Artificial Intelligence

Chapter 1

Outline

◇ Course overview

◇ What is AI?

◇ A brief history

◇ The state of the art

Administrivia

Class home page: `http://www-inst.eecs.berkeley.edu/~cs188`
for lecture notes, assignments, exams, grading, office hours, etc.

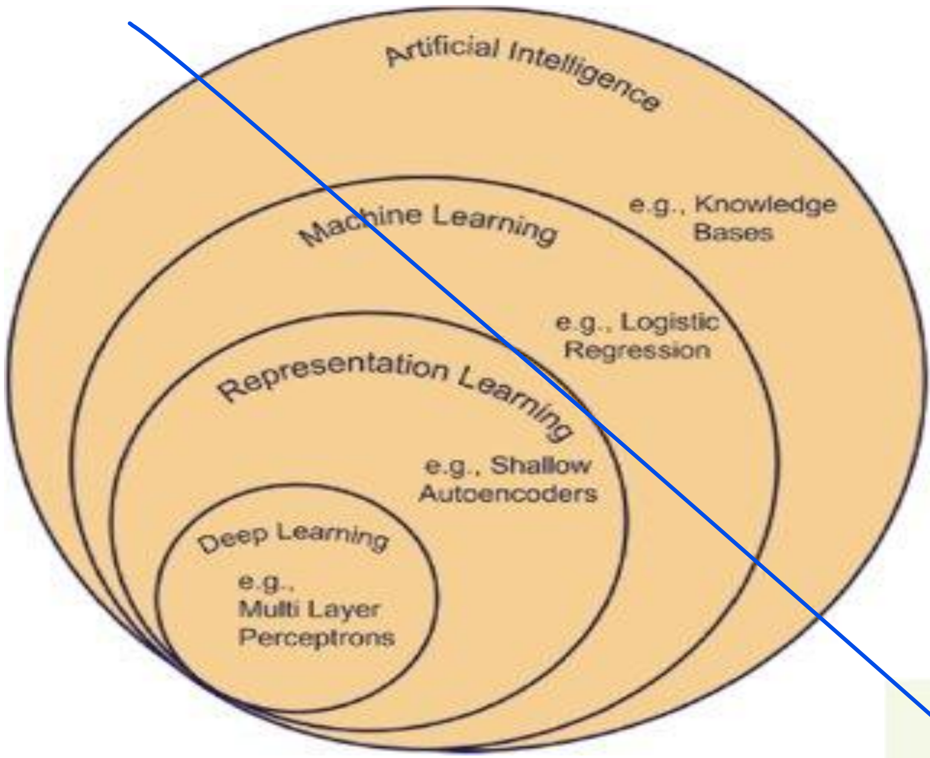
Assignment 0 (lisp refresher) due 8/31

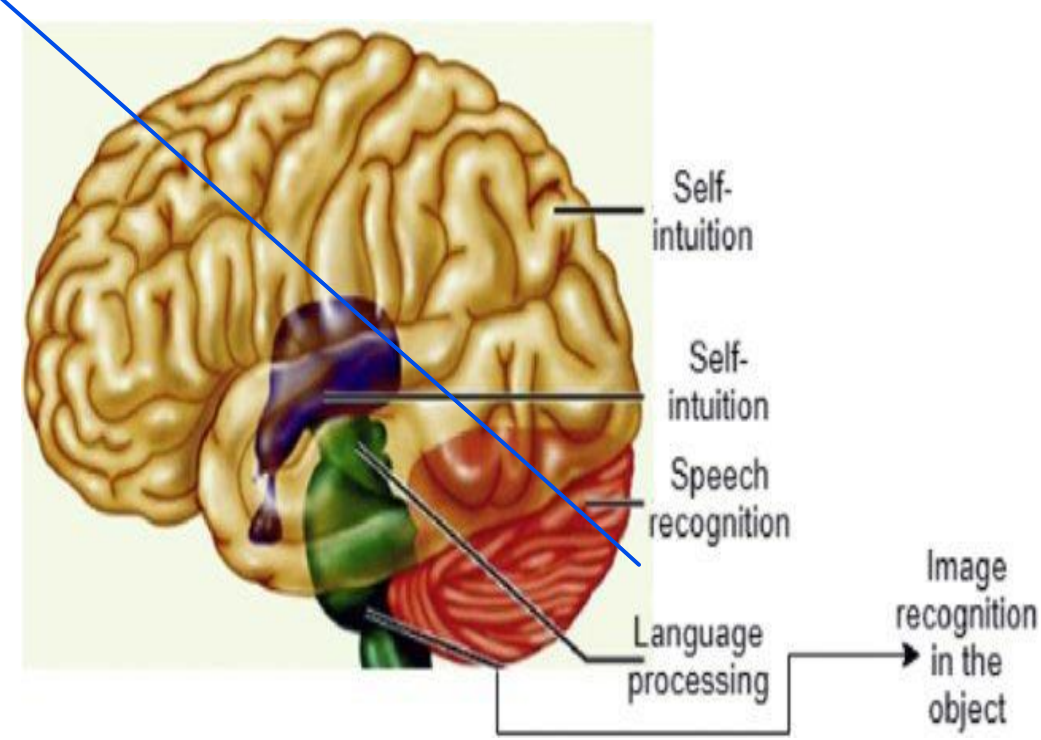
Book: Russell and Norvig <u>Artificial Intelligence: A Modern Approach</u>
Read Chapters 1 and 2 for this week's material

