Agent Architectures and Hierarchical Control

Overview:

- Agents and Robots
- Agent systems and architectures
- Agent controllers
- Hierarchical controllers

Agents and Robots

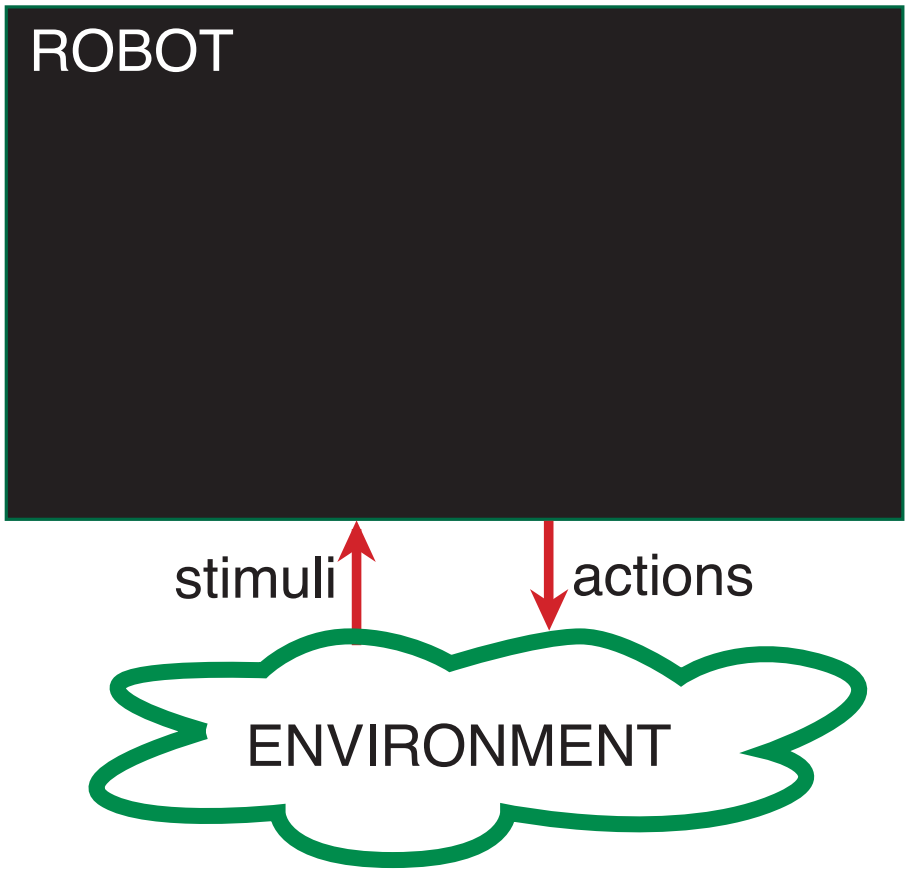
A situated agent perceives, reasons, and acts in time in an environment.

- An **agent** is something that acts in the world.
- A **purposive agent** prefers some states of the world to other states, and acts to try to achieve worlds they prefer.
- Agents interact with the environment with a **body**.
- An **embodied** agent has a physical body.
- A **robot** is an artificial purposive embodied agent.

What makes an agent?

- Agents can have sensors and effectors to interact with the environment.
- Agents have (limited) memory and (limited) computational capabilities.
- Agents reason and act in time.

Agent Systems

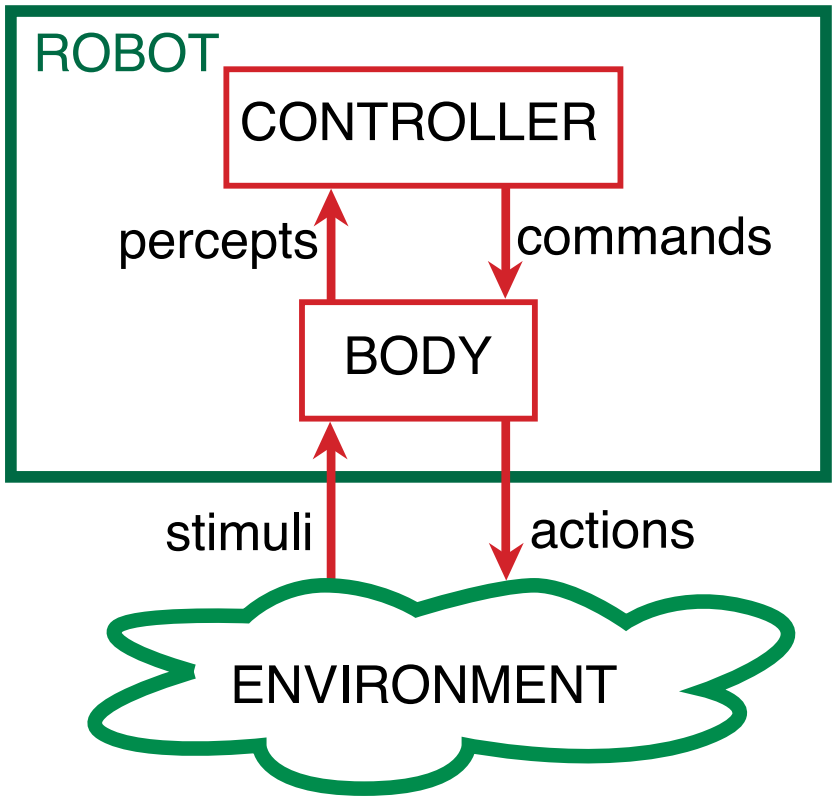


A agent system is made up of a agent and an environment.

- An agent receives stimuli from the environment
- An agent carries out actions in the environment.

Agent System Architecture

An agent is made up of a body and a controller.



- An agent interacts with the environment through its body.
- The body is made up of:
  - sensors that interpret stimuli
  - actuators that carry out actions
- The controller receives percepts from the body.
- The controller sends commands to the body.
- The body can also have reactions that are not controlled.

Implementing a controller

- A controller is the brains of the agent.
- Agents are situated in time, they receive sensory data in time, and do actions in time.
- The controller specifies the command at every time.
- The command at any time can depend on the current and previous percepts.

The Agent Functions

- Let T be the set of time points.
- A percept trace is a function from T into P, where P is the set of all possible percepts.
- A command trace is a function from T into C, where C is the set of all commands.
- A transduction is a function from percept traces into command traces.
- A transduction is causal if the command trace up to time t depends only on percepts up to t.
- A controller is an implementation of a causal transduction.

Belief States

- A causal transduction specifies a function from an agent's history at time t into its action at time t.
- An agent doesn't have access to its entire history. It only has access to what it has remembered.
- The internal state or belief state of an agent at time t encodes all of the agent's history that it has access to.
- The belief state of an agent encapsulates the information about its past that it can use for current and future actions.

Functions implemented in a controller

For discrete time, a controller implements:

- a belief state transition function
  $remember : S \times P \rightarrow S$, where S is the set of belief states and P is the set of possible percepts.
  $s_{t+1} = remember(s_t, p_t)$ means that s_{t+1} is the belief state following belief state s_t when p_t is observed.
- A command function $do : S \times P \rightarrow C$, where S is the set of belief states, P is the set of possible percepts, and C is the set of possible commands.
  $c_t = do(s_t, p_t)$ means that the controller issues command c_t when the belief state is s_t and p_t is observed.