Intelligent Agents

Instructor
Santosh Khanal

Today's Topics

- Review of Previous Lecture
- Agents
- Factors defining Agent
- Environment & its types
- Agent Architecture

Artificial Intelligence

- Views of AI fall into four categories:

Thinking humanly	Thinking rationally
Acting humanly	Acting rationally

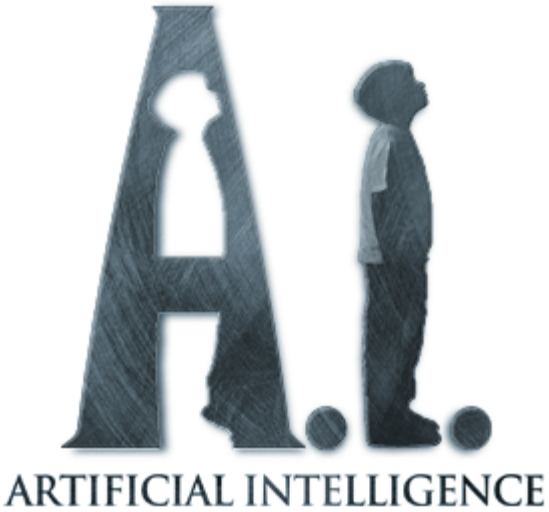
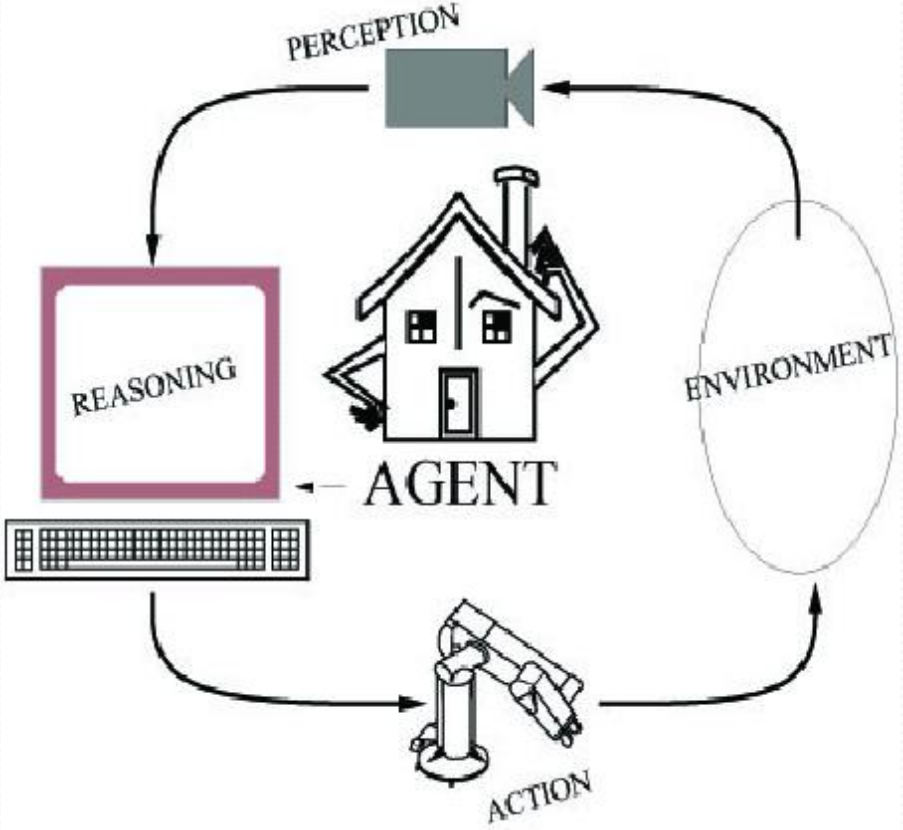


