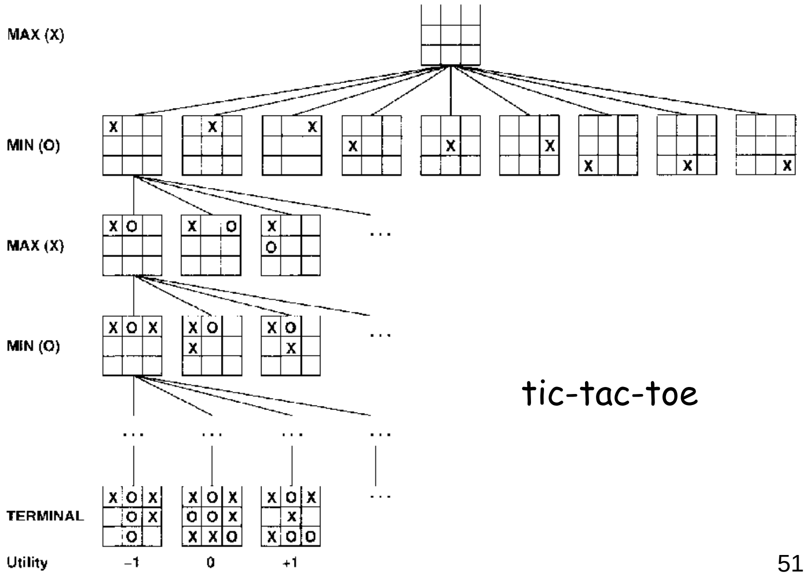
Course Overview (cont.)

Practical applications of search.

- **06-Game playing.** [AIMA Ch 5] The minimax algorithm. Resource limitations. Aplha-beta pruning. Elements of chance and non-deterministic games.

- **06-Advanced Game Playing** [AIMA Ch 16]
  - Reinforcement Q-Learning



tic-tac-toe

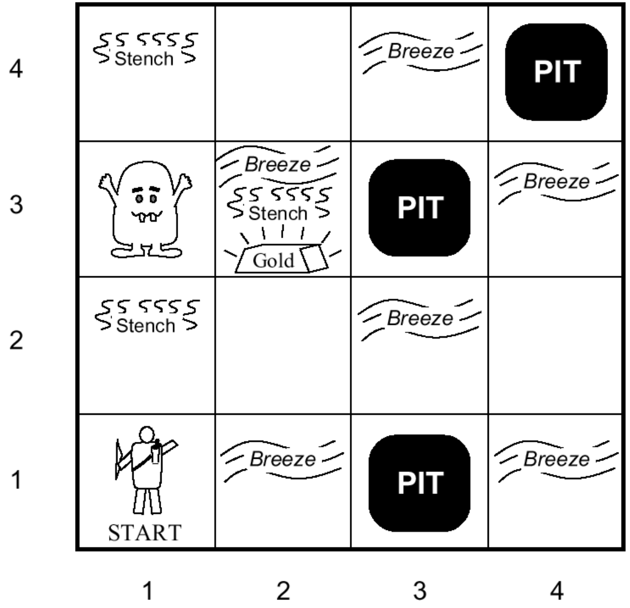
51

Course Overview (cont.)

Towards intelligent agents

- **9-Agents that reason logically 1.** [AIMA Ch 7] Knowledge-based agents. Logic and representation. Propositional (boolean) logic.

- **10-Agents that reason logically 2.** [AIMA Ch 7] Inference in propositional logic. Syntax. Semantics. Examples.



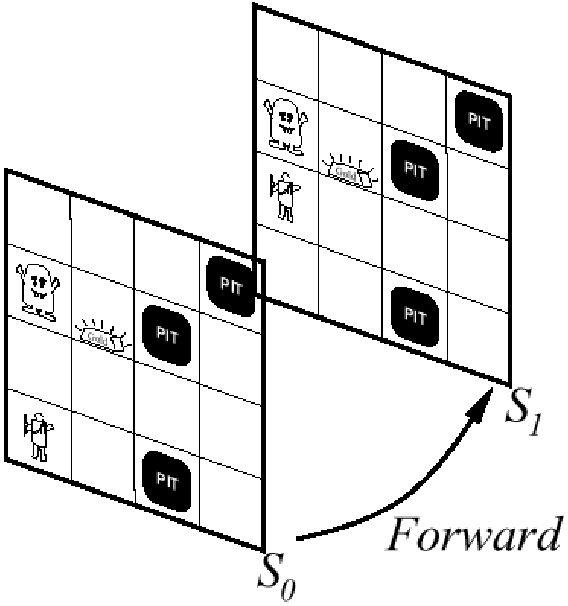
wumpus world

Course Overview (cont.)

Building knowledge-based agents: 1st Order Logic

- **11-First-order logic 1.** [AIMA Ch 8] Syntax. Semantics. Atomic sentences. Complex sentences. Quantifiers. Examples. FOL knowledge base. Situation calculus.

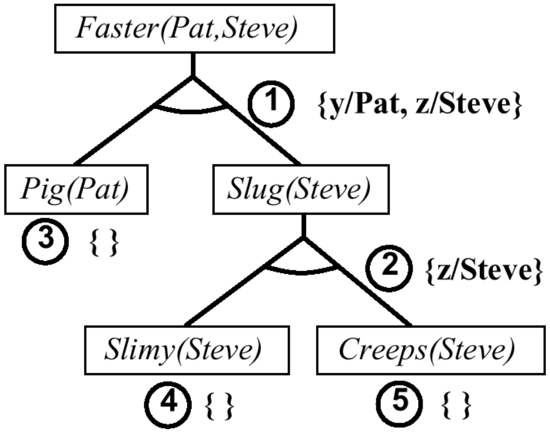
- **12-First-order logic 2.**
  [AIMA Ch 8] Describing actions.
  Planning. Action sequences.