Code: integrated lisp implementation for AIMA algorithms at
`http://www-inst.eecs.berkeley.edu/~cs188/code/`

Course overview

- ♢ intelligent agents
- ♢ search and game-playing
- ♢ logical systems
- ♢ planning systems
- ♢ uncertainty—probability and decision theory
- ♢ learning
- ♢ language
- ♢ perception
- ♢ robotics
- ♢ philosophical issues


Artificial Intelligence
e.g., Knowledge Bases
Machine Learning
e.g., Logistic Regression
Representation Learning
e.g., Shallow Autoencoders
Deep Learning
e.g., Multi Layer Perceptrons


Self-intuition
Self-intuition
Speech recognition
Language processing
Image recognition in the object

What is AI?

"[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning ..." (Bellman, 1978)	"The study of mental faculties through the use of computational models" (Charniak+McDermott, 1985)
"The study of how to make computers do things at which, at the moment, people are better" (Rich+Knight, 1991)	"The branch of computer science that is concerned with the automation of intelligent behavior" (Luger+Stubblefield, 1993)

Views of AI fall into four categories:

Thinking humanly	Thinking rationally
Acting humanly	Acting rationally

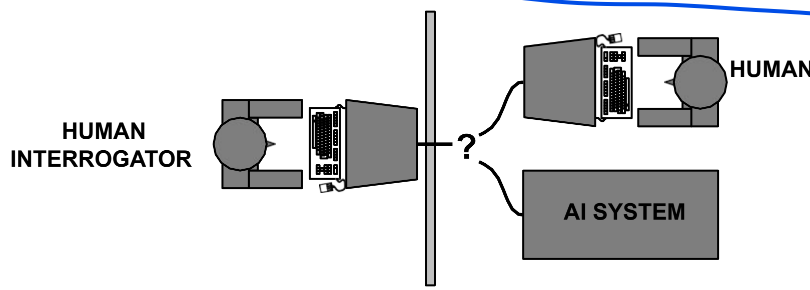
Examining these, we will plump for acting rationally (sort of)

Acting humanly: The Turing test

Turing (1950) "Computing machinery and intelligence":

◇ "Can machines think?" ⟶ "Can machines behave intelligently?"

◇ Operational test for intelligent behavior: the Imitation Game



◇ Predicted that by 2000, a machine might have a 30% chance of fooling a lay person for 5 minutes

◇ Anticipated all major arguments against AI in following 50 years

◇ Suggested major components of AI: knowledge, reasoning, language understanding, learning

Problem: Turing test is not reproducible, constructive, or amenable to mathematical analysis

Thinking humanly: Cognitive Science

1960s "cognitive revolution": information-processing psychology replaced prevailing orthodoxy of behaviorism

Requires scientific theories of internal activities of the brain
- What level of abstraction? "Knowledge" or "circuits"?
- How to validate? Requires
  1) Predicting and testing behavior of human subjects (top-down)
  or 2) Direct identification from neurological data (bottom-up)

Both approaches (roughly, Cognitive Science and Cognitive Neuroscience) are now distinct from AI

Thinking rationally: Laws of Thought

Normative (or prescriptive) rather than descriptive

Aristotle: what are correct arguments/thought processes?

Several Greek schools developed various forms of logic:
notation and rules of derivation for thoughts;
may or may not have proceeded to the idea of mechanization

Direct line through mathematics and philosophy to modern AI

Problems:
1) Not all intelligent behavior is mediated by logical deliberation
2) What is the purpose of thinking? What thoughts should I have?

Acting rationally

Rational behavior: doing ~~the right thing~~

The right thing: that which is expected to maximize goal achievement, given the available information

Doesn't necessarily involve thinking—e.g., blinking reflex—but thinking should be in the service of rational action

Aristotle (Nicomachean Ethics):

> *Every art and every inquiry, and similarly every action and pursuit, is thought to aim at some good*

Rational agents

An agent is an entity that perceives and acts

This course is about designing rational agents

Abstractly, an agent is a function from percept histories to actions:

$$f : \mathcal{P}^* \rightarrow \mathcal{A}$$

For any given class of environments and tasks, we seek the agent (or class of agents) with the best performance

Caveat: computational limitations make perfect rationality unachievable
$\rightarrow$ design best program for given machine resources

AI prehistory

Philosophy	logic, methods of reasoning mind as physical system foundations of learning, language, rationality
Mathematics	formal representation and proof algorithms computation, (un)decidability, (in)tractability probability
Psychology	adaptation phenomena of perception and motor control experimental techniques (psychophysics, etc.)
Linguistics	knowledge representation grammar
Neuroscience	physical substrate for mental activity
Control theory	homeostatic systems, stability simple optimal agent designs

Potted history of AI

1943	McCulloch & Pitts: Boolean circuit model of brain
1950	Turing's "Computing Machinery and Intelligence"
1952–69	Look, Ma, no hands!
1950s	Early AI programs, including Samuel's checkers program, Newell & Simon's Logic Theorist, Gelernter's Geometry Engine
1956	Dartmouth meeting: "Artificial Intelligence" adopted
1965	Robinson's complete algorithm for logical reasoning
1966–74	AI discovers computational complexity Neural network research almost disappears
1969–79	Early development of knowledge-based systems
1980–88	Expert systems industry booms
1988–93	Expert systems industry busts: "AI Winter"
1985–95	Neural networks return to popularity
1988–	Resurgence of probabilistic and decision-theoretic methods Rapid increase in technical depth of mainstream AI "Nouvelle AI": ALife, GAs, soft computing

State of the art

Which of the following can be done at present?

