IIT Jodhpur

Biological Vision and Applications
Module 06-03: Reactive Agents

Hiranmay Ghosh

Reactive Agent

- An agent senses the environment and reacts to it
    - An autonomous car applies the brakes on seeing a pedestrian in the front
- Reactive behavior is spontaneous and immediate
    - Pre-attentive, without deliberation

Can we model a common room AC as a Reactive Agent?

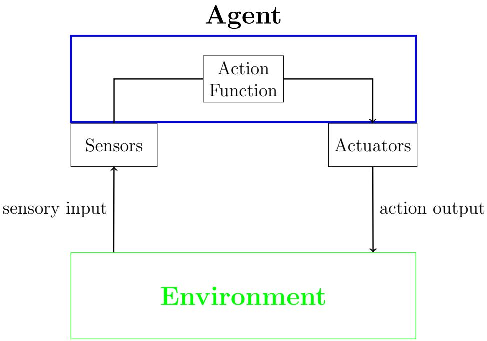
- Properties of a room AC
  - Autonomous: works without human intervention
  - Interactive: Senses and acts on the environment
  - Goal: to keep the room cool (implicit)
  - Social capability: communicates with humans
  - Knowledge: Knows how to do it's job (implicit)
  - Learning: Not applicable
- By definition, it is an agent
- It may an overkill to model such simple system as an agent
- Nevertheless, we shall use it as an example in this class

Characterizing the environment

- Accessible? … **Yes**
  - We assume that the temperature sensor provides adequate information about the temperature of the room
- Deterministic? … **No**
  - A door / window may be open
- Episodic? … **No**
  - AC is continuously working for it's lifetime
- Static or dynamic? … **Dynamic**
  - The environment can change even without any action of the AC
- Discrete or Continuous? … **Discrete**
  - Though temperature of the room can vary continuously, we model it as a finite set $\{OK, HOT\}$

Purely reactive agent

Agent without memory



- Agent has no memory
    - Does not remember history / experience
- Action is determined by the current environment state alone
- $action : S \to A$

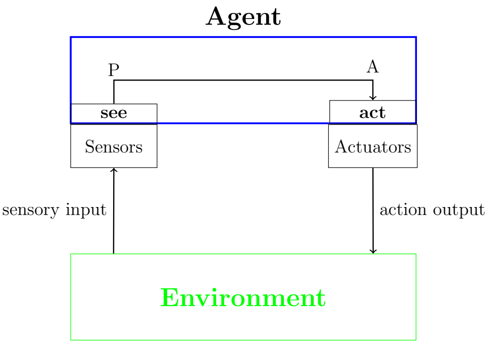
Purely reactive agent

Example: A Room AC

- Environment states: $S = \{OK, HOT\}$
- Actions: $A = \{START, STOP\}$
- Agent behavior: $action : S \rightarrow A$
  - $action : OK \rightarrow STOP$
  - $action : HOT \rightarrow START$
- Environment model: $env : S \times A \rightarrow \rho(S)$
  - ... Independent of agent behavior
  - $env : OK, START \rightarrow \{OK, HOT\}$
  - $env : OK, STOP \rightarrow \{OK, HOT\}$
  - $env : HOT, START \rightarrow \{OK, HOT\}$
  - $env : HOT, STOP \rightarrow \{HOT\}$
    - We assume that it is not snowing outside! ☺

Modeling Perception

Purely reactive agent



- "see" maps an environment state to a percept: $see(s) = p$
  - $see : S \rightarrow P$
- If $s_1 \neq s_2$, but $see(s_1) = see(s_2)$, i.e. $p_1 = p_2$
  - Environment states s_1 and s_2 are indistinguishable for the agent
  - ... e.g., there can be a person in the room that the AC cannot see
- We revise our stand
  - Agent behavior is determined by percept
  - $action : P \rightarrow A$ (NOT $S \rightarrow A$)

Purely reactive agent with perception

Example: A Room AC

- Person in room: Y
- Environment states: $S = \{(OK, Y), (OK, !Y), (HOT, Y), (HOT, !Y)\}$
- AC has a thermostat only – cannot see if somebody is in the room
- Percepts: $S \to P$
    - $see(OK, Y) \to ok$
    - $see(OK, !Y) \to ok$
    - $see(HOT, Y) \to hot$
    - $see(HOT, !Y) \to hot$
- The states (OK, Y) and $(OK, !Y)$ are indistinguishable, both lead to percept ok
    - Similarly, for (HOT, Y) and $(HOT, !Y)$

Perception for room AC

Continued

- Agent behavior: $action : P \rightarrow A$ ** not $S \rightarrow A$
  - $action : ok \rightarrow STOP$
  - $action : hot \rightarrow START$
- Environment model: $env : S \times A \rightarrow \rho(S)$
  - $env : (OK, Y), START \rightarrow \{(OK, Y), (HOT, Y), (OK, !Y), (HOT, !Y)\}$
  - $env : (OK, !Y), START \rightarrow \{(OK, Y), (HOT, Y), (OK, !Y), (HOT, !Y)\}$
  - ...