Course Overview (cont.)

Reasoning Logically

- **13/14-Inference in first-order logic.** [AIMA Ch 9] Proofs. Unification. Generalized modus ponens. Forward and backward chaining.

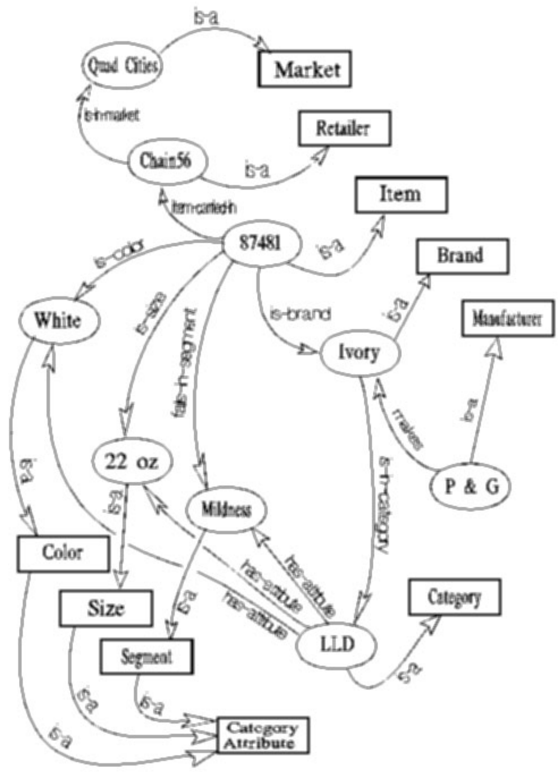


Example of backward chaining

Course Overview (cont.)

Examples of Logical Reasoning Systems

- **15-Logical reasoning systems.** [AIMA Ch 9] Indexing, retrieval and unification. The Prolog language. Theorem provers. Frame systems and semantic networks.

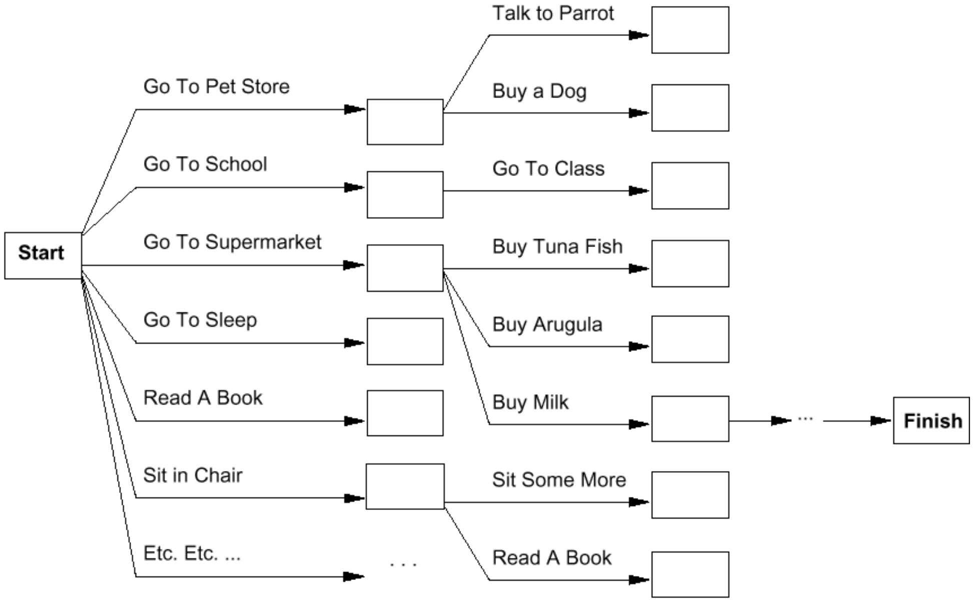


Semantic network used in an insight generator (Duke university)

Course Overview (cont.)

Systems that can Plan Future Behavior

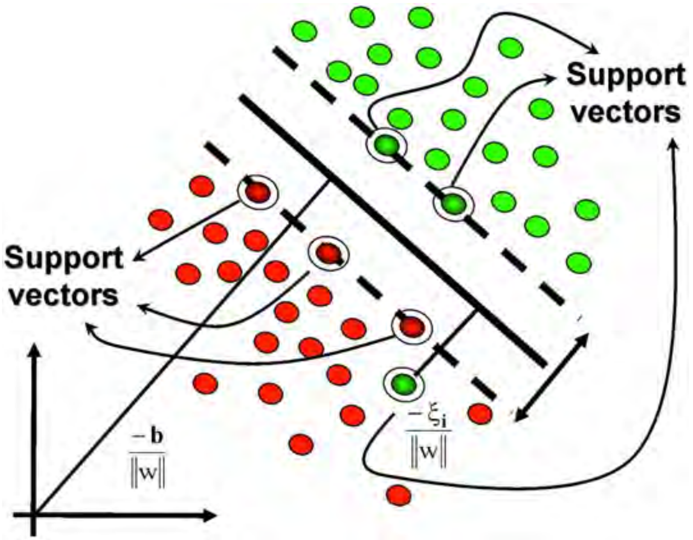
- **16-Planning.** [AIMA Ch 10] Definition and goals. Basic representations for planning. Situation space and plan space. Examples.



Course Overview (cont.)

Handling fuzziness, change, uncertainty.

