

CS440/ECE 448 Lecture 3: Agents and Rationality

Slides by Svetlana Lazebnik, 9/2016

Modified by Mark Hasegawa-Johnson, 1/2019

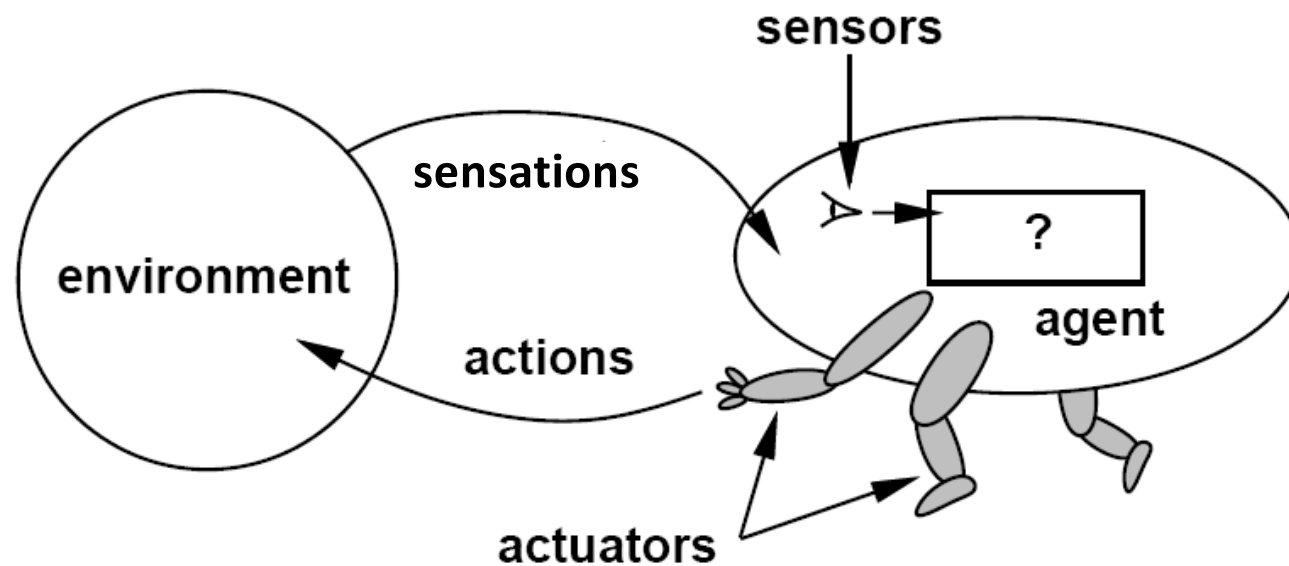


Contents

- Agent = Performance, Environment, Actions, Sensors (PEAS)
- What makes an agent ***Rational***?
- What makes an agent ***Autonomous***?
- Types of Agents: Reflex, Internal-State, Goal-Directed, Utility-Directed (RIGU)
- Properties of Environments: Observable, Deterministic, Episodic, Static, Continuous (ODESC)

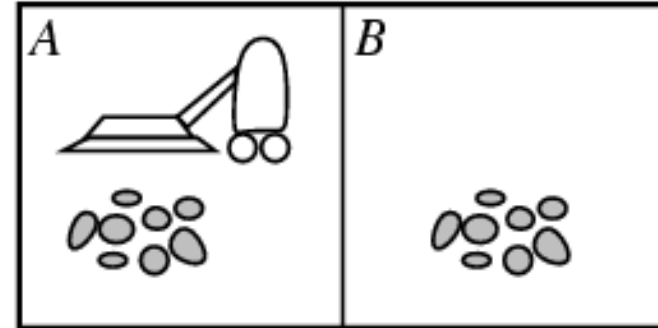
Agents

- An **agent** is anything that can be viewed as **perceiving** its **environment** through **sensors** and **acting** upon that environment through **actuators**



