Summary of agents kinds

- A **rational** agent is one that does the right thing
- The **omniscience** of an agent knows the *actual outcome assuming infinite knowledge*
- The **autonomy** of an agent is the extent to which its behaviour is determined by its own experience

Summary of agents kinds

- A **rational** agent is one that does the right thing
- The **omniscience** of an agent knows the *actual outcome assuming infinite knowledge*
- The **autonomy** of an agent is the extent to which its behaviour is determined by its own experience

Environment types

Environments in which agents operate can be defined in different ways

- Fully observable (vs. partially observable):
  - In a fully observable environment all of the environment relevant to the action being considered is observable. In such environments, the agent does not need to keep track of the changes in the environment. Ex. Chess
  - In a partially observable environment, the relevant features of the environment are only partially observable.

Environment types

- Deterministic (vs. stochastic):
  - The next state of the environment is completely determined by the current state and the action executed by the agent.
  - If an element of interference or uncertainty occurs then the environment is stochastic.
  - (If the environment is deterministic; i.e. wholly determined by the preceding state except for the actions of other agents, then the environment is strategic)

Environment types

- Episodic (vs. sequential):
  - The agent's experience is divided into atomic "episodes" (each episode consists of the agent perceiving and then performing a single action), and the choice of action in each episode depends only on the episode itself.
  - In a sequential environment, the agent engages in a series of connected episodes.

Environment types

- Episodic (vs. sequential):
  - The agent's experience is divided into atomic "episodes" (each episode consists of the agent perceiving and then performing a single action), and the choice of action in each episode depends only on the episode itself.
  - In a sequential environment, the agent engages in a series of connected episodes.

Environment types

- Static (vs. dynamic):
  - A static environment does not change while the agent is thinking.
  - The time as an agent deliberates is irrelevant.
  - The agent doesn't need to observe the world during deliberation.
  - A Dynamic Environment changes over time independent of the actions of the agent (if an agent does not respond in a timely manner, this counts as a choice to do nothing)

Environment types

- Discrete (vs. continuous):
  - A limited number of distinct, clearly defined percepts and actions is discrete, otherwise it is continuous.
- Single agent (vs. multiagent):
  - An agent operating by itself in an environment.
  - If the environment contains other intelligent agents, the agent needs to be concerned about strategic

Knowledge of Environment

- Knowledge of Environment (World)
  - Different to sensory information from environment
- World knowledge can be (pre)-programmed in
  - Can also be updated/inferred by sensory information
- Choice of actions informed by knowledge of world
  - Current state of the world
  - Previous states of the world
  - How its actions change the world
- Example: Chess agent
  - World knowledge is the board state (all the pieces)
  - Sensory information is the opponents move
  - Its moves also change the board state

Specifying the task environment PEAS

- In designing an agent, the first step must always be to specify the task environment, i.e. PEAS
- PEAS: Performance measure, Environment, Actuators, Sensors
- Consider a medical diagnosis agent

Agent Type	Performance Measure	Environment	Actuators	Sensors
Medical diagnosis system	Healthy patient, minimize costs, lawsuits	Patient, hospital, staff	Display questions, tests, diagnoses, treatments, referrals	Keyboard entry of symptoms, findings, patient's answers

PEAS: Taxi Deriver

- Consider the task of designing an automated taxi driver:
- A fully automated taxi is beyond the capabilities of existing technology
- The fully driving task is extremely open-ended

Agent Type	Performance Measure	Environment	Actuators	Sensors
Taxi driver	Safe: fast, legal, comfortable trip, maximize profits	Roads, other traffic, pedestrians, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS, odometer, accelerometer, engine sensors, keyboard

Figure 2.4 PEAS description of the task environment for an automated taxi.

