

CISC 1003 - EXPLORING ROBOTICS



ROBOT CONTROL TYPES

CISC1003

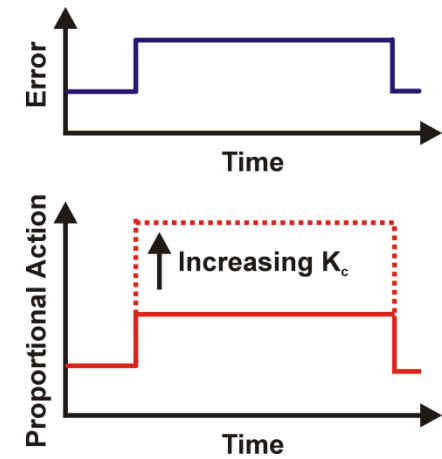
Feedback Control Types

The three most used types feedback control are:

- Proportional control (P)
- Proportional Derivative control (PD)
- Proportional Integral Derivative control (PID)

Proportional Control

- System responds in proportion to the error
 - using both the direction and the magnitude of the error.
- For wall-following robot:
 - Use distance-to-wall to determine the angle and distance and/or speed with which the robot would turn.



Proportional Control

- *Gains*: The parameters that determine the magnitude of the system's response
 - In control theory
- *Damping*:
 - The process of systematically decreasing oscillations.
 - A properly *damped system* => does not oscillate out of control.

Derivative Control

- The controller corrects for the momentum of the system as it approaches the desired state.
 - Relative to the rate the error changes
- When the system is close to the desired state, it needs to be controlled differently than when it is far from it
 - Otherwise, the controller's correction will carry the system beyond the desired state
 - cause oscillations

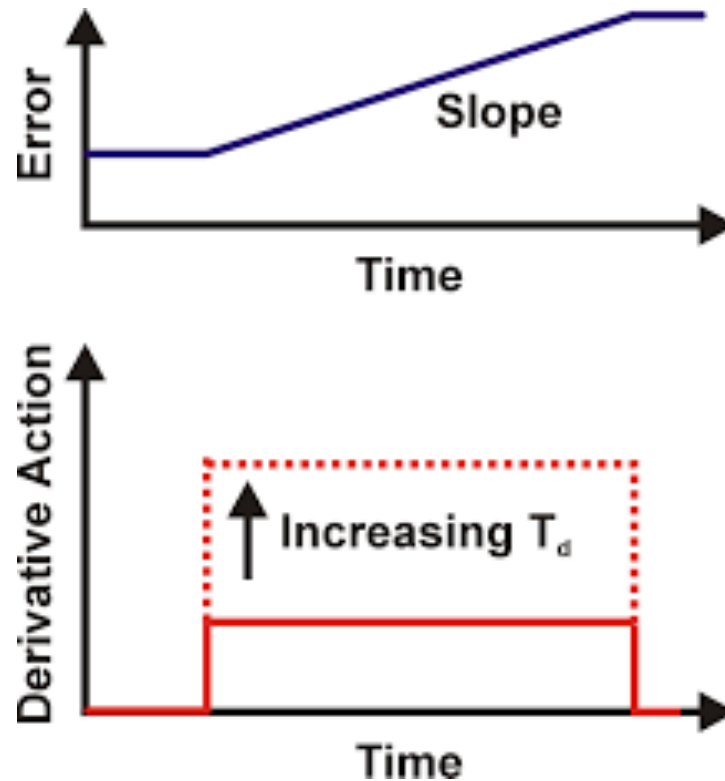
Derivative Control

- Best estimate the future trend of the error, based on the current rate of change

Control - Example

- Robot moving towards a target
- Proportional control:
 - Speed changes based on distance from target
- Derivative control:
 - Speed changes based on current speed of robot
 - The faster it is, the more we will slow it down
 - So it does not hit a wall

