CMSC 471
Fall 2012

Class #2

Tue 9/4/11
Agents/Lisp

Kevin Winner, winnerk1@umbc.edu

Today's class

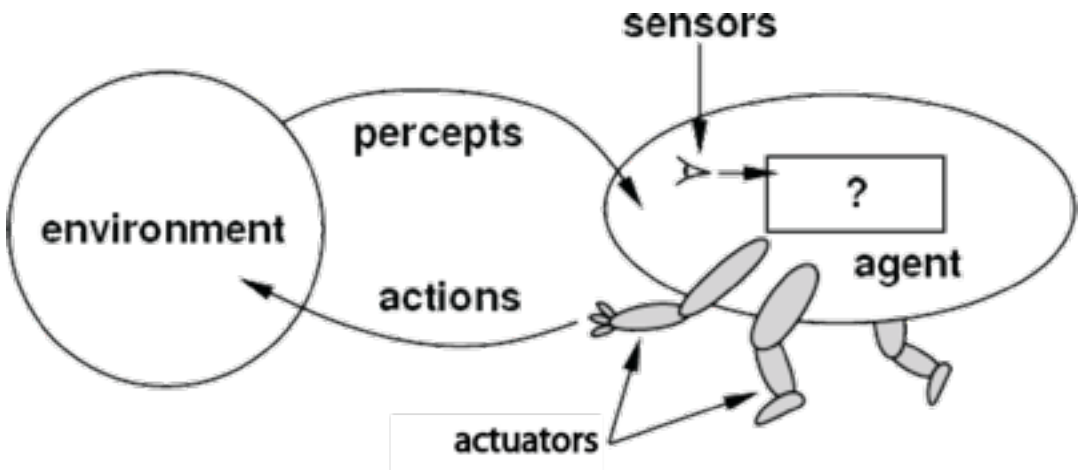
- What's an agent?
  - Definition of an agent
  - Rationality and autonomy
  - Types of agents
  - Properties of environments
- Lisp – demonstration

Intelligent Agents

Chapter 2

How do you design an intelligent agent?

- Definition: An **agent** is anything that perceives its environment through **sensors** and acts upon that environment through **actuators**.
- An **agent program** receives a series of **percepts**, and maps this percept sequence to a sequence of **actions**.



What makes an agent intelligent?

- Rationality
- Autonomy

Rationality

- An ideal **rational agent** should, for each possible percept sequence, do whatever actions will maximize its expected performance based on
  - (1) the percept sequence
  - (2) the available actions, and
  - (3) a given or learned performance measure
- Rationality → Need a performance measure to say how well a task has been achieved.
  - How do you define a performance measure?
- Rationality often includes information gathering, not "rational ignorance." (If you don't know something, find out!)
  - Especially when learning a performance measure
- Rational != omniscient/perfect

Autonomy

- A system is autonomous to the extent that its own behavior is determined by its own experience.
- Therefore, a system is not autonomous if it is guided by its designer according to a priori decisions.
- An intelligent agent does need at least
  - Enough built-in knowledge to survive.
  - The ability to learn.

Some agent program types

- **(0) Table-driven agents**
  - use a percept sequence/action table in memory to find the next action. They are implemented by a (large) **lookup table**.
- **(1) Simple reflex agents**
  - are based on **condition-action rules**, implemented with an appropriate production system. They are stateless devices which do not have memory of past world states.
- **(2) Model-based reflex agents**
  - maintain a model of the parts of the world which are presently unobservable
- **(3) Agents with goals**
  - are agents that, in addition to state information, have **goal information** that describes desirable situations. Agents of this kind take future events into consideration.
- **(4) Utility-based agents**
  - base their decisions on **classic axiomatic utility theory** in order to act rationally.

(0) Table-driven agents

- **Table lookup** of percept-action pairs mapping from every possible perceived state to the optimal action for that state
- **Problems**
  - Wildly unscalable
    - Data requirement (A chess agent would have 10^{150} entries)
    - Too big to construct
    - Too big to learn
  - Can't react and learn about the environment
- There are still uses for table-driven agents, however!

(1) Simple reflex agents

- **Rule-based reasoning** to map from current percept to optimal action; each rule handles a collection of possible perceived states
- **Problems**
  - Useful reflex agents are still usually too big to generate and to store
  - Still not adaptive to changes in the environment; requires collection of rules to be updated if changes occur
  - Still can't make actions conditional on previous state

(2) Model-based reflex agents

- Encode a **model** or “internal state” of the world to remember the past as contained in earlier percepts.
- Needed because sensors do not usually give the entire state of the world at each input, so perception of the environment is captured over time. “State” is used to encode different "world states" that generate the same immediate percept.