- ◇ Play a decent game of table tennis
- ◇ Drive along a curving mountain road
- ◇ Drive in the center of Cairo
- ◇ Play a decent game of bridge
- ◇ Discover and prove a new mathematical theorem
- ◇ Write an intentionally funny story
- ◇ Give competent legal advice in a specialized area of law
- ◇ Translate spoken English into spoken Swedish in real time

Intelligent Agents

Chapter 2

Outline

◇ PAGE (Percepts, Actions, Goals, Environment)

◇ Environment types

◇ Agent functions and programs

◇ Agent types

◇ Vacuum world

Introduction : Agents

Definition :

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

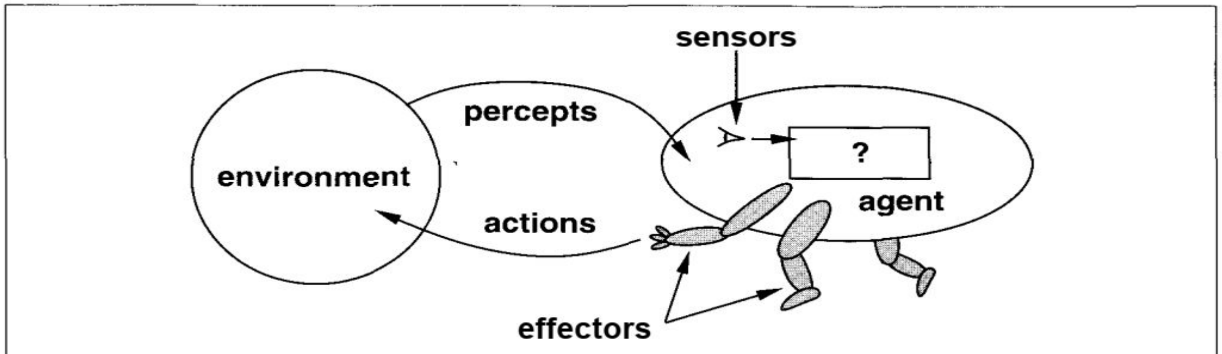


Figure 2.1 Agents interact with environments through sensors and effectors.

what is rational at any given time **depends on four things:**

- The performance measure that defines degree of success.
- Everything that the agent has perceived so far (**percept sequence**)
- What the agent knows about the environment - **prior knowledge about the environment**.
- The actions that the agent can perform.

Components of AGENTS : PAGE

Must first specify the setting for intelligent agent design

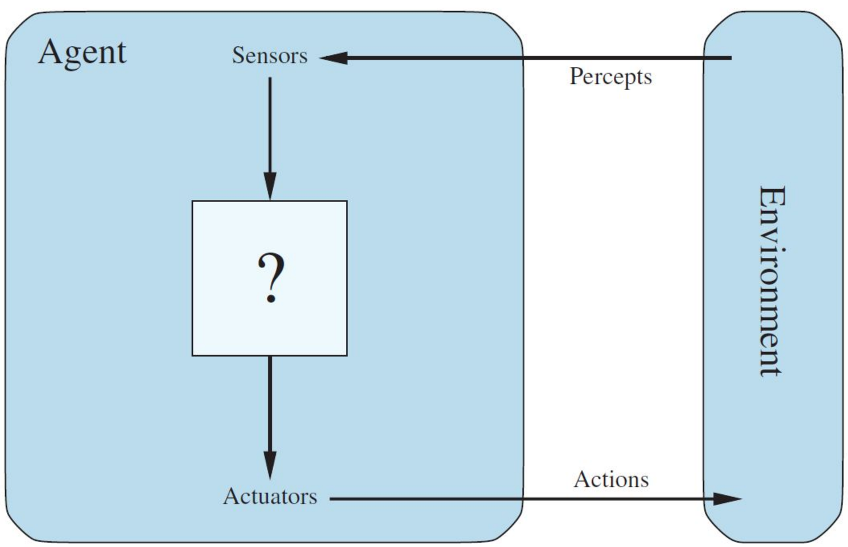
Consider, e.g., the task of designing an automated taxi:

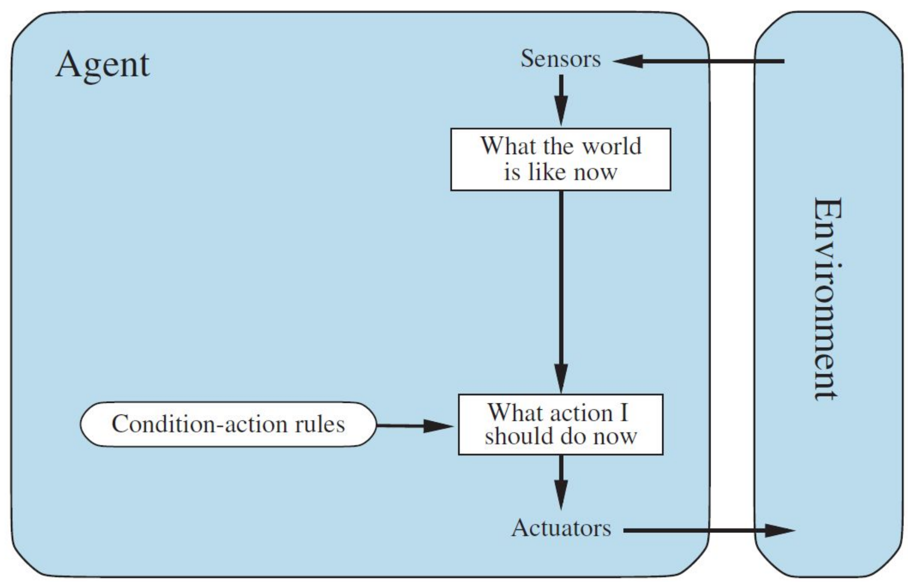
Percepts??