Example: Vacuum-Agent

- **Environment = tuple of variables:**
 - Location, status of both rooms,
e.g., $S = \{ \text{Loc}=A, \text{Status}=(\text{Dirty}, \text{Dirty}) \}$
- **Action = variable drawn from a set:**
 $A \in \{ \text{Left}, \text{Right}, \text{Suck}, \text{NoOp} \}$
- **Sensors = tuple of variables:**
 - Location, and status of Current Room Only
e.g., $S = \{ \text{Loc}=A, \text{Status} = \text{Dirty} \}$



function Vacuum-Agent([location,status]) returns an **action**

- *if* **Loc=A**
 - *if* **Status=Dirty** then return **Suck**
 - *else if* I have never visited **B** then return **Right**
 - *else* return **NoOp**
- *else*
 - *if* **Status=Dirty** then return **Suck**
 - *else if* I have never visited **A** then return **Left**
 - *else* return **NoOp**

Specifying the task environment

- **PEAS: Performance, Environment, Actions, Sensors**
- **P:** a function the agent is maximizing (or minimizing)
 - Assumed given
- **E:** a formal representation for *world states*
 - For concreteness, a tuple $(var_1=val_1, var_2=val_2, \dots, var_n=val_n)$
- **A:** actions that change the state according to a *transition model*
 - Given a state and action, what is the successor state (or distribution over successor states)?
- **S:** observations that allow the agent to infer the world state
 - Often come in very different form than the state itself
 - E.g., in tracking, observations may be pixels and state variables 3D coordinates

PEAS Example: Autonomous taxi

- How does it measure its performance?

$$P = \textit{profit} + \textit{customer_satisfaction}$$

s.t. no laws broken?

- What is its environment?

Quantify? What variables, in what format?

- What are the actuators?

- What are the sensors?

Another PEAS example: Spam filter

- Performance measure

Is a false accept as expensive as a false reject?

Performance per e-mail, or in aggregate?

- Environment

User's e-mail account? Server hosting thousands of users?

- Actuators?

- Sensors?

Performance Measure

- An agent's performance is measured by some performance or **utility** measure
- Utility = function of the current environment E_t , and of the history of all actions from time 1 to time t , $A_{1:(t-1)}$:

$$U_t = f(E_t, A_{1:(t-1)})$$

- Example: autonomous vacuum cleaner:

$$U_t = \# \text{ currently dirty rooms} - \frac{1}{2}(\# \text{ movements so far})$$

Example Problem: Spam Filter

Consider a Spam Filter. Design an environment (a set of variables E_t , some of which may be unobservable by the agent), an action variable A_t , and a performance variable (utility) U_t .

Specify the form of the equation by which U_t depends on A_t and E_t . Make sure that U_t summarizes the costs of all actions from A_1 through A_t . Make sure that U_t expresses the idea that false acceptance (mislabeling non-spam as spam) is not as expensive as false rejection (mislabeling spam as non-spam).

Possible answer:

$$E_t = \{M_1, \dots, M_t, L_1, \dots, L_t\}$$

M_t = text of t'th e-mail

$L_t = 1$ if t'th e-mail is spam, else $L_t = 0$

$A_t = 1$ if spam filter rejects t'th e-mail, else $A_t = 0$

$$U_t = - \sum_{\tau=1}^t C_{FR} A_{\tau} (1 - L_{\tau}) + C_{FA} L_{\tau} (1 - A_{\tau})$$

Where C_{FA} is the cost of a false acceptance, and C_{FR} is the cost of a false rejection.

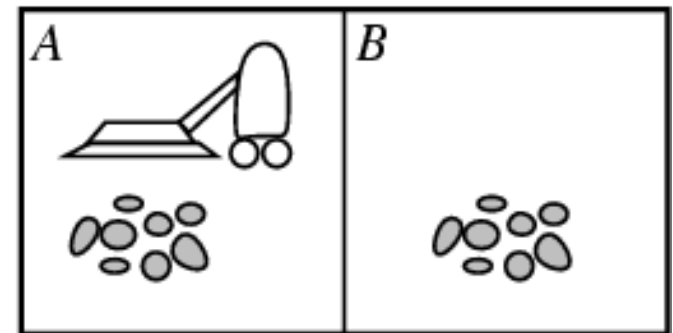
What makes an agent *Rational*?