PEAS: Tutor

- Agent: Interactive English tutor
- Performance measure: Maximize student's score on test
- Environment: Set of students
- Actuators: Screen display (exercises, suggestions, corrections)
- Sensors: Keyboard

Internet shopping agent

Performance measure?? price, quality, appropriateness, efficiency

Environment?? current and future WWW sites, vendors, shippers

Actuators?? display to user, follow URL, fill in form

Sensors?? HTML pages (text, graphics, scripts)

Agent architectures

Agent = Architecture + Program

- Program
  - Method of turning environmental input into actions
  - An agent program maps from percept to action and updates its internal state.
- Architecture
  - Hardware/software (OS etc.) on which agent's program runs

➢ Table based agent

➢ Percept based agent or reflex agent

➢ Subsumption Architecture

Table based agent

- In table based agent the action is looked up from a table based on information about the agent's percepts. A table is simple way to specify a mapping from percepts to actions. The mapping is implicitly defined by a program. The mapping may be implemented by a rule based system, by a neural network or by a procedure.
- There are several disadvantages to a table based system. The tables may become very large. Learning a table may take a very long time, especially if the table is large. Such systems usually have little autonomy, as all actions are pre-determined.

Table based agent

- Drawbacks:
  - Huge table
  - Take a long time to build the table
  - No autonomy
  - Even with learning, need a long time to learn the table entries

Percept based agent (reflex agent)

- In percept based agents,
  1. information comes from **sensors - percepts**
  2. changes the agents current **state of the world**
  3. triggers **actions through the effectors**
- Such agents are called reactive agents or stimulus-response agents. Reactive agents have no notion of history. The current state is as the sensors see it right now. The action is based on the current percepts only.
- The following are some of the characteristics of percept-based agents.
  - ➢ Efficient
  - ➢ No internal representation for reasoning, inference.
  - ➢ No strategic planning, learning.
  - ❖ Not good for multiple, opposing, goals.

Subsumption Architecture

- This architecture is based on reactive systems. Brooks(1986) notes that in lower animals there is no deliberation and the actions are based on sensory inputs. But even lower animals are capable of many complex tasks. His argument is to follow the evolutionary path and build simple agents for complex worlds.
- The Subsumption Architecture built in layers.

The main features of Brooks' architecture are.

- ➢ There is no explicit knowledge representation
- ➢ Behaviour is distributed, not centralized
- ➢ Response to stimuli is reflexive
- ➢ The design is bottom up, and complex behaviours are fashioned from the combination of simpler underlying ones.
- ➢ Individual agents are simple

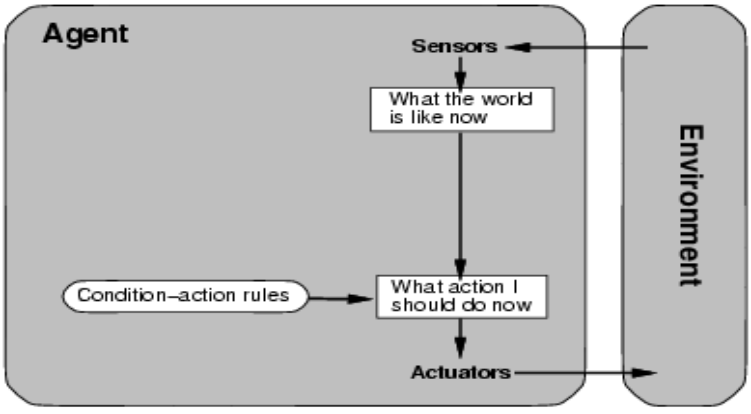
Agent types

Four basic types :

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents

Simple reflex agents

The simplest kind of agent is the simple reflex agent. These agents select actions on the basis AGENT
of the *current percept, ignoring the rest of the percept history.*