Actions??

Goals??

Environment??





PAGE

Must first specify the setting for intelligent agent design

Consider, e.g., the task of designing an automated taxi:

Percepts?? video, accelerometers, gauges, engine sensors, keyboard, GPS, ...

Actions?? steer, accelerate, brake, horn, speak/display, ...

Goals?? safety, reach destination, maximize profits, obey laws, passenger comfort, ...

Environment?? US urban streets, freeways, traffic, pedestrians, weather, customers, ...

Internet shopping agent

Percepts??

Actions??

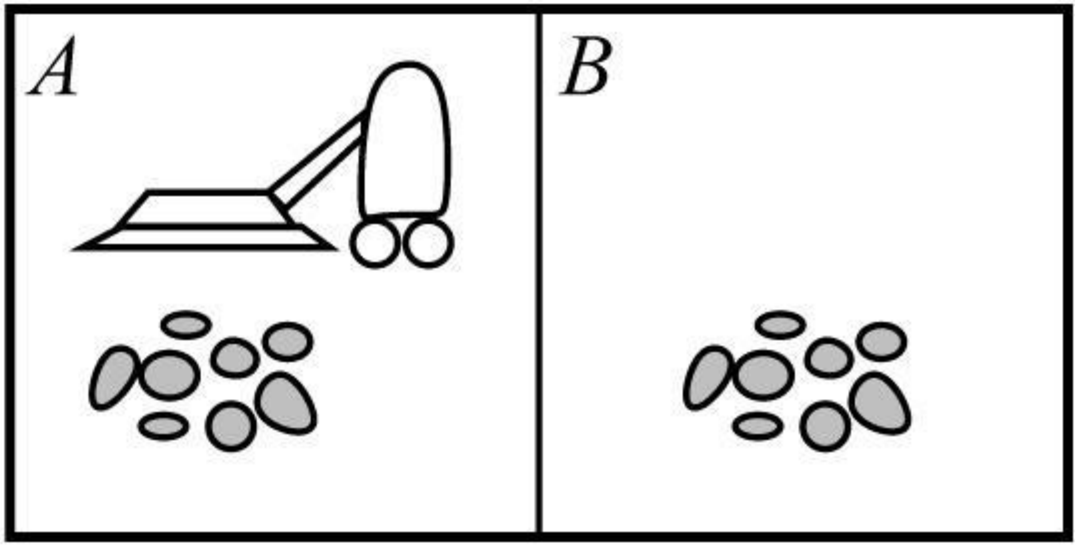
Goals??

Environment??

Agent Type	Percepts	Actions	Goals	Environment
Medical diagnosis system	Symptoms, findings, patient's answers	Questions, tests, treatments	Healthy patient, minimize costs	Patient, hospital
Satellite image analysis system	Pixels of varying intensity, color	Print a categorization of scene	Correct categorization	Images from orbiting satellite
Part-picking robot	Pixels of varying intensity	Pick up parts and sort into bins	Place parts in correct bins	Conveyor belt with parts
Refinery controller	Temperature, pressure readings	Open, close valves; adjust temperature	Maximize purity, yield, safety	Refinery
Interactive English tutor	Typed words	Print exercises, suggestions, corrections	Maximize student's score on test	Set of students

Figure 2.3 Examples of agent types and their PAGE descriptions.

Vacuum-cleaner world with Two Environments



- Percepts: location and status , e.g., [A,Dirty]
- Actions: *Left*, *Right*, *Suck*, *NoOp*
- *Partial Tabulation of agent function*

Percept sequence	Action
$[A, Clean]$	$Right$
$[A, Dirty]$	$Suck$
$[B, Clean]$	$Left$
$[B, Dirty]$	$Suck$
$[A, Clean]$, $[A, Clean]$	$Right$
$[A, Clean]$, $[A, Dirty]$	$Suck$
$\vdots$	$\vdots$

Rational agents

Without loss of generality, "goals" specifiable by performance measure defining a numerical value for any environment history

Rational action: whichever action maximizes the expected value of the performance measure given the percept sequence to date

Definition : A rational Agent is one that does the right thing conceptually

Performance of Good Agent :

measured based on,

1) it is rational at different circumstance,
2) no need to move in the environment once all the dirt is cleaned
3) If clean square become dirt, the agent should check and clean

Structure of Intelligent Agents:

- The job of AI is to design an **agent program** that implements the **agent function** the mapping from the percepts to action.

- The program will run on some sort of computing devices with physical sensors and actuators . This is called as **Agent Architecture,**

Agent = Architecture + Agent Program

Architecture = the machinery that an agent executes on.
Agent Program = an implementation of an agent function.