Derivative Control



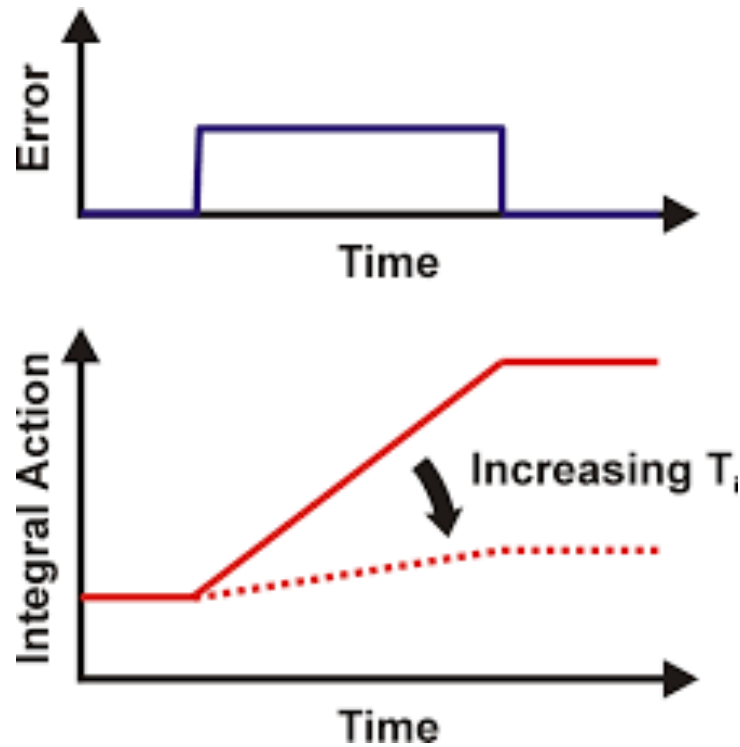
Integral Control

- The system integrates (sums up) these incremental errors over time
 - When reaching a predetermined threshold the system does something to compensate/correct.
 - Once the cumulative error gets large enough
- Example: A robot lawn mower with an error when turning to cut sections of grass.
 - With a system to detect the error, it can compensate for it over time.

Integral Control

- Account for past behavior of system
- Integrates the error over time

Integral Control



Control - Example

- Robot moving towards a target
- Proportional control:
 - Speed changes based on distance from target
- Derivative control:
 - Speed changes based on current speed of robot
 - The faster it is, the more we will slow it down

Control - Example

- Integral control:
 - Speed change based on overall distance from wall
 - If distance was small for a long time \Rightarrow small change required
 - May be good for a wall-following program
 - If distance changes drastically, then overall integral will minimize changes
 - The sum of changes changes less in time
 - Reduce the need to make constant changes to the robot speed
 - Helps overcome inaccurate sampling

Derivative Control Systems

- Proportional Derivative control (PD)
 - Applied in most industrial plants
 - Extremely useful for process control.
- Proportional Integral Derivative control (PID)
 - A combination of proportional P , integral I , and derivative D control terms.