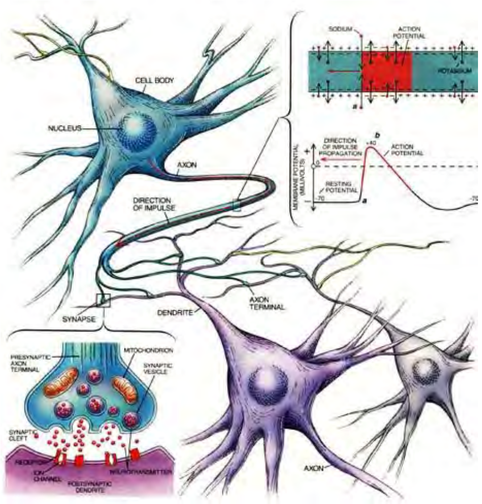
18-Learning from examples. [AIMA 18 + handout]. Supervised learning, learning decision trees, support vector machines.

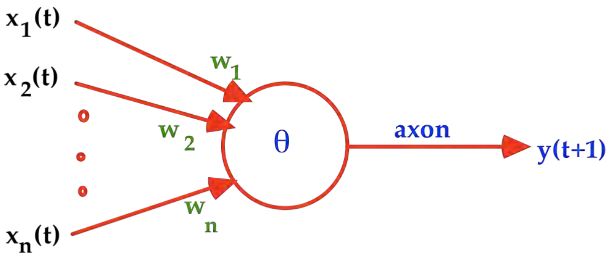


Course Overview (cont.)

Learning with Neural networks

- **19/20-Learning with Neural Networks.** [Handout + AIMA 18] Introduction to perceptrons, Hopfield networks, self-organizing feature maps. How to size a network? What can neural networks achieve? Advanced concepts – convnets, deep learning, stochastic gradient descent, dropout learning, autoencoders, applications and state of the art.





Course Overview (cont.)

Handling uncertainties, fuzziness, and changes

- **21/22-Probabilistic Reasoning.** [AIMA Ch 13, 14, 15]
  Reasoning under uncertainty – probabilities, conditional independence, Markov blanket, Bayes nets. Probabilistic reasoning in time. Hidden Markov Models, Kalman filters, dynamic Bayesian networks.

For assessing diagnostic probability from causal probability:

$$P(Cause|Effect) = \frac{P(Effect|Cause)P(Cause)}{P(Effect)}$$

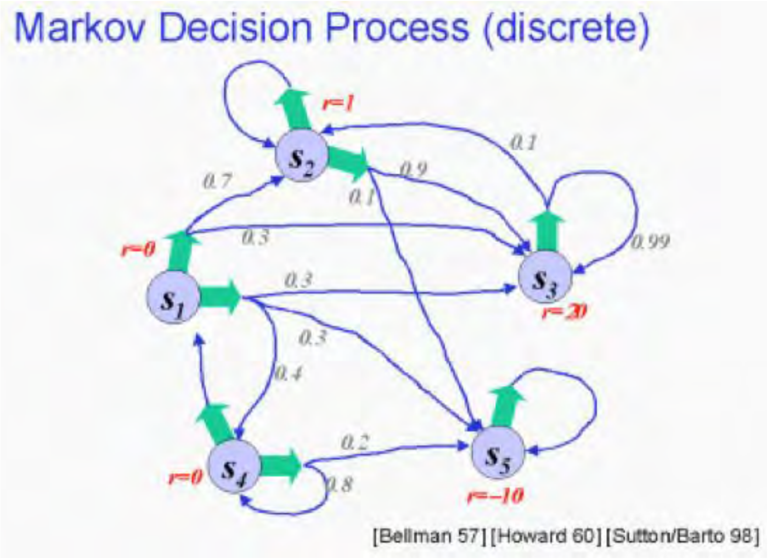
E.g., let M be meningitis, S be stiff neck:

$$P(M|S) = \frac{P(S|M)P(M)}{P(S)} = \frac{0.8 \times 0.0001}{0.1} = 0.0008$$

Course Overview (cont.)

Handling uncertainties, fuzziness, and changes

- **23-Probabilistic decision making.** [AIMA 16, 17] – utility theory, decision networks, value iteration, policy iteration, Markov decision processes (MDP), partially-observable MDP (POMDP).

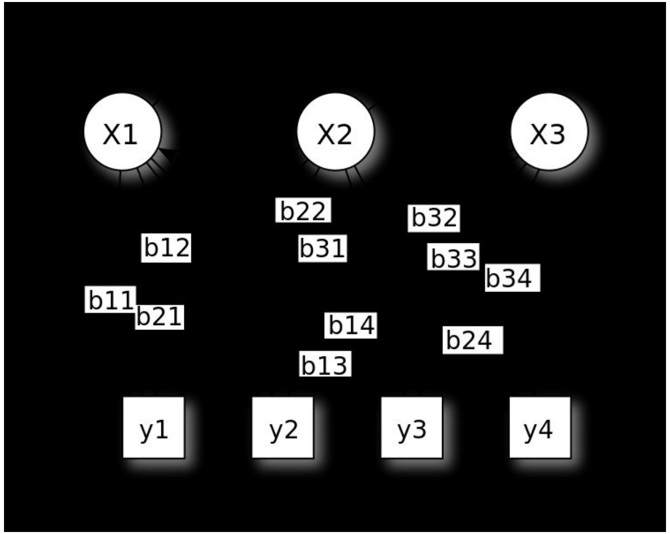


Course Overview (cont.)

Handling uncertainties, fuzziness, and changes

- **24-Probabilistic reasoning over time.** [AIMA 15]

Temporal models, Hidden Markov Models, Kalman filters, Dynamic Bayesian Networks, Automata theory.