Agent functions and programs

An agent is completely specified by the agent function
mapping percept sequences to actions

(In principle, one can supply each possible sequence to see what it does. Obviously, a lookup table would usually be immense.)

One agent function (or a small equivalence class) is rational

Aim: find a way to implement the rational agent function concisely

An agent program takes a single percept as input, keeps internal state:

function SKELETON-AGENT(*percept*) **returns** action
  static: *memory*, the agent's memory of the world

  memory ← UPDATE-MEMORY(*memory*, *percept*)
  action ← CHOOSE-BEST-ACTION(*memory*)
  memory ← UPDATE-MEMORY(*memory*, *action*)
  return *action*

AIMA code

The code for each topic is divided into four directories:

- `agents`: code defining agent types and programs
- `algorithms`: code for the methods used by the agent programs
- `environments`: code defining environment types, simulations
- `domains`: problem types and instances for input to algorithms

(Often run algorithms on domains rather than agents in environments.)

```
(setq joe (make-agent :name 'joe :body (make-agent-body)
                      :program (make-dumb-agent-program)))

(defun make-dumb-agent-program ()
  (let ((memory nil))
    #'(lambda (percept)
        (push percept memory)
        'no-op)))
```

Agent types

- The job of AI is to design an **agent program** that implements the **agent function** the mapping from the percepts to action.

- The program will run on some sort of computing devices with physical sensors and actuators . This is called as **Agent Architecture**.

Agent= Architecture + Agent Program

Agent Types:

Categories based on : How to build a real program to implement the mapping from percepts to action.

- Simple reflex agents
- Agents that keep track of the world
- Goal-based agents
- Utility-based agents

Agent

Sensors

What the world
is like now

Environment

What action I
should do now

Actuators

Simple Reflex Agent:

Intelligent agent that makes decisions based on the current percept without any internal representation of the world.

- **Condition-Action Rule** – It is a rule that maps a state (condition) to an action.

- **Characteristics:**
  - They choose **actions only based on the current percept.** without considering past experiences or future consequences.
  - Their environment is **completely observable.** (Lacks an internal model of the world or history of interactions.)
  - Makes decisions based on predefined rules or mappings from percepts to actions.

function SIMPLE-REFLEX-AGENT(*percept*) **returns** *action*
  static: *rules*, a set of condition-action rules

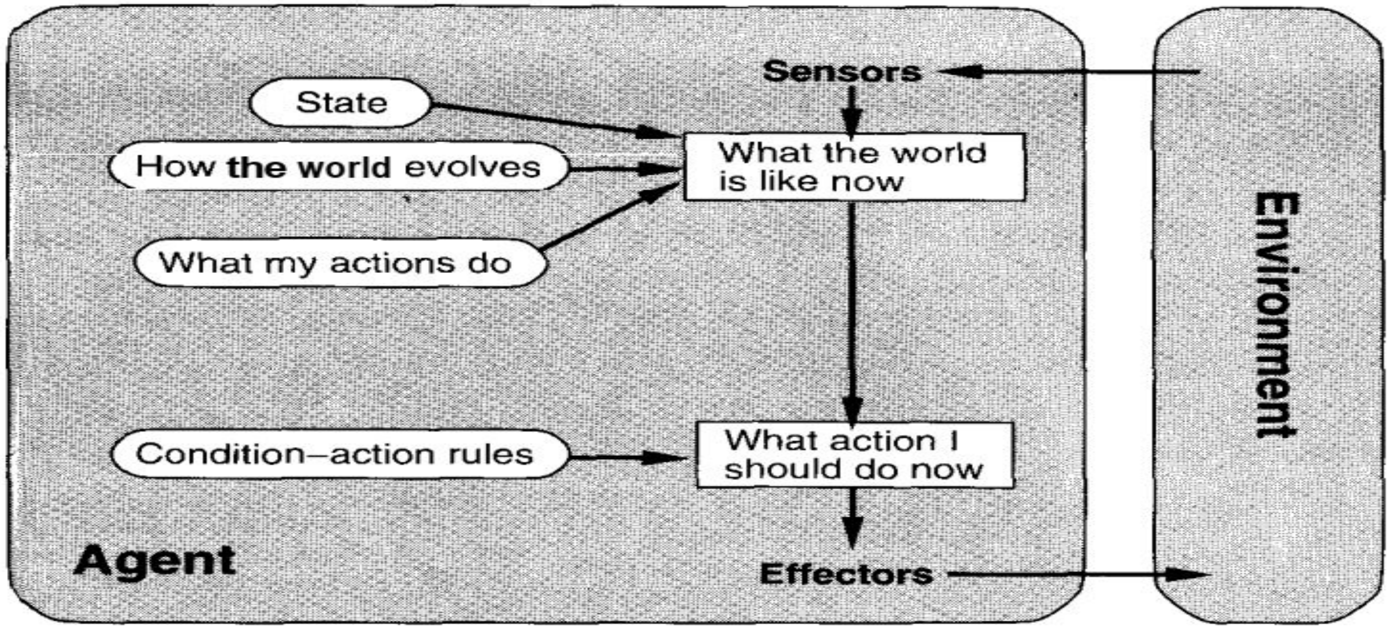
  state ← INTERPRET-INPUT(*percept*)
  rule ← RULE-MATCH(*state*, *rules*)
  action ← RULE-ACTION[*rule*]
  return *action*