- RATIONAL AGENT is one with RATIONAL BEHAVIOR: For each possible percept sequence, a **rational agent** should select an action that is expected to maximize its **performance measure**, *given the evidence provided by the percept sequence and the agent's built-in knowledge*
- RATIONAL THOUGHT: all of the above, + agent calculates and maximizes the performance measures in its own brain.
- **Performance measure (utility function):**
An objective criterion for success of an agent's behavior
- Example Exam Problems:
 - Can a rational agent make mistakes?
 - Does rational behavior require rational thought?

Back to the Vacuum-Agent

function Vacuum-Agent([location,status]) returns an **action**

- *if* **Loc=A**
 - *if* **Status=Dirty** then return **Suck**
 - *else if* I have never visited **B** then return **Right**
 - *else* return **NoOp**
- *else*
 - *if* **Status=Dirty** then return **Suck**
 - *else if* never visited **A** then **Left**
 - *else* return **NoOp**



- Example Exam Problem: Is this agent ***Rational?***

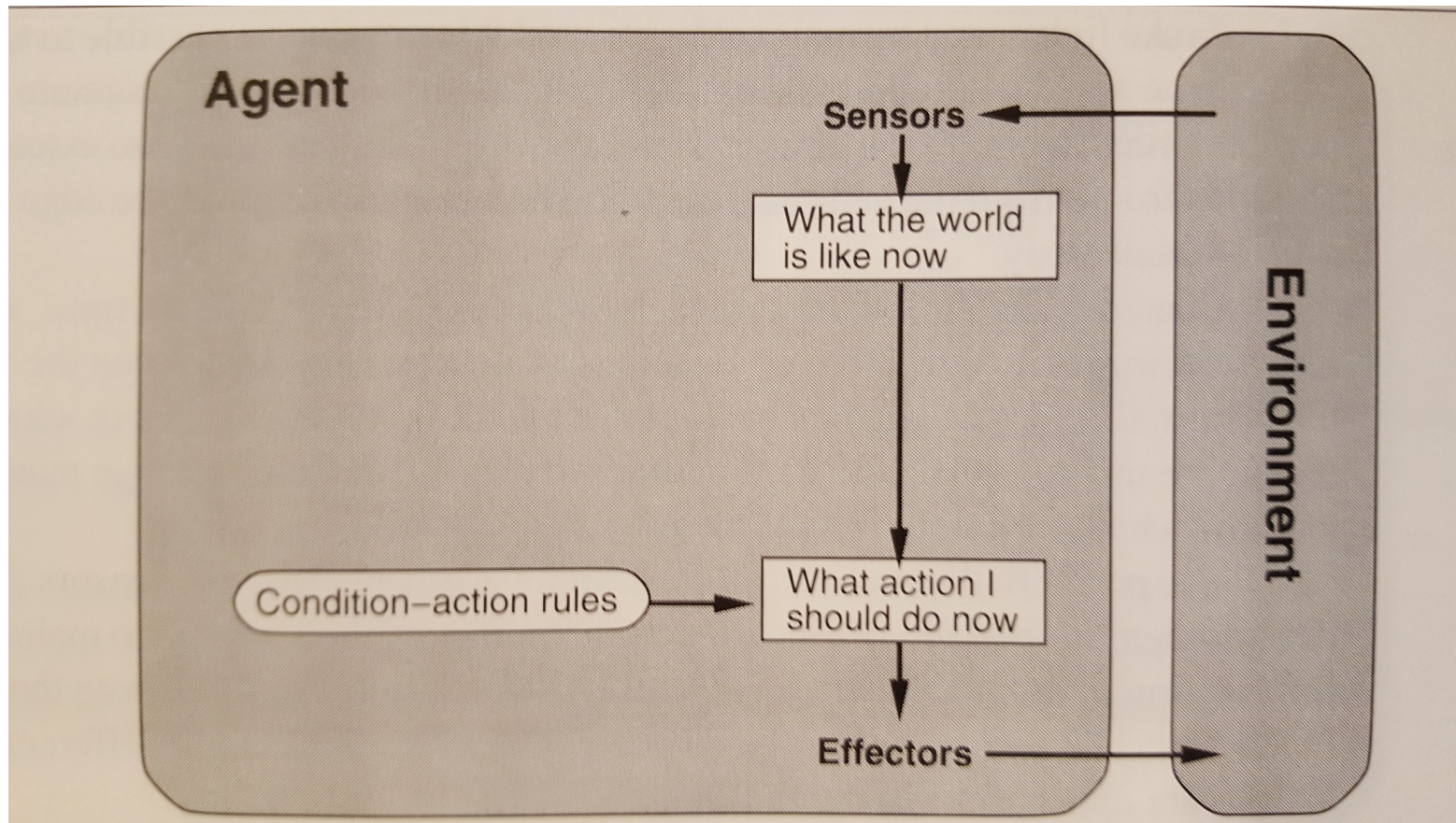
What makes an agent *Autonomous*?