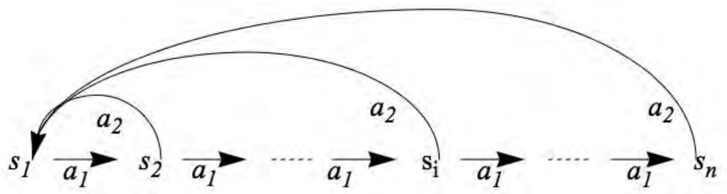


Course Overview (cont.)

Handling uncertainties, fuzziness, and changes

- **25-Probability-based learning.** [AIMA 20-21]

Probabilistic Models, Naïve Bayes Models, EM algorithm, Reinforcement Learning.

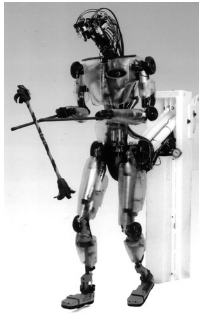


$$Q_{k+1}(s,a) \leftarrow \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma \max_{a'} Q_k(s',a') \right]$$

Course Overview (cont.)

What challenges remain?

- **26-Towards intelligent machines.** [AIMA Ch 26, 27] The challenge of robots: with what we have learned, what hard problems remain to be solved? Different types of robots. Tasks that robots are for. Parts of robots. Architectures. Configuration spaces. Navigation and motion planning. Towards highly-capable robots. What have we learned. Where do we go from here?



More from the Discussion Session later today

INTELLIGENT AGENTS

Defining Intelligent Agents

- Intelligent Agents (IA)
- Environment types
- IA Behavior
- IA Structure
- IA Types

What is an (Intelligent) Agent?

- An over-used, over-loaded, and misused term.

- Anything that can be *viewed as* **perceiving** its **environment** through **sensors** and **acting** upon that environment through its **effectors** to maximize progress towards its **goals**.

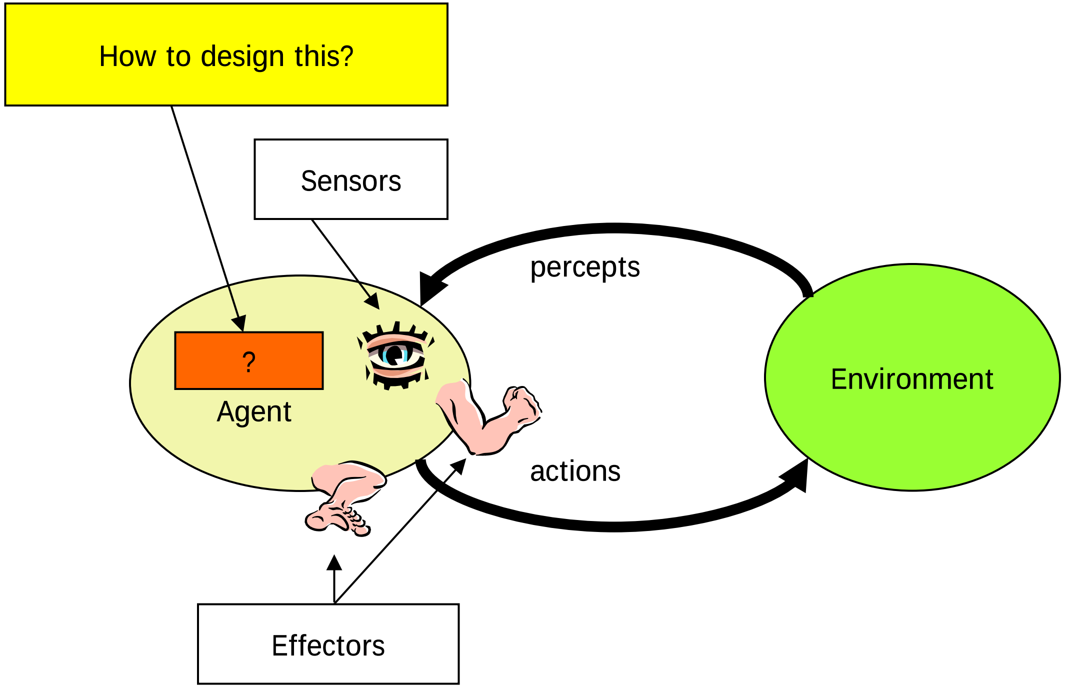
What is an (Intelligent) Agent?

- **PAGE** (Percepts, Actions, Goals, Environment)

- Task-specific & specialized: well-defined goals and environment

- The notion of an agent is meant to be a tool for analyzing systems, It is not a different hardware or new programming languages

Agent and Environment Interactions



A Windshield Wiper Agent (Example)

How do we design a agent that can wipe the windshields when needed?

- Goals?
- Percepts?
- Sensors?
- Effectors?
- Actions?
- Environment?

A Windshield Wiper Agent (Cont’d)

- Goals: Keep windshields clean & maintain visibility
- Percepts: Raining, Dirty
- Sensors: Camera (moist sensor)
- Effectors: Wipers (left, right, back)
- Actions: Off, Slow, Medium, Fast
- Environment: Inner city, freeways, highways, weather ...