Fig: source google

Components of an AI System



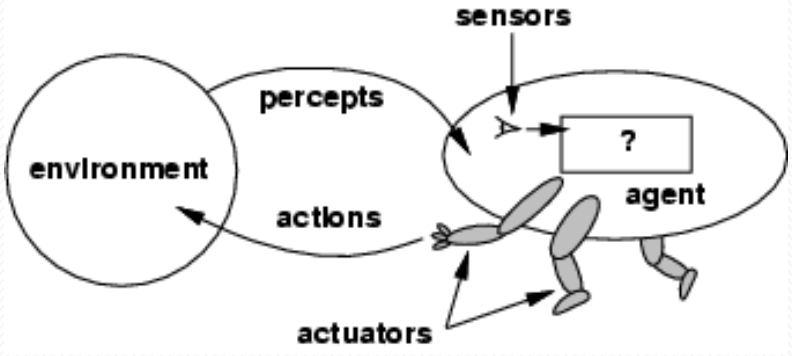
An agent perceives its environment through sensors and acts on the environment through actuators.

Human: sensors are eyes, ears, actuators (effectors) are hands, legs, mouth.

Robot: sensors are cameras, sonar, lasers, ladar, bump, effectors are grippers, manipulators, motors

The agent's behavior is described by it function that maps percept to action.

Agents



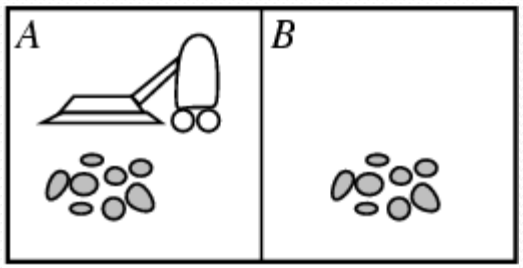
- The agent function maps from percept histories to actions:

$$[f: \mathcal{P}^{\star} \rightarrow \mathcal{A}]$$

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

Vacuum-cleaner world

- Example



- Sensors: camera
- Percepts: location and contents, e.g., [A,Dirty]
- Actions: *Left*, *Right*, *Suck*, *NoOp*

Rational Agents

- "do the right thing"

- For each possible percept sequence, a rational agent should select an action that is expected to maximize its performance measure, given the evidence provided by the percept sequence and whatever built-in knowledge the agent has.
- Rationality is distinct from omniscience
- Autonomous Agent

Rationality

- At a given time it depends upon
  - The performance measure that define the success criteria
  - The agent's prior knowledge of the environment
  - The actions that the agent can perform
  - The agent's percept sequence

Factors Defining Agents

- PEAS: Performance measure, Environment, Actuators, Sensors

- Example
  - performance measure of a vacuum-cleaner agent could be amount of dirt cleaned up, amount of time taken, amount of electricity consumed, amount of noise generated, etc.
  - Environment??
  - Actuators??
  - Sensors??

PEAS

- Must first specify the setting for intelligent agent design

- Consider, e.g., the task of designing an automated taxi driver: & PEAS
  - Performance measure: Safe, fast, legal, comfortable trip, maximize profits
  - Environment: Roads, other traffic, pedestrians, customers
  - Actuators: Steering wheel, accelerator, brake, signal, horn
  - Sensors: Cameras, sonar, speedometer, GPS, odometer, engine sensors, keyboard

PEAS

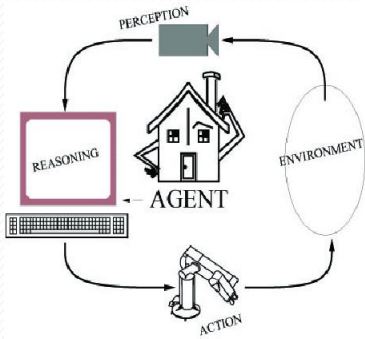
- Agent: Medical diagnosis system
- Performance measure: Healthy patient, minimize costs, lawsuits
- Environment: Patient, hospital, staff
- Actuators: Screen display (questions, tests, diagnoses, treatments, referrals)
- Sensors: Keyboard (entry of symptoms, findings, patient's answers)

PEAS

- Agent: Part-picking robot
- Performance measure: Percentage of parts in correct bins
- Environment: Conveyor belt with parts, bins
- Actuators: Jointed arm and hand
- Sensors: Camera, joint angle sensors

