Intelligent Agents

An agent perceives its environment through sensors and acts on this environment using effectors

percept - an agents perceptual input at any given instant
percept sequence - complete history of what the agent has perceived

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

- Human agent: eyes, ears, and other organs for sensors; hands, legs, mouth, and other body parts for actuators
- Robotic agent: cameras and infrared range finders for sensors; various motors for actuators

Rational Agent - do the right thing
strive for success

How does it do this?
Evaluation - how do we know it succeeded

Use a performance measure - domain dependent measure and should be objective

What is rational at any given time depends on 4 things

1) performance measure defines degree of success
2) percpet sequence - what agent has perceived so far
3) knowledge about environment
4) action agent performs

What is the difference between omniscience and rationality?
omniscience - agent knows the actual outcome of its actions.
in reality omniscience is impossible

rationality concerned with expected success (what is usual) given current perception. Not perfection. What is the percept sequence to date? Not what will happen in the future.

in terms of evaluation - careful not to penalize something it can't perceive or do some action it is incapable of.

ideal rational agent - for each percept sequence p in $\{p_0, p_1, p_2\}$ = set of all sequences, agent A chooses p_i such that performance measure is maximized.

mappings are the association between a percept sequence and an agent's action.

Mappings can be large and even infinite in some instances.

ideal mappings describe ideal agents. ideal mappings specify actions for any and every situation

agents should be autonomous

autonomous agents act based, not solely on knowledge but on knowledge and experience.

autonomous agents will adapt to their environments

agent architecture - makes percepts from an agent program available to the system

To design an agents program need to first know its task environment

PEAS - performance measure, environment, actuators, and sensors
What would be the PEAS description of a web bot that finds the lowest stereo price?

Properties of task environments:
Agent Environments

Fully observable vs. partially observable
fully observable - sensors have total access to the environment at each point in time. The sensors can detect all aspects of the environment relative to the problem. because of this agents do not need to keep track of the world

partially observable - Environment may be partially observable due to noisy or inaccurate sensors. Inability to sense a portion of the environment.

unobservable environment - agent has no senses.

Deterministic vs. Stochastic - The next state of the environment can be determined from current state and actions of agent. (deterministic) This is evaluated from the point of view of the agent. otherwise the environment is stochastic. stochastic environments have some randomness associated with them. Agents in a stochastic environment use probabilistic plans.

uncertain environments are not fully observable or not deterministic.

nondeterministic environment - actions are characterized by possible outcomes, with no probabilities attached to them.

Episodic vs. Sequential -
episodic - agents experience is divided into episodes. Each episode doesn't affect the actions of the agent in another episode.
sequential - current decisions could affect future decisions

Static vs. Dynamic -
static environment doesn't change. Easier environment for agent. Does not have to be constantly checking its environment
dynamic environment - continuously asks the agent what it wants to do.
semidynamic - the environment doesn't change with the passage of time but the agent's performance score does.

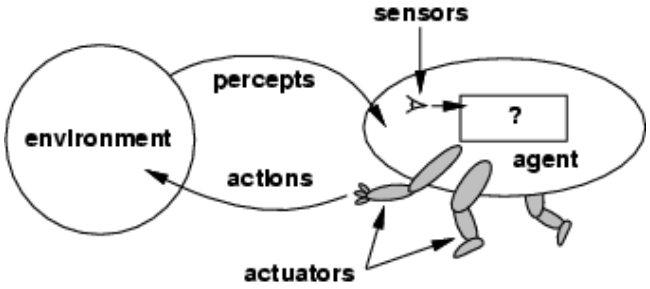
Discrete vs. Continuous - applies to the state of the environment, the way time is handled, and to the percepts and actions of the agent. discrete - limited # of distinct clearly definable percepts and actions such as a lookup environment. Ex. chess game
continuous - driving environment. environment changes as you move down the road

Single Agent vs. Multiagent - one agent vs. many. solving crossword puzzle - single agent chess two agents.

competitive multiagent environments - when the agent A acts to maximize it's performance measure while minimizing agent B's performance measure.

cooperative multiagent environment - agents work together to maximize all agent's performance measure. ex. soccer playing agents. Also partially competitive. Why?

known vs. unknown
refers to the agents knowledge about the physics of the environment.
the agent knows how the environment works.



The agent function maps from percept histories to actions:

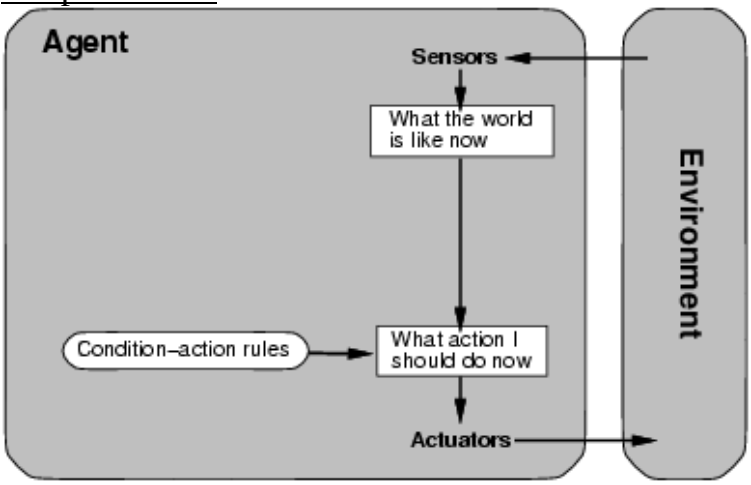
$$[f: P^* \rightarrow A]$$

The agent program runs on the physical architecture to produce *f*
agent = architecture + program