Interactions Among Agents

Collision Avoidance Agent (CAA)

- Goals: Avoid running into obstacles
- Percepts ?
- Sensors?
- Effectors ?
- Actions ?
- Environment: Freeway

Lane Keeping Agent (LKA)

- Goals: Stay in current lane
- Percepts ?
- Sensors?
- Effectors ?
- Actions ?
- Environment: Freeway

Interactions Among Agents

Collision Avoidance Agent (CAA)

- Goals: Avoid running into obstacles
- Percepts: Obstacle distance, velocity, trajectory
- Sensors: Vision, proximity sensing
- Effectors: Steering Wheel, Accelerator, Brakes, Horn, Headlights
- Actions: Steer, speed up, brake, blow horn, signal (headlights)
- Environment: Freeway

Lane Keeping Agent (LKA)

- Goals: Stay in current lane
- Percepts: Lane center, lane boundaries
- Sensors: Vision
- Effectors: Steering Wheel, Accelerator, Brakes
- Actions: Steer, speed up, brake
- Environment: Freeway

Conflict Resolution by Action Selection Agents

- **Override:** CAA overrides LKA
- **Arbitrate:** <u>if</u> Obstacle is Close <u>then</u> CAA <u>else</u> LKA
- **Compromise:** Choose action that satisfies both agents
- Any combination of the above
- **Challenges:** <u>Doing the right thing at the right time</u>

The Right Thing = The Rational Action

- **Rational Action:** The action that maximizes the expected value of the performance measure given the percept sequence to date
  - Rational = Best ?
  - Rational = Optimal ?
  - Rational = Omniscience ?
  - Rational = Clairvoyant (supernatural) ?
  - Rational = Successful ?

The Right Thing = The Rational Action

- **Rational Action:** The action that maximizes the expected value of the performance measure given the percept sequence to date

  - Rational = Best — Yes, to the best of its knowledge
  - Rational = Optimal — Yes, to the best of its abilities (incl. its constraints)
  - Rational ≠ Omniscience
  - Rational ≠ Clairvoyant (supernatural)
  - Rational ≠ Successful

Behavior and Performance of IAs

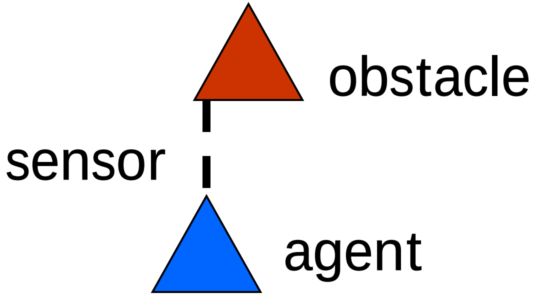
- **Perception** (sequence) **to Action Mapping:** $f : \mathcal{P}^* \rightarrow \mathcal{A}$
  - **Ideal mapping:** specifies which actions an agent ought to take at any point in time
  - **Implementation:** Look-Up-Table, Closed Form, Algorithm, etc.

- **Performance measure:** a *subjective* measure to characterize how successful an agent is (e.g., speed, power usage, accuracy, money, etc.)

- (degree of) **Autonomy:** to what extent is the agent able to make decisions and take actions on its own?

“Look-Up-Table” Agent

Distance	Action
10	No action
5	Turn left 30 degrees
2	Stop



Closed Form

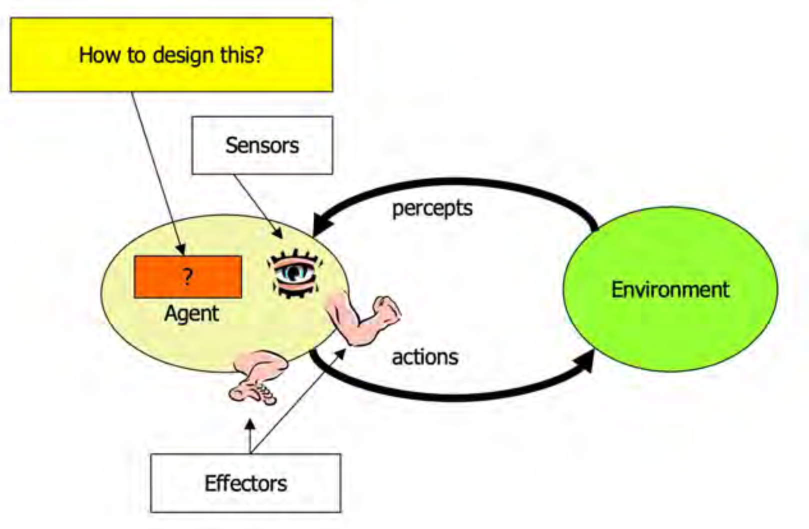
- Output (degree of rotation) = F(distance)
- E.g., F(d) = 10/d (distance cannot be less than 1/10)

How is an Agent different from other software?

Is A/C Thermostats an Agent?
Is Zoom an agent?

- Agents are **autonomous**, that is, they act on behalf of the user
- Agents contain some level of **intelligence**, from <u>fixed rules</u> to <u>learning</u> engines that allow them to adapt to changes in the environment
- Agents don't only act **reactively**, but sometimes also **proactively**
- Agents have **social ability**, that is, they communicate with the user, the system, and other agents as required
- Agents may also **cooperate** with other agents to carry out more complex tasks than they themselves can handle
- Agents may **migrate** from one system to another to access remote resources or even to meet other agents

Types of Environments



- Characteristics
  - Accessible vs. Inaccessible
  - Deterministic vs. Nondeterministic
  - Episodic vs. Non-episodic
  - Hostile vs. Friendly
  - Static vs. Dynamic
  - Discrete vs. Continuous

Environment Types

- Characteristics
  - Accessible (observable) vs. inaccessible (partial observable)
    - Accessible: sensors give access to **complete** state of the environment.
  - Deterministic vs. nondeterministic
    - Deterministic: the next state can be determined based on the current state and the action.
  - Episodic (history-insensitive) vs. non-episodic (Sequential, history-sensitive)
    - Episode: each perceive and action pairs
    - In episodic environments, the quality of action does not depend on the previous episode and does not affect the next episode.
    - Example: Episodic: mail sorting system; non-episodic: chess (why? “castling”), Go?