Talespin

- Performance: Entertainment value of generated story
- Environment: Generate text-based stories that are creative and understandable
  - One day Joe Bear was hungry. He asked his friend Irving Bird where some honey was. Irving told him there was a beehive in the oak tree. Joe threatened to hit Irving if he didn't tell him where some honey was.
  - Henry Squirrel was thirsty. He walked over to the river bank where his good friend Bill Bird was sitting. Henry slipped and fell in the river. Gravity drowned. Joe Bear was hungry. He asked Irving Bird where some honey was. Irving refused to tell him, so Joe offered to bring him a worm if he'd tell him where some honey was. Irving agreed. But Joe didn't know where any worms were, so he asked Irving, who refused to say. So Joe offered to bring him a worm if he'd tell him where a worm was. Irving agreed. But Joe didn't know where any worms were, so he asked Irving, who refused to say. So Joe offered to bring him a worm if he'd tell him where a worm was...
- Actuators: Add word/phrase, order parts of story
- Sensors: Dictionary, Facts and relationships stored in database
- Reasoning: Planning



Agent Type	Percepts	Actions	Goals	Environment
Medical diagnosis system	Symptoms, findings, patient's answers	Questions, tests, treatments	Healthy patient, minimize costs	Patient, hospital
Satellite image analysis system	Pixels of varying intensity, color	Print a categorization of scene	Correct categorization	Images from orbiting satellite
Part-picking robot	Pixels of varying intensity	Pick up parts and sort into bins	Place parts in correct bins	Conveyor belt with parts
Refinery controller	Temperature, pressure readings	Open, close valves; adjust temperature	Maximize purity, yield, safety	Refinery
Interactive English tutor	Typed words	Print exercises, suggestions, corrections	Maximize student's score on test	Set of students

Environment Properties

- Fully observable vs. partially observable
- Deterministic vs. stochastic / strategic
- Episodic vs. sequential
- Static vs. dynamic
- Discrete vs. continuous
- Single agent vs. multiagent

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock						
Chess without a clock						

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker						

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

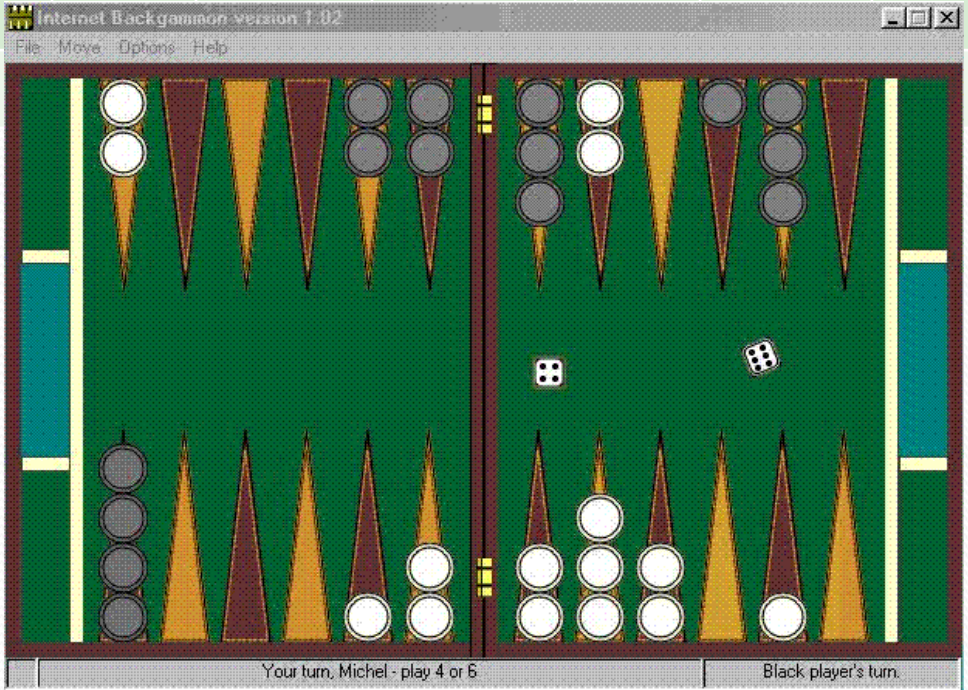
Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