Control Architectures

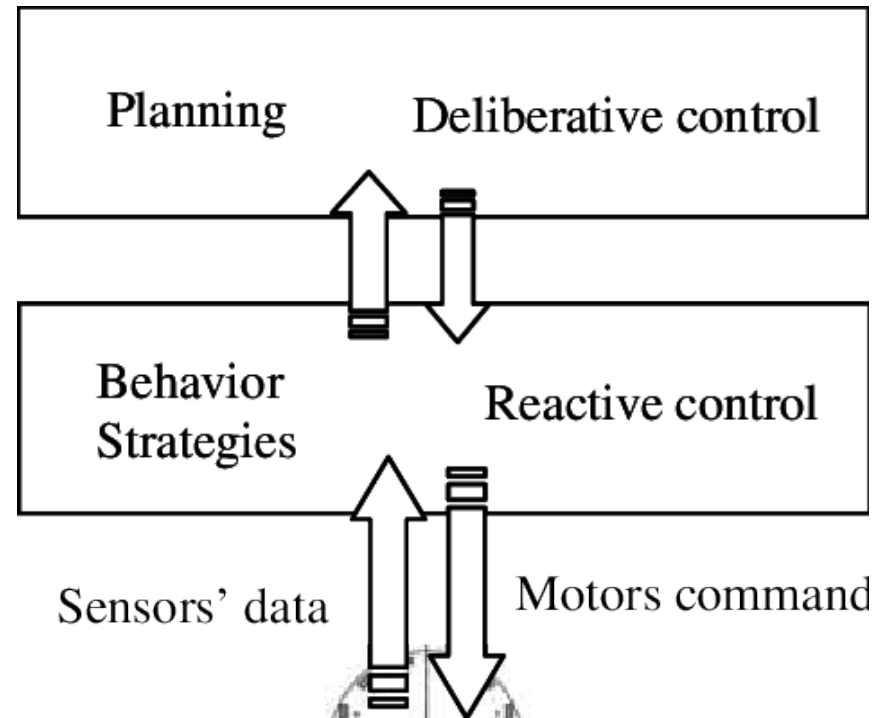
- The controller functions as the robot's brains
 - So robot can be autonomous, achieve its goals.
- The robot can sense multiple things at once.
 - The controller decides what the robot needs to observe.

Control Architectures

- Not all control architectures are the same
- Different capabilities allow for different robot functionality
 - Regardless of which language is used to program a robot, architecture will determine functionality
 - Programming language is just a tool to program robot

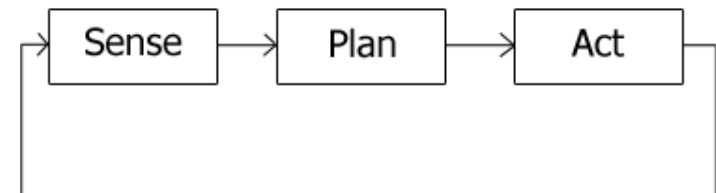
Control Architectures

- May include two components:
 - Deliberative Control
 - Reactive Control



Deliberative Control

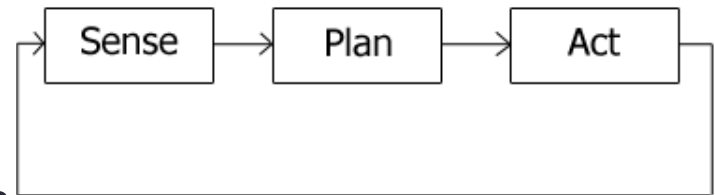
- *Planning:*
 - The process of looking ahead at the outcomes of the possible actions
 - searching for the sequence of actions that will reach the desired goal.
- *Search:*
 - An inherent part of planning.
 - involves looking through the available representation “in search of” the goal state.



Deliberative Control (cont.)

- Deliberative, planner-based architectures involve three steps that need to be performed in sequence:

1. Sensing (S)
2. Planning (P)
3. Acting (A), executing the plan.



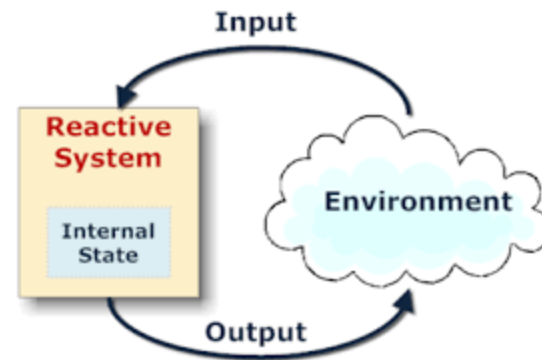
- These steps need to be performed in sequence

Reactive Control

- Reactive rules are similar to *reflexes*
 - innate responses that do not involve any thinking.
- How to design a good reactive system?
 - Keep it simple and straightforward
 - Define robot states:
 - Situations that can be detected by the robot's sensors
 - Each unique state should trigger only one unique robot action