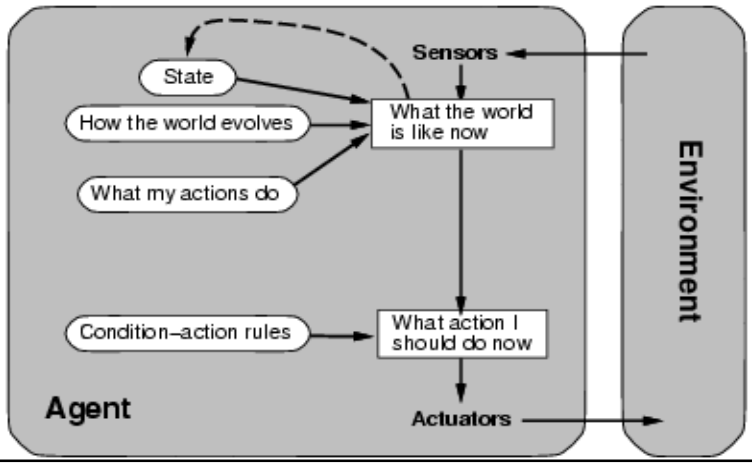


Model-based reflex agents

- The most effective way to handle partial observability is to *keep track of part of the world it can't see now.*
- *The agent should maintain some sort of* ***internal*** **state** that depends on the **percept history** and reflects at least some of the unobserved aspects of the current state.
- State based agents differ from percept based agents in that such agents maintain some sort of state based on the percept sequence received so far.
- The state is updated regularly based on what the agent senses, and the agent's actions.
- Keeping track of the state requires that the agent has knowledge about how the world evolves, and how the agent's actions affect the world.

Model-based reflex agents

• information comes from **sensors - percepts**
• based on this, the agent changes the current **state of the world**
• based on state of the world and knowledge (memory), it fires **actions through the Actuators**

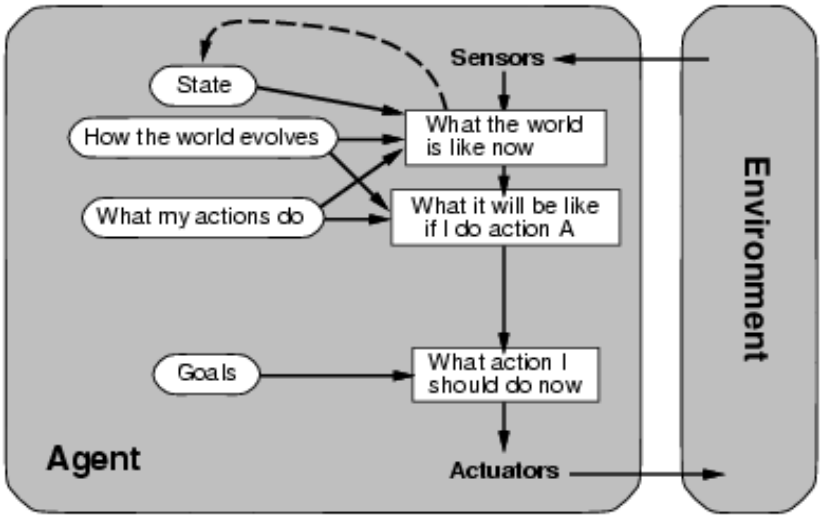


Goal-based agents

The goal based agent has some goal which forms a basis of its actions.
Such agents work as follows:
• information comes from **sensors - percepts**
• changes the agents current **state of the world**
• based on **state of the world and knowledge (memory) and goals/intentions, it chooses actions and does them through the Actuators.**

Remark:
Goal formulation based on the current situation is a way of solving many problems and search is a universal problem solving mechanism in AI.