- Russell & Norvig: “A system is autonomous to the extent that its behavior is determined by its own experience.”
- A Rational Agent might not be Autonomous, if its designer was capable of foreseeing the maximum-utility action for every environment.
- Example: Vacuum-Agent

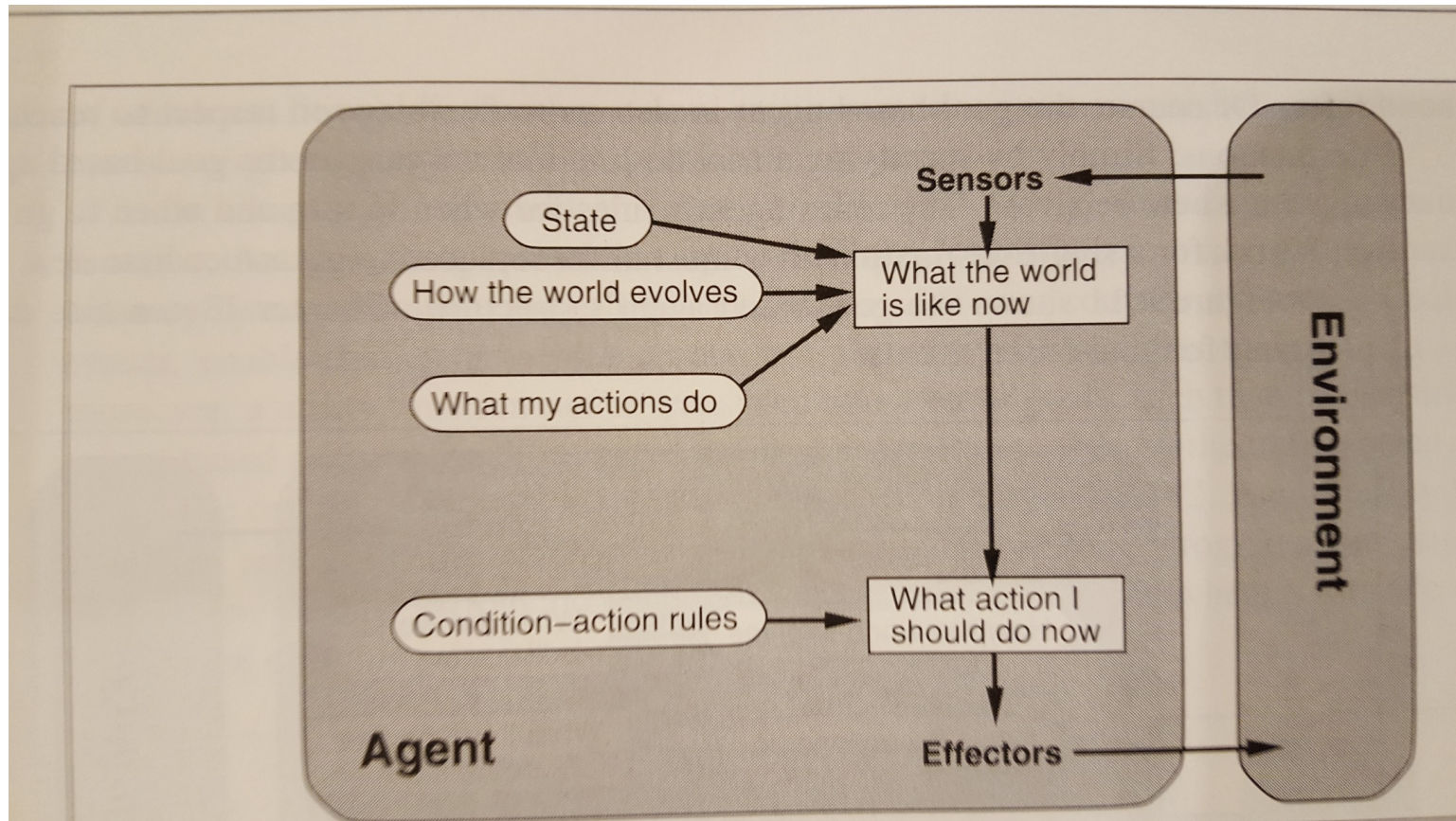
Types of Agents

- Reflex agent: no concept of past, future, or value
 - Might still be Rational, if the environment is known to the designer with sufficient detail
- Internal-State agent: knows about the past
- Goal-Directed agent: knows about the past and future
- Utility-Directed agent: knows about past, future, and value

