time

$$C_{out} = K_d \frac{d(e(t))}{dt} + c_0$$

- Advantage: have a rapid response to changes in the system
- Disadvantages: sensitive to noise in sensors, unstable

PID Controller

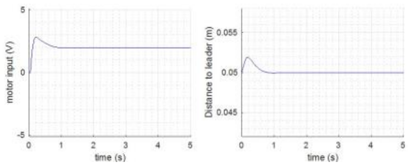
Combines proportional, integral, and derivative control and has the best system response

$$C_{out} = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{d(e(t))}{dt} + C_0$$

- Advantages: rapid convergence, can be precisely tuned to precisely fit to a given problem
- Disadvantages: tuning the parameters between the three components can be difficult

Follower Robot – PID Control

$K_p = 1000\ K_i = 3000\ K_d = 50$



Transfer Functions

Feedback System	Transfer Function
P	K_p
PI	$K_p + \frac{K_i}{\alpha D}$
PD	$K_p + K_d \alpha D$
PID	$K_p + \frac{K_i}{\alpha D} + K_d \alpha D$

Feedback Control Summary

Feedback control is needed in real world systems to

- Remedy systemic issues — friction, manufacturer variability in parts,
- Account for sensor error – noise in measurements, drift in localization
- Correct for variability – applied effort for motors, external forces, change in object positions

Feedback requires considering

- What information (P,I,D) to consider for error measurements
- What is the desired time response (fast or slow?, robust to noise?)
- Stability of the Transfer function – really require full course on Control Theory.