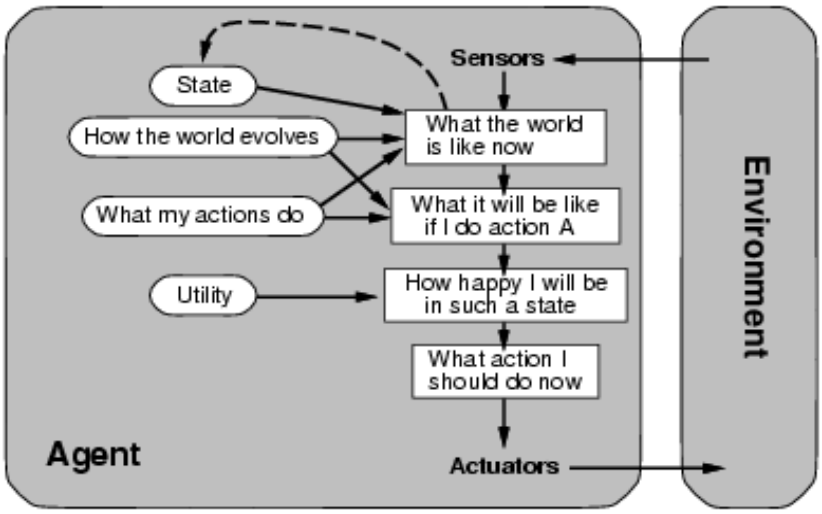
Goal-based agents



Utility-based agents

- Goals alone are not really enough to generate high-quality behavior in most environments.
- Utility based agents provides a more general agent framework.
- In case that the agent has multiple goals, this framework can accommodate different preferences for the different goals.
- Such systems are characterized by a **utility function** that maps a state or a sequence of states **onto a real number**
- The agent acts so as to maximize expected utility

Utility-based agents

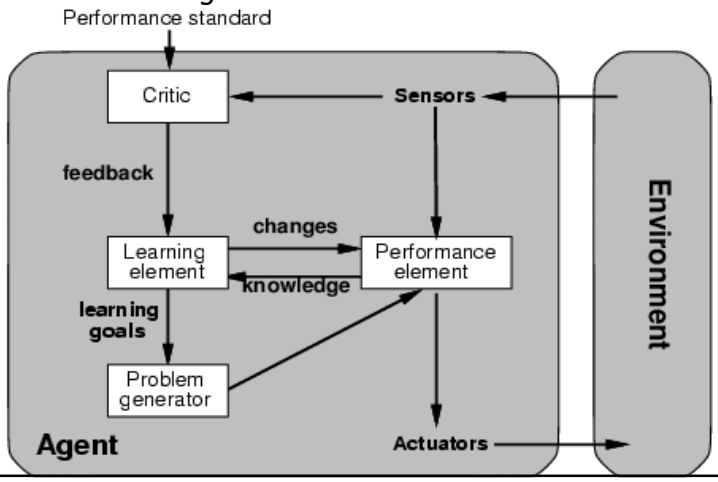


Learning agents

- Learning mechanisms can be constructed to improve every part of the agent.
- Learning allows an agent to operate in initially unknown environments.
- The learning element modifies the performance .
- Learning is required for true autonomy

Learning agents

The critic tells the learning element how well the agent is doing with respect to a fixed performance standard. The critic is necessary because the percepts themselves provide no indication of the agent's success.

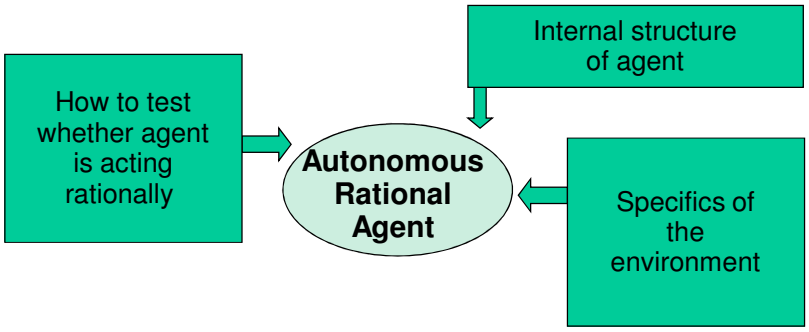


Conclusion

- AI deals with exciting but hard problems. A goal of AI is to build intelligent agents that act so as to optimize performance.
- An **agent perceives** and acts in an environment, has an **architecture,** and is implemented by an **agent program.**
- An **ideal agent** always chooses the action which **maximizes its expected performance**
- An **autonomous** agent uses its **own experience** rather than built-in knowledge of the environment by the designer.