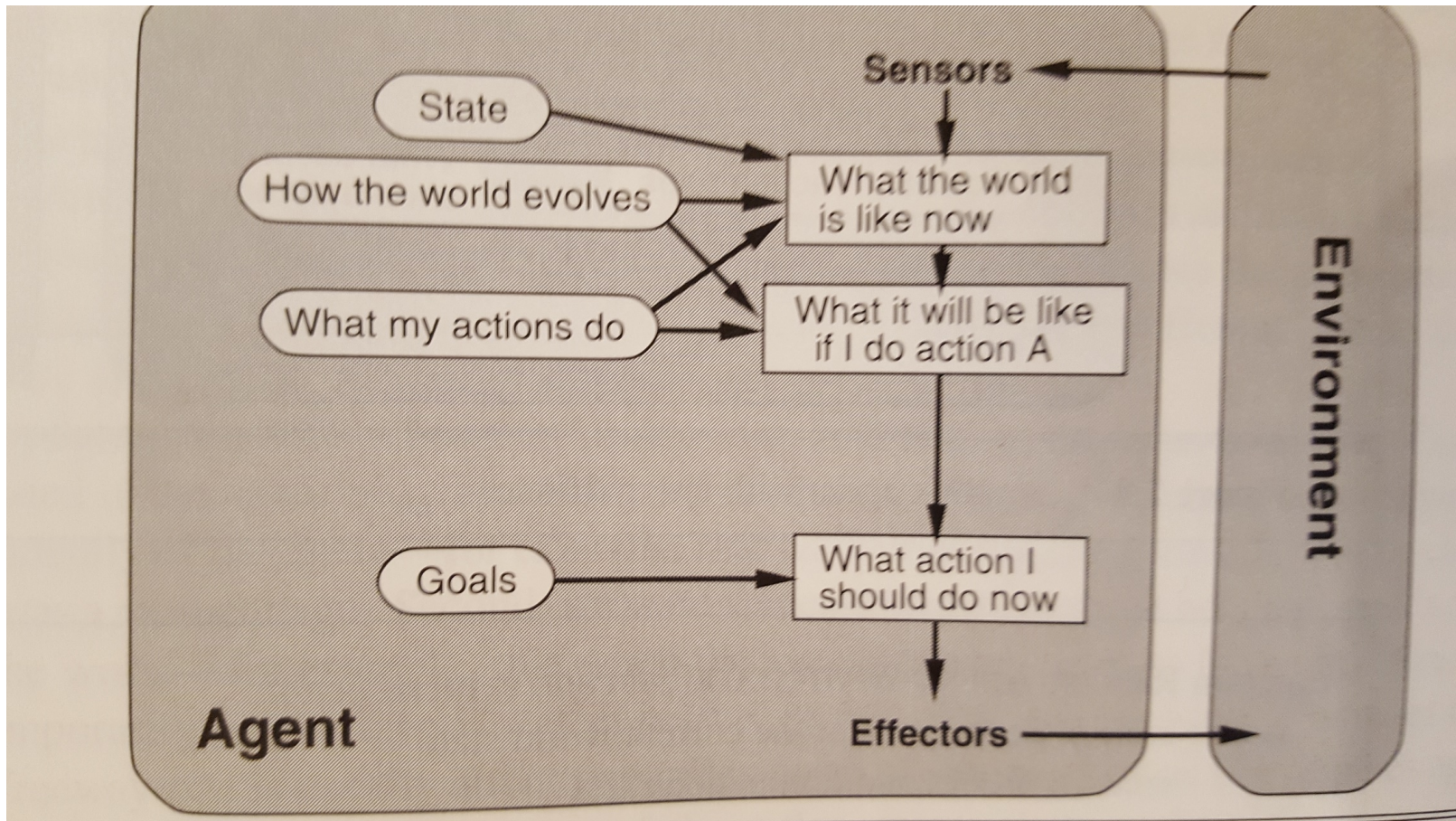
Reflex Agent



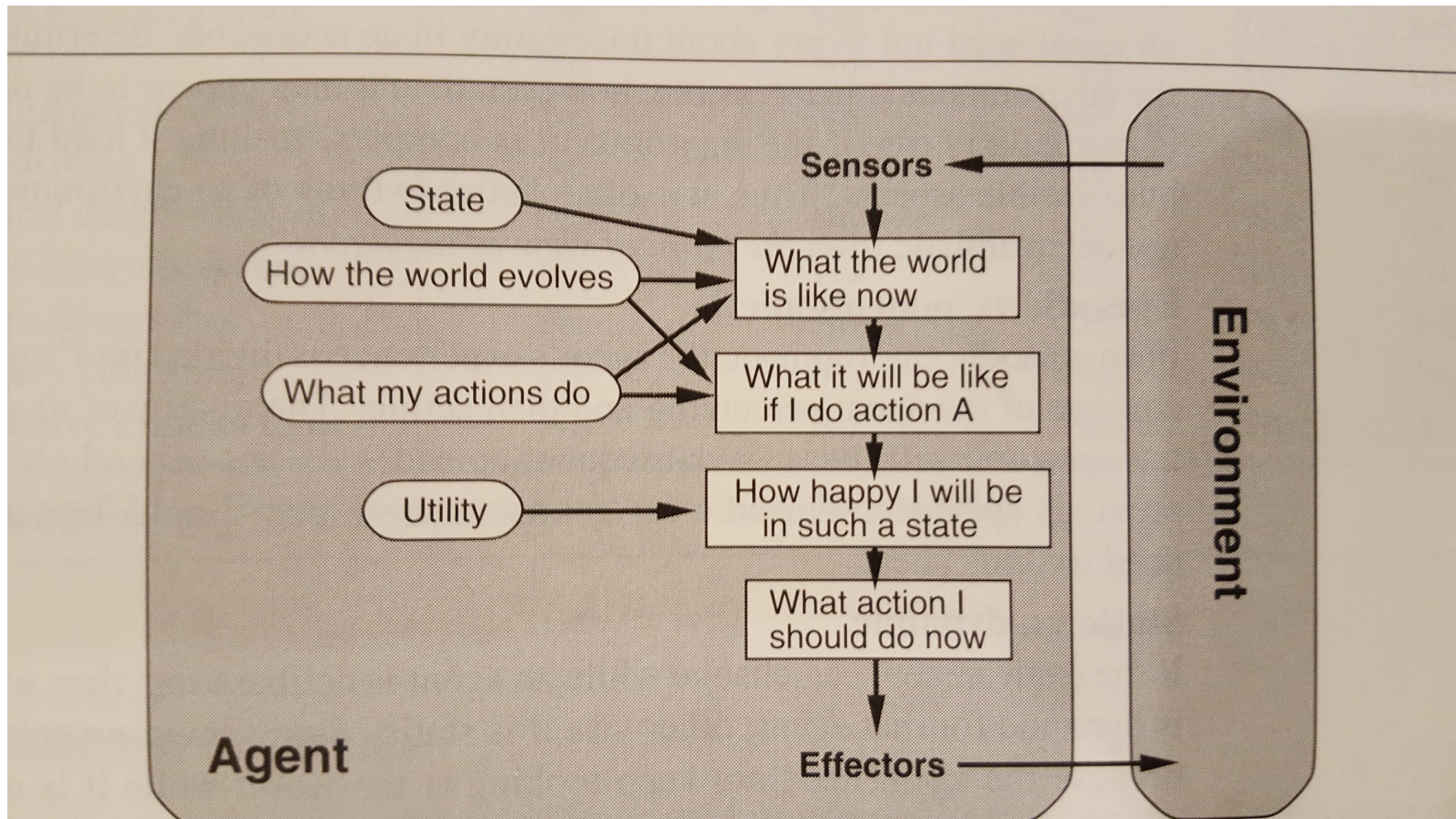
Internal-State Agent



Goal-Directed Agent



Utility-Directed Agent



PEAS

- **P**erformance measure: Determined by the system designer, attempts to measure some intuitive description of behavior goodness.
- **A**ctions: Determined by the system designer, usually trades off cost versus utility
- **S**ensors: Determined by the system designer, usually trades off cost versus utility
- **E**nvironment: Completely out of the control of the system designer.

Properties of Environments

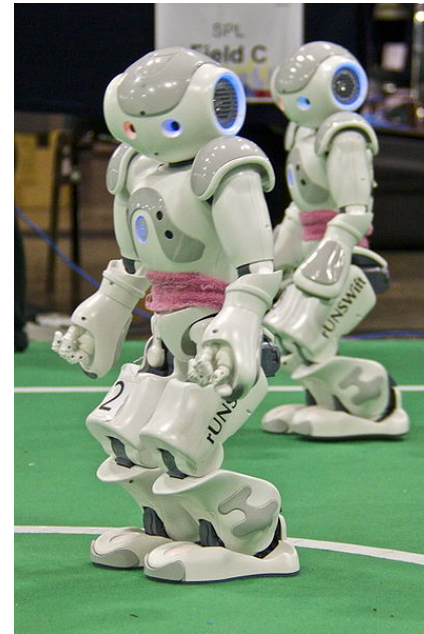