Simplest agent - lookup table, not suitable for all domains.
#of percept sequences can be large for some domains. ex. chess 35^{100}
downside, for chess, long time to build table, the agent is not autonomous

4 types of agents programs

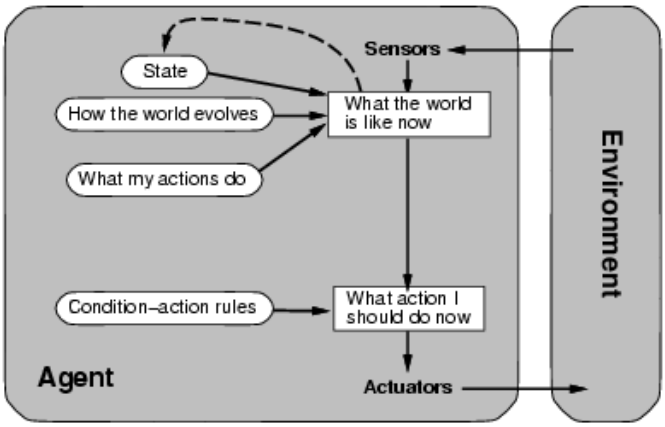
Simple Reflex



agent - acts on a set of condition - action rules

if x happens do y
(lookup table) (rule)

Model based agents



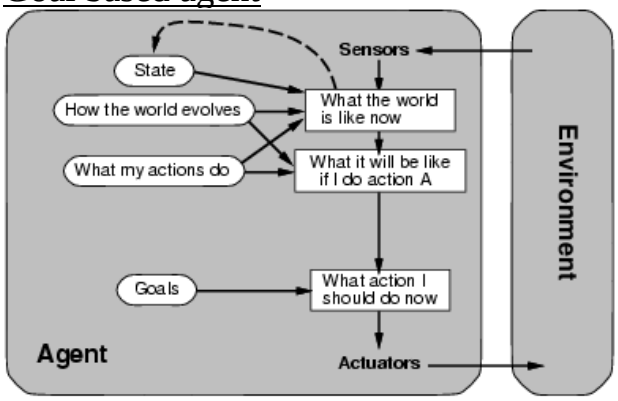
Agents that keep track of the world
These agents incorporate simple reflex agents and add the ability to evaluate the current state of the world in terms of how things evolve.

Multiple states may look the same but depending on the state of the world different actions are appropriate.

Ex. baseball agent - ball is on the floor in front of the pitcher. Short pop? Pitcher dropped the ball?

State is evaluated in terms of how it changed from the previous state.

Goal based agent

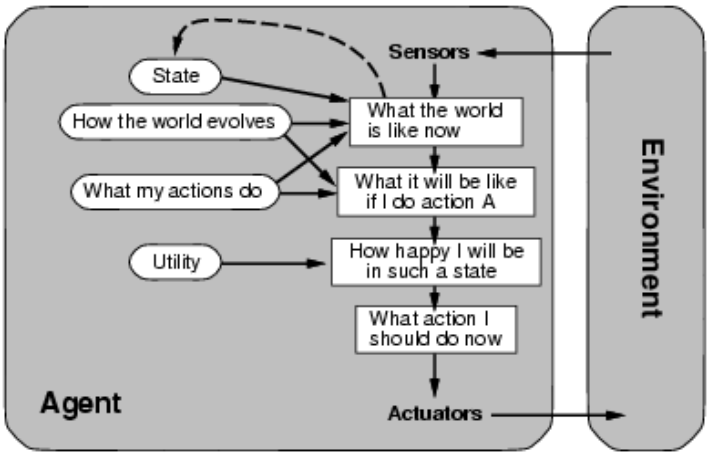


Agents have all of the above and goal.

Goal is a desirable situation. Involves consideration of the future. Goals provide a direction between happy and unhappy states

Goal based agents are more flexible. change in environment causes knowledge base to be updated causing actions to be chosen accordingly. Reflex agents have lots of rules that need to be changed.

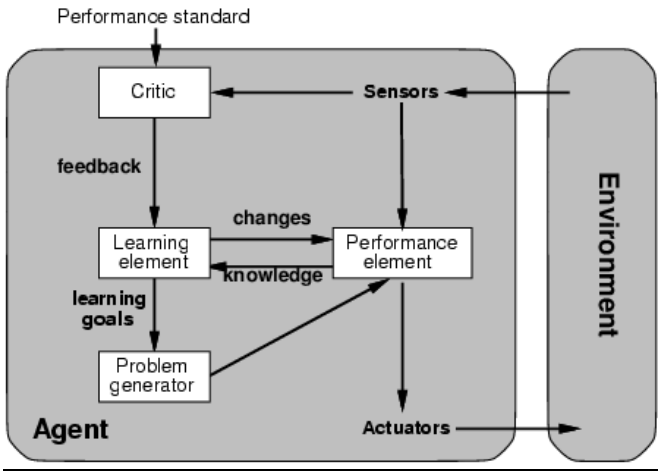
Utility Based Agents



uses a utility function to decide which world state is better for an agent. The utility function maps a state to a number which is a measure of happiness. Goal is the ultimate happy state (true happiness) ecstatic!!

We can have varying degrees of happy along the way.
if one world state is preferred over another, it has a higher utility for the agent.

Learning Agents



4 components:

learning element - responsible for making improvements.

performance element - what we have up to now considered to be the entire agent. Takes percepts and then decides on actions.

critic - gives the learning element feedback on how the performance element is doing and if it needs to be modified.

problem generator - it suggests actions that lead to new and informative experiences.

short term, if an agent always tried to maximize the performance measure it may not find good long term solutions to a problem. The problem generator allows the agent to explore and experience some sub optimal performance in the hopes for better long term performance.

HW.

Chapter 2 - 2.3, 2.4, 2.5, 2.9