Intelligent Agents

Chapter 2

(Adapted from Stuart Russel, Dan Klein, and others. Thanks guys!)

Outline

- Agents and environments
- Rationality
- PEAS (Performance measure, Environment, Actuators, Sensors)
- Environment types
- Agent types

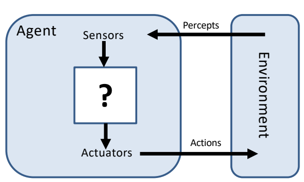
Agents and Environments

- Agents include:
  - Humans
  - Robots
  - Softbots
  - Thermostats
  - More…
- The agent function represents the "intelligence"
  - Map from percept histories to actions:

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

- An agent program running on physical architecture implements the agent function

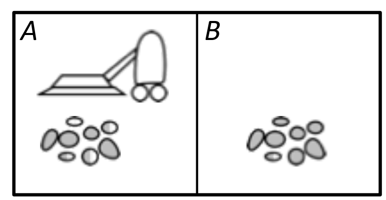
The line between agent and environment depends on the level of abstraction.



Environment considered as a black box, completely external to the agent

- *even if it's simulated by local code.*
- *Agent has accept to world **only** via percepts.*

Vacuum-cleaner world



Percepts: location and contents, e.g., [*A, Dirty*]

Actions: *Left*, *Right*, *Suck*, *NoOp*

So: super simple world!
- 1-D environment, just two locations
- Only four possible actions, uniformly available in all locations

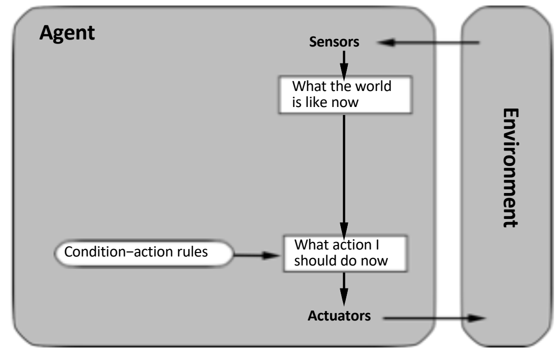
A (reflex) vacuum-cleaner agent

```
function Reflex-Vacuum-Agent( [location,status]) returns an action
    if status = Dirty then return Suck  else if
    location = A then return Right  else if
    location = B then return Left
```

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*
.	.
.	.
.	.

- What is the right function?

A first example: **Simple reflex agents**



- Focus on now. No state, no history. Just reacts. True Zen machine!
- Does this *ever* make sense as a design?

Reflex Agents = Table-lookup?

- Could express as table instead of function.
  - Complete map from percept (histories) to actions
  - Actions "computed" by simply looking up appropriate action in table

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*
..	..

- Drawbacks:
  - Huge table!
  - Rigid, no autonomy, flexibility
  - Even with learning, need a long time to "learn" all entries in complex world.
- Better agent programs: produce complex behaviors from compact specifications (programs)

Rationality

Fixed performance measure evaluates the environment sequence
– one point per square cleaned up in time T ?
– one point per clean square per time step, minus one per move?
– penalize for $> k$ dirty squares?
– More?

A rational agent chooses whichever action maximizes the expected value of the performance measure given current knowledge
- Knowledge = initial knowledge + the percept sequence to date

Rational $\neq$ omniscient
- percepts may not supply all relevant information

Rational $\neq$ clairvoyant about action efficacy
- action outcomes may not be as expected

Hence, rational $\neq$ *guaranteed* successful

Rationality motivates $\Rightarrow$ exploration, learning, autonomy

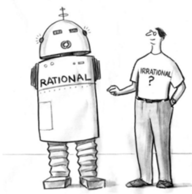
Rationality and Goals

- "to maximize expected outcome". What does that mean?
  - Rationality is inherently based on having some *goal* that we want to achieve
  - Performance measure: expresses extend of satisfaction, progress towards
- Suppose: We have a game:
  - Flip a biased coin (probability of heads is h…not neces
  - Tails = loose $1; Heads= win $1
- What is the expected winnings in a series of flips?
  - (1)h + (-1)(1-h) = 2h-1

- Rational to play? Depends…
  - What if performance measure is total money?
  - What if performance measure is spending rate?
  - Why might a human play this game at expected loss?
    - Vegas, baby!