Summary

- To design an agent: Think about these



- Reflex agents respond immediately to percepts.
- Goal-based agents act in order to achieve their goal(s).
- Utility-based agents maximize their own utility function.

The role of search in AI

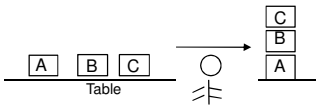
- There is a hypothesis: intelligent action can be reduced to search.
- While intelligence is certainly more than search, the hypothesis is attractive: search is well understood, easily mechanized, manageable, and so on.
- Formally represented knowledge can also be easily used in search.
- Search means systematic traversal of a space of possible solutions of a problem.

The role of search

- A search space is usually a graph (often simple as a tree).
  - A node represents a partial solution.
  - An edge represents a step in the construction of a solution.
- The purpose of search may be:
  - to find a path in the graph from a start node to a goal node (that is, from an initial to a final situation),
  - to find a goal node.

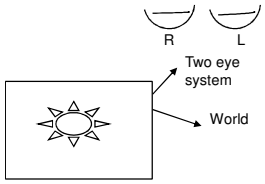
Search Ex1: Planning

- (a) which block to *pick*, (b) which to *stack*, (c) which to *unstack*, (d) whether to *stack* a block or (e) whether to *unstack* an already stacked block. These options have to be searched in order to arrive at the right sequence of actions.



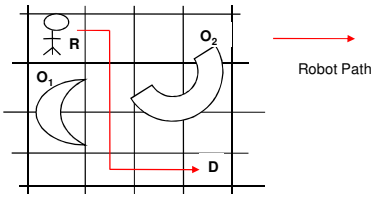
Search Ex2: Vision

- A search needs to be carried out to find which point in the image of *L* corresponds to which point in *R*. Naively carried out, this can become an $O(n^2)$ process where n is the number of points in the image.



Search Ex3: Robot Path Planning

- searching amongst the options of moving ***L***eft, ***R***ight, ***U***p or ***D***own. Additionally, each movement has an associated cost representing the relative difficulty of each movement. The search then will have to find the *optimal*, *i.e.*, the *least cost* path.



Search Ex4: Natural Language Processing

- search among many combinations of parts of speech on the way to deciphering the meaning. This applies to every level of processing- *syntax, semantics, pragmatics* and *discourse*.

The	man	would	like	to	play.
Noun		Verb	Preposition	Verb	

Search Ex5: Expert Systems

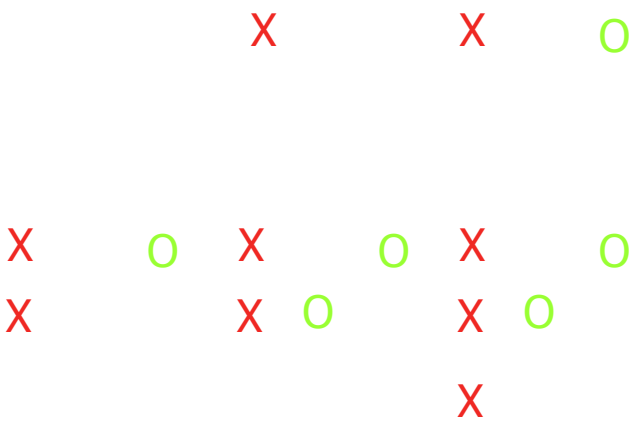
Search among rules, many of which can apply to a situation:

If-conditions
the infection is primary-bacteremia
AND the site of the culture is one of the sterile sites
AND the suspected portal of entry is the gastrointestinal tract
THEN
there is suggestive evidence (0.7) that infection is bacteroid

(from MYCIN)

Search Ex6: Game Playing

Path to goal isn't quite right.



Search Space Problems

- State Space : Graph of states (Express constraints and parameters of the problem)
- Operators : Transformations applied to the states.
- Start state : S_0 (Search starts from here)
- Goal state(s) : $\{G\}$ - Search terminates here.
- Cost : Effort involved in using an operator.
- Optimal path : Least cost path