The agent program for a simple reflex agent in the 2-state vacuum environment

function *REFLEX-VACUUM-AGENT([location,status])*
  returns *action*

  if *status* = *Dirty* **then return** *Suck*
  else if *location* = *A* **then return** *Right*
  else if *location* = B **then return** Left

Reflex agents with state



function REFLEX-AGENT-WITH-STATE(*percept*) **returns** *action*
  static: *state*, a description of the current world state
      rules, a set of condition-action rules

  state ← UPDATE-STATE(*state*, *percept*)
  rule ← RULE-MATCH(*state*, *rules*)
  action ← RULE-ACTION[*rule*]
  state ← UPDATE-STATE(*state*, *action*)
  return *action*

It is responsible for creating new internal state description

Figure 2.10 A reflex agent with internal state. It works by finding a rule whose condition matches the current situation (as defined by the percept and the stored internal state) and then doing the action associated with that rule.

Model-Based Reflex Agents

Definition : It maintains an internal **model or representation of the world** and **uses this** model to **make decisions** and plan actions.

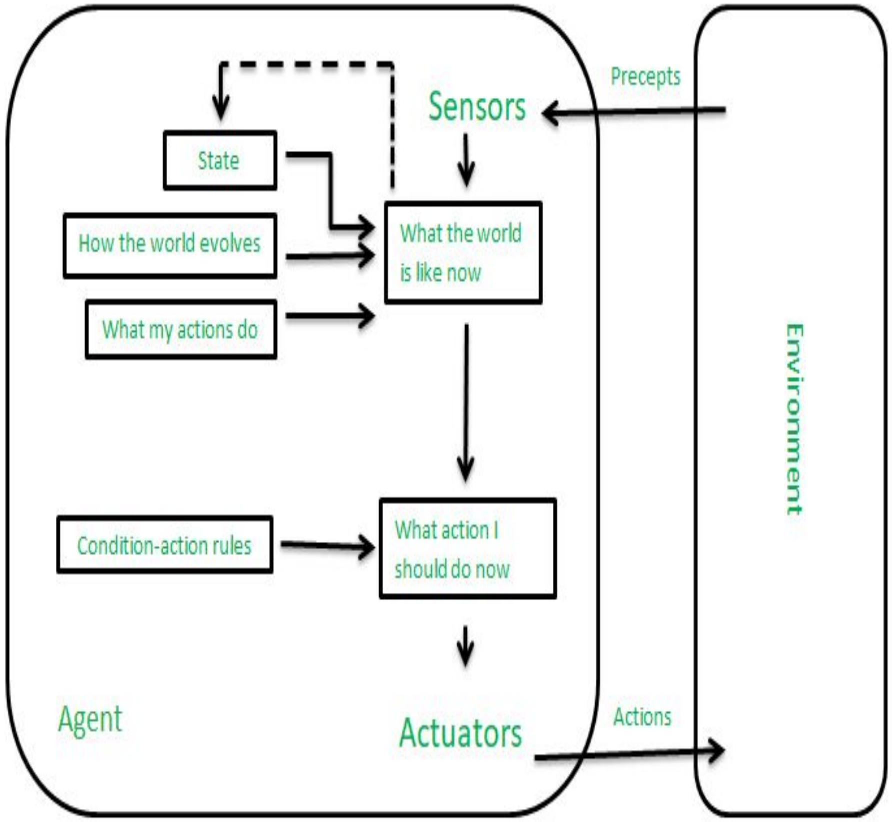
It can **work in a partially observable environment**, and track the situation.

Two important factors:
Model: It is knowledge about "how things happen in the world”
Internal State: It is a representation of unobserved aspects of current state depending on percept history.