Environment Types

- Characteristics
  - Hostile vs. friendly
  - Static vs. dynamic
    - Dynamic if the environment changes during deliberation
  - Discrete vs. continuous
    - Chess vs. driving

Environment types

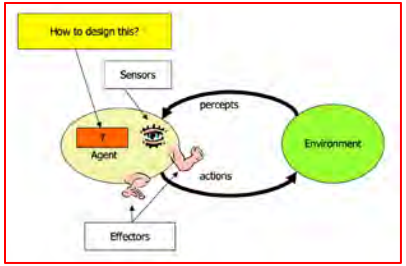
Environment	Accessible	Deterministic	Episodic	Static	Discrete
Operating System					
Virtual Reality					
Office Environment					
Mars					

Environment types

Environment	Accessible	Deterministic	Episodic	Static	Discrete
Operating System	Yes	Yes	No	No	Yes
Virtual Reality	Yes	Yes	Yes/no	No	Yes/no
Office Environment	No	No	No	No	No
Mars	No	Semi	No	Semi	No

The environment types largely determine the agent design.

Structure of Intelligent Agents



- **Agent** = architecture + program

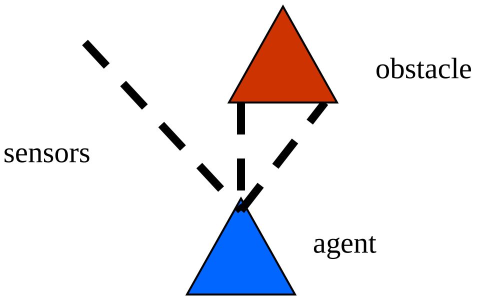
- **Agent Program:** the implementation of $f : \mathcal{P}^* \rightarrow \mathcal{A}$, the agent's perception-action mapping

function Skeleton-Agent(*Percept*) **returns** *Action*
  memory $\leftarrow$ UpdateMemory(memory, *Percept*)
  Action $\leftarrow$ ChooseBestAction(memory)
  memory $\leftarrow$ UpdateMemory(memory, *Action*)
return *Action*

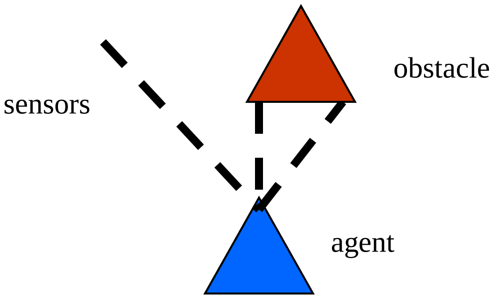
- **Architecture:** a device that can execute the agent program (e.g., general-purpose computer, specialized device, beobot, etc.)

Using a look-up-table to encode $f : \mathcal{P}^* \rightarrow \mathcal{A}$

- **Example:** Collision Avoidance
  - Sensors: 3 proximity sensors
  - Effectors: Steering Wheel, Brakes
- How to generate?
- How large?
- How to select action?



Using a look-up-table to encode $f : \mathcal{P}^* \rightarrow \mathcal{A}$



- Example: Collision Avoidance
  - Sensors: 3 proximity sensors
  - Effectors: Steering Wheel, Brakes

- How to generate: for each $p \in \mathcal{P}_l \times \mathcal{P}_m \times \mathcal{P}_r$ generate an appropriate action, $a \in \mathcal{S} \times \mathcal{B}$

- How large: size of table = # possible percepts times # possible actions = $|\mathcal{P}_l|\ |\mathcal{P}_m|\ |\mathcal{P}_r|\ |\mathcal{S}|\ |\mathcal{B}|$
  E.g., P = {close, medium, far}^{3}
  A = {left, straight, right} $\times$ {on, off}
  then size of table = 27 rows
- Total possible combinations (ways to fill table): 27*3*2=162

- How to select action? Search.

TYPES OF AGENT

Agent Types

- Reflex agents
- Reflex agents with internal states
- Goal-based agents
- Utility-based agents
- Learning agents