An AC with a camera

... In addition to a thermometer

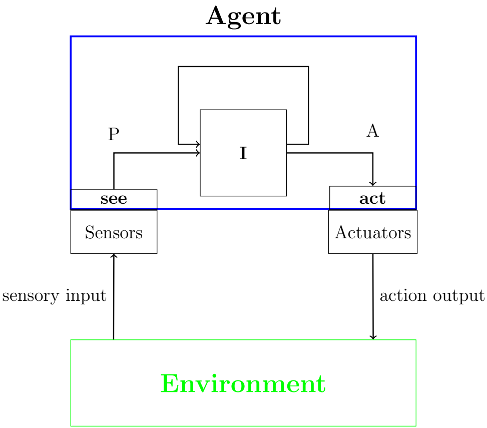
- Environment states: ** Same as earlier
  - $S = \{(OK, Y), (OK, !Y), (HOT, Y), (HOT, !Y)\}$
- Percepts: $S \rightarrow P$
  - $see(OK, Y) \rightarrow ok, y$
  - $see(OK, !Y) \rightarrow ok, !y$
  - $see(HOT, Y) \rightarrow hot, y$
  - $see(HOT, !Y) \rightarrow hot, !y$

A room AC with a camera

Continued

- The AC should switch on only if the AC sees a person in the room and perceives the temperature to be hot
- Agent behavior: $action : P \rightarrow A$
  - $action : ok, y \rightarrow STOP$
  - $action : ok, !y \rightarrow STOP$
  - $action : hot, y \rightarrow START$
  - $action : hot, !y \rightarrow STOP$
- Environment model: $env : S \times A \rightarrow \rho(S)$ ** Same as earlier
  - $env : (OK, Y), START \rightarrow \{(OK, Y), (HOT, Y), (OK, !Y), (HOT, !Y)\}$
  - $env : (OK, !Y), START \rightarrow \{(OK, Y), (HOT, Y), (OK, !Y), (HOT, !Y)\}$
  - ...

Reactive agent with memory



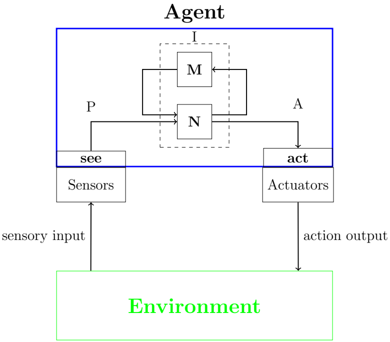
- $see : S \to P$
- $next : P \times I \to I$
- $action : I \to A$

- I stores the history
  - Memory is finite
  - How many steps?
  - Abstracted form?

Reactive agent with memory

Example: A Room AC (no camera) with a constraint

- If it is already ON, it cannot START, and vice-versa

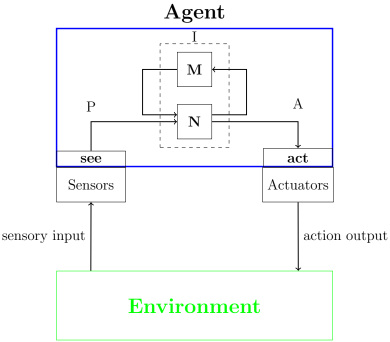


- $next : P \times M \rightarrow N$
- $mem : N \rightarrow M$, $action : N \rightarrow A$

- Percepts: $P = \{ok, hot\}$
- Actions:
  $A = \{START, STOP, NOP\}$
- Memory: $M = \{ON, OFF\}$
- Agent behavior $next : P \times M \rightarrow N$
  - $next(ok, ON) = (STOP, OFF)$
  - $next(ok, OFF) = (NOP, OFF)$
  - $next(hot, ON) = (NOP, ON)$
  - $next(hot, OFF) = (START, ON)$

Example: A Room AC (no camera) with a constraint

… Continued



- $next : P \times M \to N$
- $mem : N \to M$, $action : N \to A$

- Memory: $N \to M$
  - $mem(STOP, OFF) \to OFF$
  - $mem(NOP, OFF) \to OFF$
  - $mem(NOP, ON) \to ON$
  - $mem(START, ON) \to ON$
- Action: $N \to A$
  - $action(STOP, OFF) \to OFF$
  - $action(NOP, OFF) \to NOP$
  - $action(NOP, ON) \to NOP$
  - $action(START, ON) \to ON$

Can we model a Reactive Agent as a table look up?

Example: A Room AC

- Agent behavior: $N = (A, M)$
  - $next(ok, ON) = (STOP, OFF)$
  - $next(ok, OFF) = (NOP, OFF)$
  - $next(hot, ON) = (NOP, ON)$
  - $next(hot, OFF) = (START, ON)$

- Table lookup for *next*

	ok	hot
ON	$STOP, OFF$	NOP, ON
OFF	NOP, OFF	$START, ON$

- Theoretically possible, but may need extremely large tables

Quiz

Quiz 06-03

End of Module 06-03