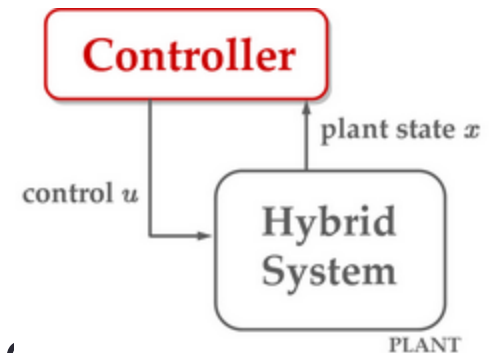
Reactive Control

- Reactive rules must support parallelism
 - To handle checking multiple sensors.

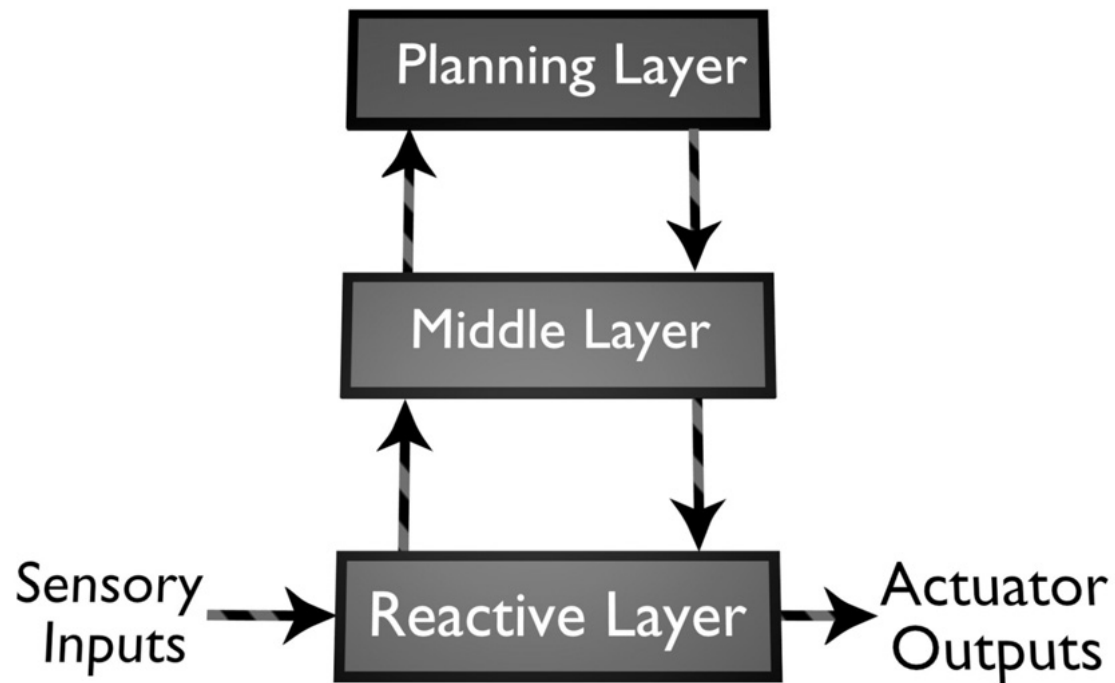


Hybrid Control

- Involves the combination of reactive and deliberative control
 - within a single robot control system
- Components:
 - Planner
 - Middle layer that links the layers together.
 - (by issuing commands).
 - Reactive layer



Hybrid Control



Behavior-Based Control

- *Behavior-based control* (BBC) involves the use of "behaviors" as modules for control.
- Behaviors achieve and/or maintain complex goals.
 - A *homing* behavior:
 - Achieves the goal of getting the robot to the home location.
 - A *wall-following* behavior:
 - maintains the goal of following a wall.

Behavior-Based Control

- Behaviors take time to execute and are not instantaneous.
- Requires constantly monitoring the sensors and other behavior status variables.

Behavior-Based Control