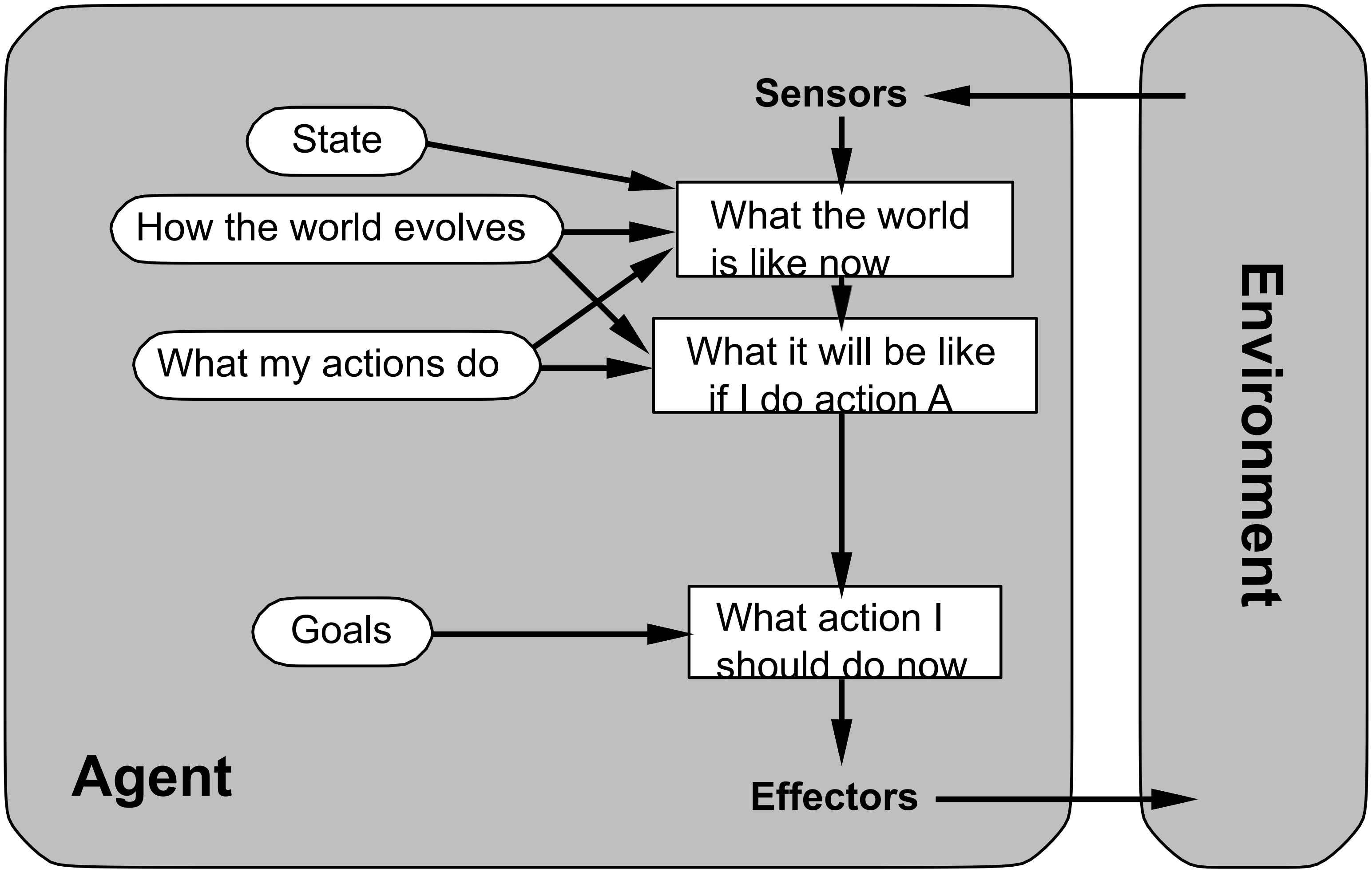
These agents have the model, "which is knowledge of the world" and based on the model they perform actions.

Updating the agent state requires information about:

How the world evolves
How the agent's action affects the world.



Goal-based agents



Goal-Based Agents

How It's evolve ? : Knowing about the **current state of the environment is not always enough** to decide what to do.

The agent needs to **know its goal** which describes **desirable situations.**

For example, at a road junction, the taxi can turn left, right, or go straight on. The right decision depends on where the taxi is trying to get to(Passengers destination)

Working : The agent program can combine this with! information about the results of possible actions in order to choose actions that achieve the goal.

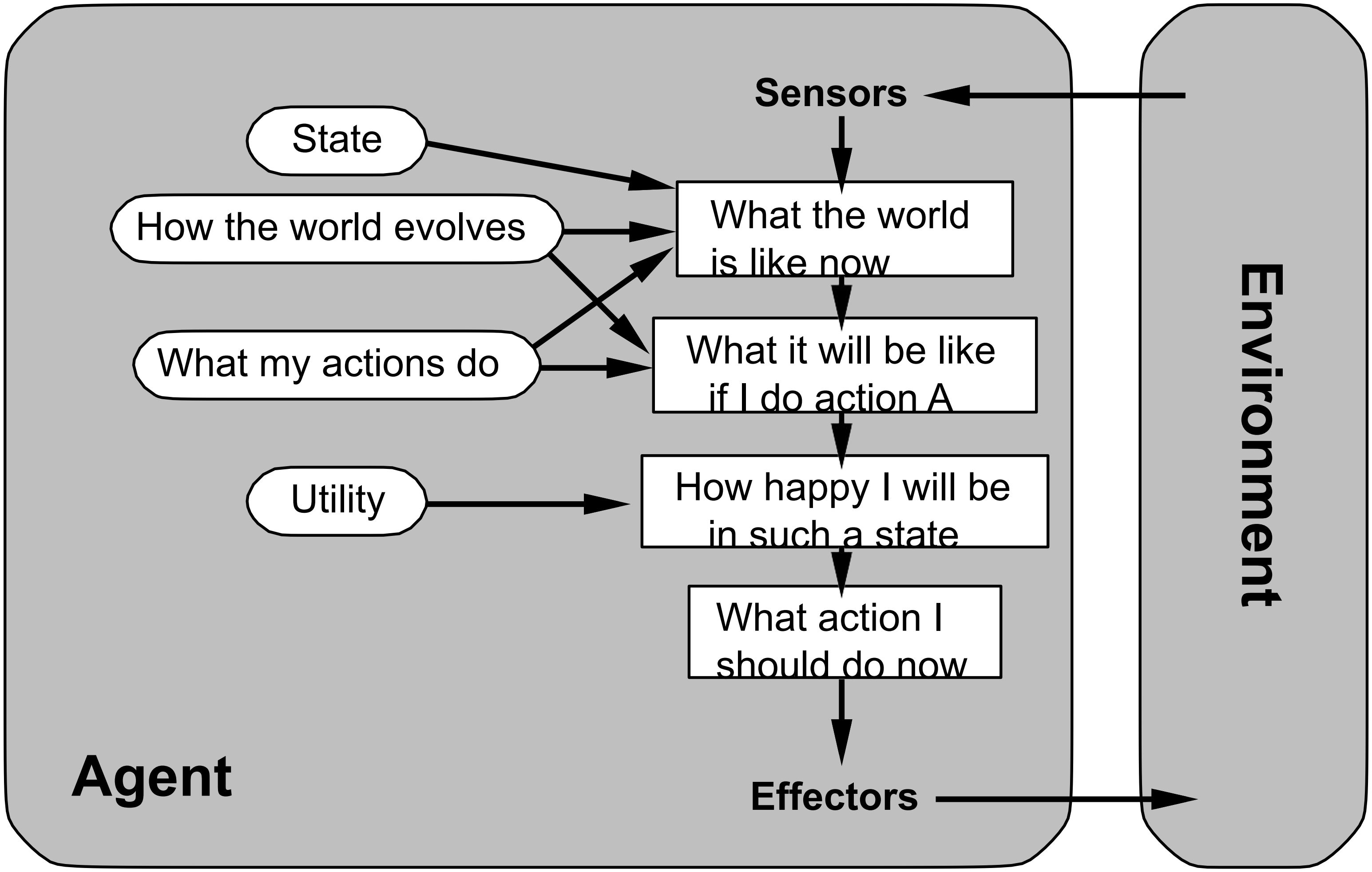
AI devoted to finding action sequences to achieve the agent's goals,
by, **1. Search** **2. Planning**