(3) Goal-based agents

- Choose actions so as to achieve a (given or computed) goal.
- A goal is a description of a desirable situation.
- **Deliberative** instead of **reactive**.
- May have to consider long sequences of possible actions before deciding if goal is achieved – involves consideration of the future, *"what will happen if I do...?"*
- **Search** and **Planning** are fields of AI virtually dedicated to goal-based agents

(4) Utility-based agents

- When there are multiple possible alternatives, how to decide which one is best?
- A goal specifies a crude distinction between a happy and unhappy state, but often need a more general performance measure that describes "degree of happiness."
- Utility function **U: State → Reals** indicating a measure of success or happiness when at a given state.
- Allows decisions comparing choice between conflicting goals, and choice between likelihood of success and importance of goal (if achievement is uncertain).
- **Reinforcement learning** uses exclusively utility-based agents

Properties of Environments

- Fully observable/partially observable
- Deterministic/stochastic
- Episodic/sequential
- Static/dynamic
- Discrete/continuous
- Single agent/multiagent
- Known/Unknown

Properties of Environments

- **Fully observable/Partially observable.**
  - If an agent's sensors give it access to the complete state of the environment needed to choose an action, the environment is **fully observable**.
  - Such environments are convenient, since the agent is freed from the task of keeping track of the changes in the environment.
- **Deterministic/Stochastic**.
  - An environment is **deterministic** if the next state of the environment is completely determined by the current state of the environment and the action of the agent; in a **stochastic** environment, there are multiple, unpredictable outcomes
  - In a fully observable, deterministic environment, the agent need not deal with uncertainty.

Properties of Environments

- **Episodic/Sequential**.
  - An **episodic** environment means that subsequent episodes do not depend on what actions occurred in previous episodes.
  - In a **sequential** environment, the agent engages in a series of connected episodes.
  - Such environments do not require the agent to plan ahead.
- **Static/Dynamic.**
  - A **static** environment does not change while the agent is thinking.
  - The passage of time as an agent deliberates is irrelevant.
  - The agent doesn't need to observe the world during deliberation.

Properties of Environments

- **Discrete/Continuous.**
  - If the number of distinct percepts and actions is limited, the environment is **discrete**, otherwise it is **continuous**.
- **Single agent/Multi-agent.**
  - If the environment contains other intelligent agents, the agent needs to be concerned about strategic, game-theoretic aspects of the environment (for either cooperative *or* competitive agents)
  - Multi-agent systems change the problem so significantly that most algorithms no longer work or need heavy modification
- **Known/Unknown**
  - In a **known** environment, the agent understands fully the laws which govern the environment

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire							
Back-gammon							
Google car							
Assembly line robot							
Medical diagnosis							

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire	**No**	**Yes**	**No**	**Yes**	**Yes**	**Yes**	**Yes**
Back-gammon							
Google car							
Assembly line robot							
Medical diagnosis							

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire	No	Yes	No	Yes	Yes	Yes	Yes
Back-gammon	**Yes**	**No**	**No**	**Yes**	**Yes**	**No**	**Yes**
Google car							
Assembly line robot							
Medical diagnosis							

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire	No	Yes	No	Yes	Yes	Yes	Yes
Back-gammon	Yes	No	No	Yes	Yes	No	Yes
Google car	**No**	**No**	**No**	**No**	**No**	**No**	**No**
Assembly line robot							
Medical diagnosis							

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire	No	Yes	No	Yes	Yes	Yes	Yes
Back-gammon	Yes	No	No	Yes	Yes	No	Yes
Google car	No	No	No	No	No	No	No
Assembly line robot	**No**	**No**	**Yes**	**No**	**Yes***	**Yes**	**No***
Medical diagnosis							

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire	No	Yes	No	Yes	Yes	Yes	Yes
Back-gammon	Yes	No	No	Yes	Yes	No	Yes
Google car	No	No	No	No	No	No	No
Assembly line robot	No	No	Yes	No	Yes*	Yes	No*
Medical diagnosis	**No**	**No**	**No**	**No**	**No**	**Yes**	**No**

Characteristics of environments

	Fully observable	Deterministic	Episodic	Static	Discrete	Single agent	Known
Solitaire	No	Yes	No	Yes	Yes	Yes	Yes
Back-gammon	Yes	No	No	Yes	Yes	No	Yes
Google car	No	No	No	No	No	No	No
Assembly line robot	No	No	Yes	No	Yes*	Yes	No*
Medical diagnosis	**No**	**No**	**No**	**No**	**No**	**Yes**	**No**

→ Lots of real-world domains fall into the hardest case!

Lisp Revisited

Once more unto the breach