Summary: Rationality

- Remember: rationality is ultimately defined by:
  - Performance measure
  - Agent's prior (initial) knowledge of world
  - Agent's percepts to date (updates to world)
  - Available actions



- Some thought questions:
  - Is it rational to inspect the street before crossing?
  - Is it rational to try new things?
  - Is it rational to update beliefs?
  - Is it rational to construct conditional plans of action in advance?
- Could now go into:
  - empirical risk minimization (statistical classification)
  - Expected return maximization (reinforcement learning)
- Wait till later! Let's get clearer concept of agents first!

PEAS: Specifying Task Environments

- To design a rational agent, we must specify the task environment
  - We've done this informally so far…vague
  - The characteristics of the task environment determine much about agents!
  - Need to formalize…
- PEAS: Dimensions for specifying task environments
  - Performance measure: metrics to measure performance
  - Environment: Descr. of areas/context agent operates in
  - Actuators: Ways that agent can intervene/act in the world
  - Sensors: Information channels through which agent gets info about world
- Consider, e.g., the task of designing an automated taxi:
  - Performance measure??
  - Environment??
  - Actuators??
  - Sensors??

PEAS: Specifying Task Environments

- To design a rational agent, we must specify the task environment
  - We've done this informally so far…vague
  - The characteristics of the task environment determine much about agents!
  - Need to formalize…
- PEAS: Dimensions for specifying task environments
  - Performance measure: metrics to measure performance
  - Environment: Descr. of areas/context agent operates in
  - Actuators: Ways that agent can intervene/act in the world
  - Sensors: Information channels through which agent gets info about world
- Consider, e.g., the task of designing an automated taxi:
  - Performance measure?? safety, destination, profits, legality, comfort…
  - Environment?? US streets/freeways, traffic, pedestrians,weather…
  - Actuators?? steering, accelerator, brake, horn, speaker/display…
  - Sensors?? video, accelerometers, gauges, engine sensors,keyboard, GPS…

PEAS: Internet shopping agent

- Performance measure??
- Environment??
- Actuators??
- Sensors??

PEAS: Spam filtering agent

- Performance measure??
- Environment??
- Actuators??
- Sensors??

Environments: A more concise framework

- PEAS gave us a framework for outlining key agent features
  - One of those was environment…but we just had a general description
  - Much more useful to think about the *kind* of environment it represents
  - Need a concise, formal framework classifying kinds of environments!
  - Based on six dimensions of difference:

1. Observability: Full vs. Partial
   1. Fully: An agent's sensors give it access to the complete state of the environment at each point in time.
   2. Partially observable: An agent's sensors give it access to only some partial slice of the environment at each point in time.
2. Determinism: Deterministic vs. stochastic
   1. Deterministic: The next state of the environment is completely determined by the current state and the action executed by the agent.
   2. Stochastic: State and actions are known/succeed based on some statistical model. Knowledge is fallible, as are action outcomes.
3. Contiguity: Episodic vs. sequential
   1. Episodic: The agent's experience is divided into independent atomic "episodes"; each episode consists of the agent perceiving and then performing a single action
   2. Sequential: The agent's experience is a growing series of states; new action is based not only on actual state, but on state/action in previous episodes.

Environments: A more concise framework

4. Stability: Static vs. Dynamics
   1. Static: Environment is unchanging while the agent is deliberating
   2. Dynamic: Environment is fluid, keeps evolving while agent plans action
5. Continuity: Discrete vs. Continuous
   1. Discrete: A limited number of distinct, pre-defined percepts and actions possible.
   2. Continuous: An unlimited number of actions are possible, infinite percepts readings possible.
6. Actors: Single vs. multi-agent
   1. Single: Agent is operating solo in environment. Sole agent of change
   2. Multi-agent: There are other agents/actors to consider, take into account, coordinate with…compete against.

- What is the real world like?
  - Depends on how you frame the world
  - What your "world" is. How much detail of it you represent.

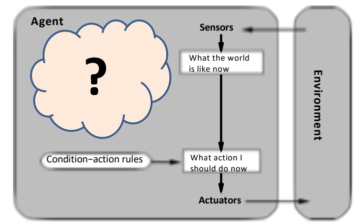
Thinking about Environment types

	Solitaire	Backgammon	Internet shopping	Taxi
Observable??				
Deterministic??				
Episodic??				
Static??				
Discrete??				
Single-agent??				