- Fully observable vs. partially observable
- Deterministic vs. stochastic
- Episodic vs. sequential
- Static vs. dynamic
- Discrete vs. continuous
- Single agent vs. multi-agent
- Known vs. unknown

Fully observable vs. partially observable

- Do the agent's sensors give it access to the complete state of the environment?
 - For any given world state, are the values of all the variables known to the agent?



VS.



Source: L. Zettlemoyer

Deterministic vs. stochastic

- Is the next state of the environment completely determined by the **current state** and the **agent's action**?
 - Is the transition model **deterministic** (unique successor state given current state and action) or **stochastic** (distribution over successor states given current state and action)?
 - **strategic**: the environment is deterministic except for the actions of other agents



vs.

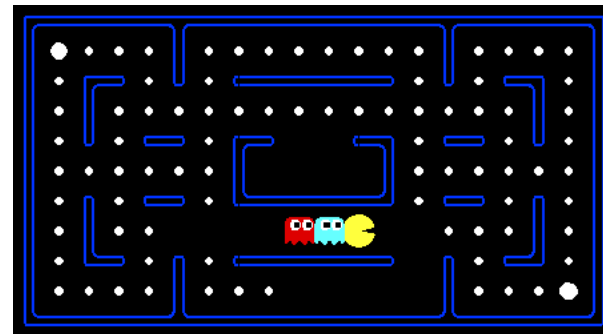


Episodic vs. sequential

- Is the agent's experience divided into unconnected episodes, or is it a coherent sequence of observations and actions?
 - Does each problem instance involve just one action or a series of actions that change the world state according to the transition model?