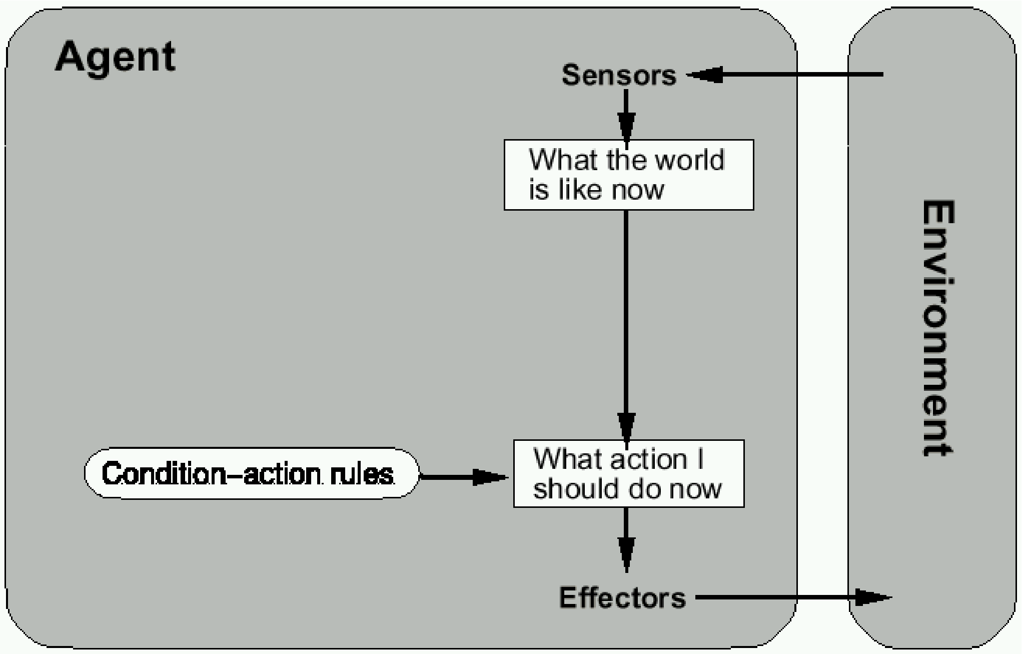
Agent Types

- Reflex agents
  - Reactive: No memory
- Reflex agents with internal states
  - W/o previous state, may not be able to make decision
    - E.g. brake lights at night.
- Goal-based agents
  - Goal information needed to make decision

Agent Types

- Utility-based agents
  - How well can the goal be achieved (degree of happiness)
  - What to do if there are conflicting goals?
    - Speed and safety
  - Which goal should be selected if several can be achieved?
- Learning agents
  - How can I adapt to the environment?
  - How can I learn from my mistakes?

Reflex Agents

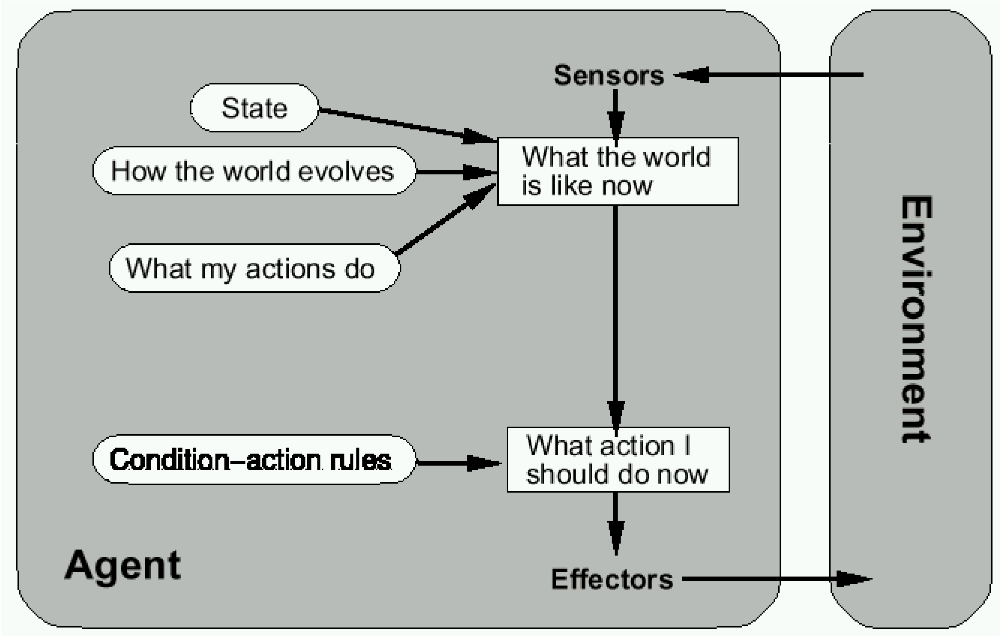


Reactive Agents

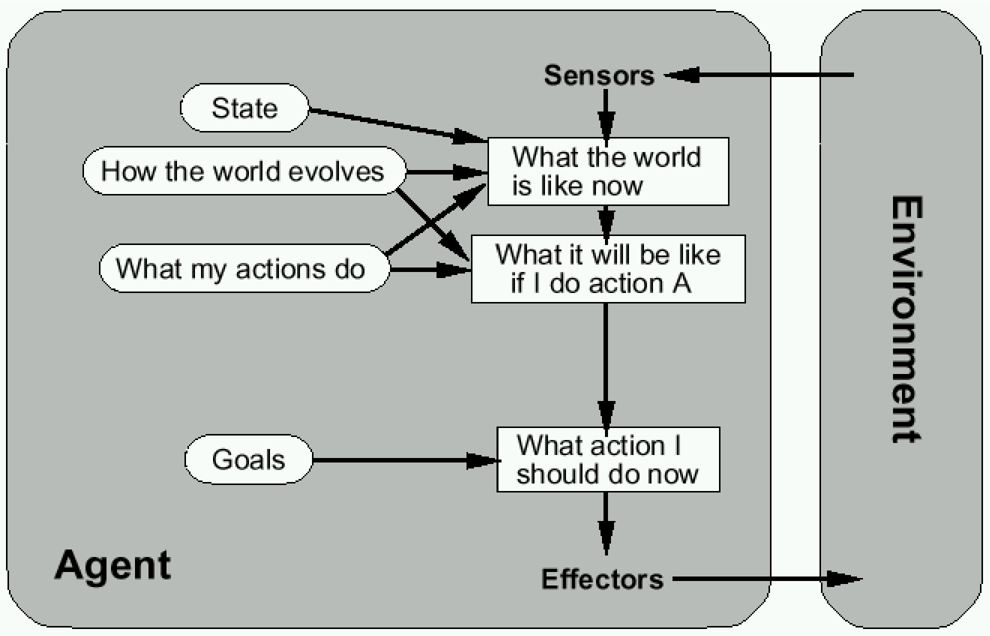
- Reactive agents do not have internal symbolic models.
- Act by stimulus-response to the current state of the environment.
- Each reactive agent is simple and interacts with others in a basic way.
- Complex patterns of behavior emerge from their interaction.