17

Characterizing capabilities: **Agent Types**

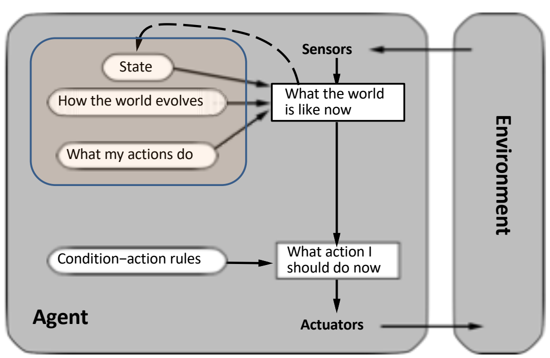
The bare basics: The simple **Reflex Agent** we examined before…



- Reflex Agent: No state, no history. Just reacts. Table lookup…
- Adding functionality leads to new (more flexible) agent types:
  - Reflex agents with state
  - Goal-based agents
  - Utility-based agents
- All can be turned into *learning agents*
  - Focus on *dynamically improving* the components agent contains

18

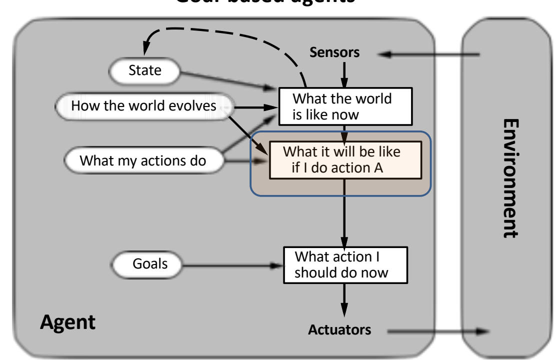
Reflex agents with state



- Add internal model of world:
  - Current state not just "current sensor read". Percept *history*
  - Models aspects beyond sensors: world model could deduce added info
  - Action is <u>still</u> just table lookup: based on configurations of world state

19

Goal-based agents

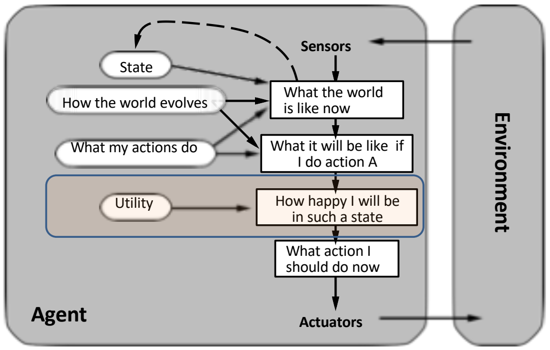


Not just reacting, but trying to change state *towards some goal:*

1. Get percepts, add to state
2. Allow world model to deduce new knowledge…comes to quiescence
3. Use a *planning module* to reason about possible future states
4. Choose action to lead to desired future *goal states*

20

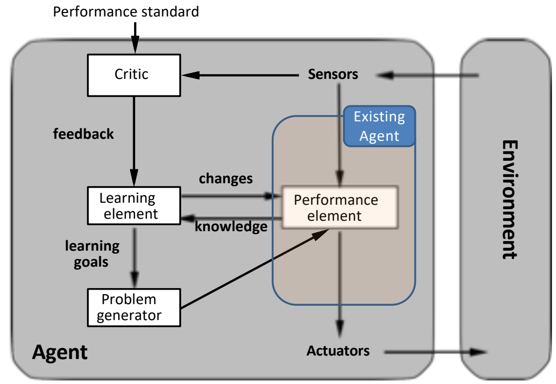
Utility-based agents



- Goal states alone are too simplistic:
- Some goal states are “more satisfying” than others.
  - Goal state is not unique/defined/attainable
  - “Happiness” often more continuous function based on many factors
  - “goal” = get to strongest possible state
  - Action is uncertain: get to strongest expected state…based on probability

21

Learning: Any agent may be self-improving



- Learning ability is orthogonal to agent type: can be applied to any agent!
  - Modules above added on top of any basic agent description
  - Essentially rewrites/improves any element of existing agent dynamically

22

Summary

- Agents interact with environments through actuators and sensors
- PEAS descriptions outline task environment and agent’s access to it
- The agent function describes what the agent does in all circumstances

  f: (initial state + P*) → A
- For non-reflex agents: Some sort of performance measure
  - evaluates the current (P* → current state)
  - Boolean goal function vs. Utility function
- A perfectly rational agent *maximizes expected performance*
- Agent programs implement (some) agent functions
- Environments are categorized along several dimensions:
  - observable? deterministic? episodic? static? discrete? single-agent?
- Several basic agent architectures exist:
  - reflex, reflex with state, goal-based, utility-based
  - Learning can be added to any agent type

23