Decision making **depends** on not only the **current state also consider future**
The goal-based agent appears less efficient, it is far more flexible.

Comparison :

The reflex agent's rules for when to turn and when to go straight will only work for a single destination	**the goal-based agent is also more flexible with respect to reaching different destinations.**

Utility-based agents



Utility-Based Agents

How It's evolve ? : Similar to the goal-based agent but provide an extra component of utility measurement which makes them different by providing a measure of success at a given state. “**Goals alone are not really enough to generate high-quality behavior”.**

Example factors : like speed, safety, reliability, or cost.

Utility is a function that maps a state to a real number, indicating the associated degree of happiness or satisfaction for the agent.

Working : It allows for a comparison of different world states or sequences of states based on the degree of happiness they would bring to the agent..

Comparison :

utility-based agents consider the overall utility of actions,	goal-based agents focus on immediate goal satisfaction.
In some cases, a utility function can be translated into a set of goals, resulting in identical decisions between utility-based and goal-based agents	

Agent Environment in AI (or) Properties of environments

- **Accessible** vs. **inaccessible:**

  If an agent can obtain complete and accurate information about the state's environment, then such an environment is called an Accessible environment else it is called inaccessible.

  Ex : Measure the temperature of the room is accessible but for earth is inaccessible

- **Deterministic** vs. **nondeterministic:**

  If an agent's current state and selected action can completely determine the next state of the environment, then such environment is called a deterministic environment.

  A stochastic environment is **random** in nature and cannot be determined completely by an agent.

  In a deterministic, fully observable environment, agent does not need to worry about uncertainty.

- **Episodic** vs. **nonepisodic**

  In an episodic environment, there is a series of one-shot actions, and only the current percept is required for the action.

  However, in Sequential environment, an agent requires memory of past actions to determine the next best actions.

Agent Environment (Types)in AI (or) Properties of environments

- **Static vs Dynamic:**

  If the environment can change itself while an agent is deliberating then such environment is called a dynamic environment else it is called a static environment.

  Static environments are easy to deal because an agent does not need to continue looking at the world while deciding for an action.

  Taxi driving is an example of a dynamic environment whereas Crossword puzzles are an example of a static environment.

Discrete vs Continuous:

If in an environment there are a finite number of percepts and actions that can be performed within it, then such an environment is called a discrete environment else it is called continuous environment.

A chess game comes under discrete environment as there is a finite number of moves that can be performed.

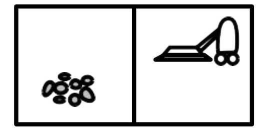
A self-driving car is an example of a continuous environment.

Environment	Accessible	Deterministic	Episodic	Static	Discrete
Chess with a clock	Yes	Yes	No	Semi	Yes
Chess without a clock	Yes	Yes	No	Yes	Yes
Poker	No	No	No	Yes	Yes
Backgammon	Yes	No	No	Yes	Yes
Taxi driving	No	No	No	No	No
Medical diagnosis system	No	No	No	No	No
Image-analysis system	Yes	Yes	Yes	Semi	No
Part-picking robot	No	No	Yes	No	No
Refinery controller	No	No	No	No	No
Interactive English tutor	No	No	No	No	Yes

Figure 2.13 Examples of environments and their characteristics.

The vacuum world

`code/agents/environments/vacuum.lisp`



Percepts `(<bump> <dirt> <home>)`

Actions `shutoff forward suck (turn left) (turn right)`

Goals (performance measure on environment history)
- +100 for each piece of dirt cleaned up
- -1 for each action
- -1000 for shutting off away from home

Environment
- grid, walls/obstacles, dirt distribution and creation, agent body
- movement actions work unless bump into wall
- suck actions put dirt into agent body (or not)

Accessible? Deterministic? Episodic? Static? Discrete?