CS 188

Fall 2015

8/27

Rationality is defined in terms of achieving maximum utility by some pre-defined metric or set of goals/intentions. Rationality depends only on the usefulness of the choice reached rather than on any aspect of the process that led to that choice. For example, a rational process for playing tic-tac-toe could be formed simply by creating a table for all game states; this would not have consist of any decision process whatsoever.

AI, economics, statistics, operations research, etc. assume utility to be **exogenously specified**. The difficulty of competent machines is a question of value misalignment.

9/1

Sensors are preceptors of the environment and *actuators* are methods by which it manipulates the environment through actions.

An *agent function* maps from percept histories to actions.

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

An *agent program* I runs on some machine M to implement f:

$$f = Agent(I, M)$$

Not all agent functions can be implemented by some agent program. E.g. halting problems, NP-hard problems, chess (combinatorically large amounts of information needed to process)

Use a *performance measure* to evaluate effectiveness. Care must be taken to ensure that the performance measure accurately measures the execution of the task designed. A performance measure is a measure on the *environment*.

A *rational agent* maximizes the expected value of the performance measure. Rationality depends on prior knowledge of environment, action, previous percepts.

PEAS model: performance measure, environment, actuators, sensors.

Environment characteristics: observable (fully/partial), number of agents, deterministic/stochastic, static/dynamic, discrete/continuous, known/unknown (e.g. don't know the dynamics of the system, but still try to maximize utility)

Effects of the environment on agent design:

- partially observable → agent requires memory
- multi-agent → randomness may be necessary
- static → can utilize time to implement a rational decision
- continuous time → will have some continuously operating controller

Agent types (increasing generality/complexity):

- simple reflex
- state-based reflex agents
- goal-based agents
- utility-based agents

Two of these are reflexive, the next two are planning-based.