VS.



Static vs. dynamic

- Is the world changing while the agent is thinking?
 - **Semidynamic:** the environment does not change with the passage of time, but the agent's performance score does



vs.

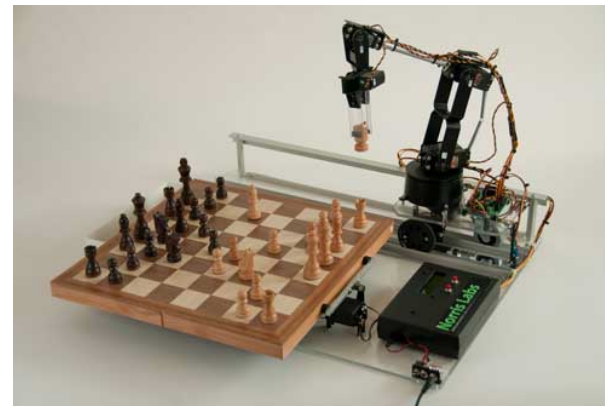


Discrete vs. continuous

- Does the environment provide a countable (discrete) or uncountably infinite (continuous) number of distinct percepts, actions, and environment states?
 - Are the values of the state variables discrete or continuous?
 - Time can also evolve in a discrete or continuous fashion
 - “Distinct” = different values of utility



VS.

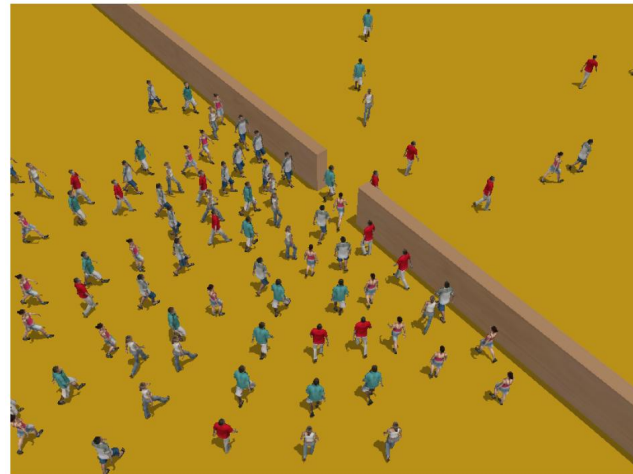


Single-agent vs. multiagent

- Is an agent operating by itself in the environment?



vs.



Known vs. unknown

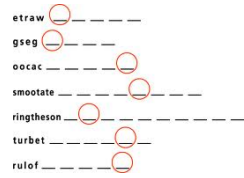
- Are the rules of the environment (transition model and rewards associated with states) known to the agent?
 - Strictly speaking, not a property of the environment, but of the agent's state of knowledge



vs.



Examples of different environments



Word jumble solver



Chess with a clock



Scrabble



Autonomous driving

Observable

Fully

Fully

Partially

Partially

Deterministic

Deterministic

Strategic

Stochastic

Stochastic

Episodic

Episodic

Sequential

Sequential

Sequential

Static

Static

Semidynamic

Static

Dynamic

Discrete

Discrete

Discrete

Discrete

Continuous

Single agent

Single

Multi

Multi

Multi

Preview of the course

- **Deterministic environments:** search, constraint satisfaction, logic
 - Can be **sequential** or **episodic**
- **Strategic environments:** minimax search, games
 - Might be either **deterministic** (e.g., chess) or **stochastic** (e.g., poker)
 - Might be **fully observable** (e.g., chess) or **partially observable** (e.g., battleship)
- **Stochastic environments**
 - **Episodic:** Bayesian networks, pattern classifiers
 - **Sequential, known:** Markov decision processes
 - **Sequential, unknown:** reinforcement learning