- **Benefits:** robustness, fast response time
- **Challenges:** scalability, how intelligent? how do you debug them?

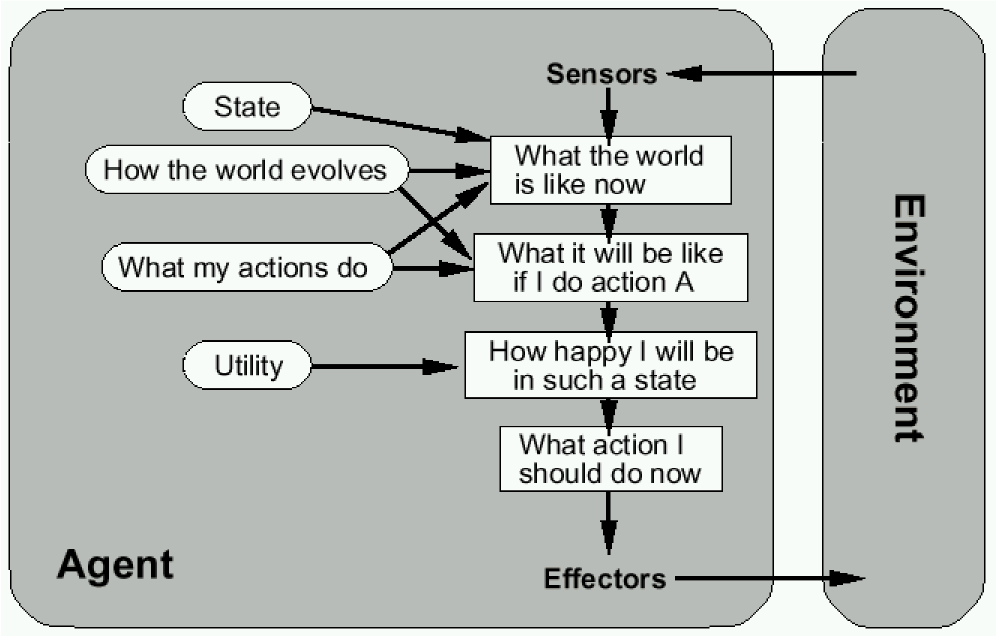
Reflex Agents with Internal States



Goal-Based Agents



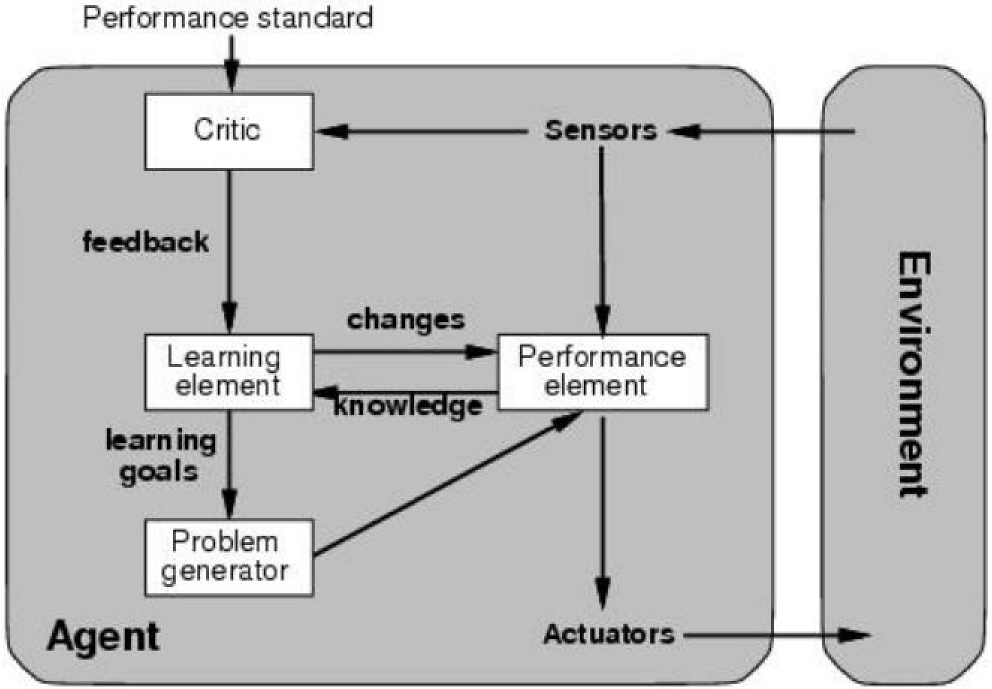
Utility-Based Agents



Learning Agents

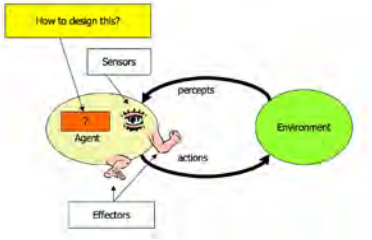
Critic: Determines outcomes of actions and gives feedback

Learning element: Takes feedback from critic and improves performance element



Problem generator: creates new experiences to promote learning

Summary on Intelligent Agents



- **Intelligent Agents:**
  - Anything that can be *viewed as* **perceiving** its **environment** through **sensors** and **acting** upon that environment through its **effectors** to maximize progress towards its **goals**.
  - PAGE (Percepts, Actions, Goals, Environment)
  - Described as a Perception (sequence) to Action Mapping: $f : \mathcal{P}^* \rightarrow \mathcal{A}$
  - Using look-up-table, closed form, etc.

- **Agent Types:** Reflex, state-based, goal-based, utility-based, learning

- **Rational Action:** The action that maximizes the expected value of the performance measure given the percept sequence to date