Environment Examples

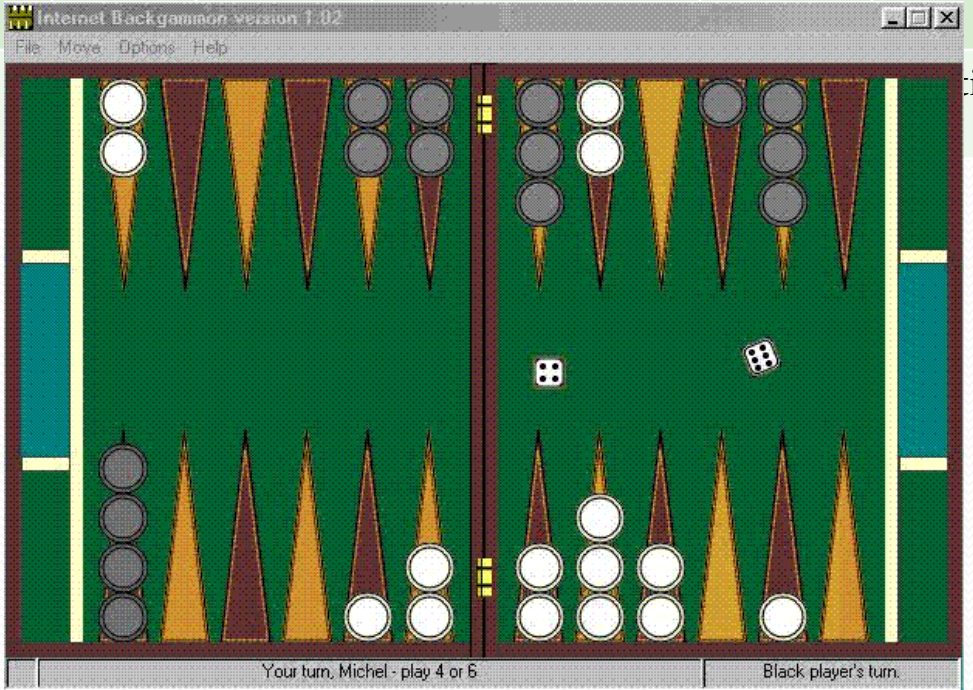
Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partia	Strategi	Sequentia	Stati	Discret	Multi
Backgammon						



Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

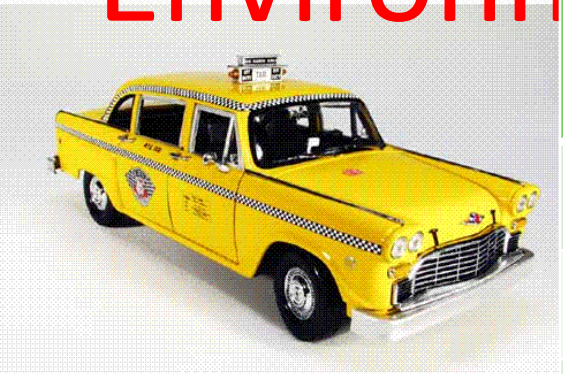
Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partia	Strategi	Sequentia	Stati	Discret	Multi
Backgammon						ti



Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

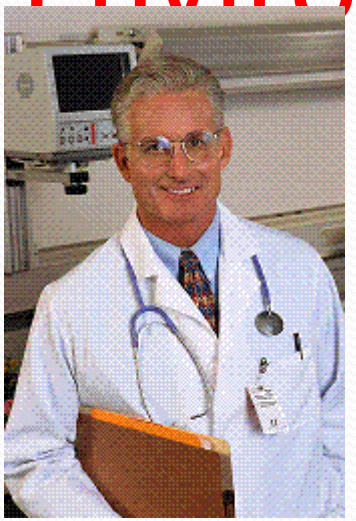
Environment Examples



Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

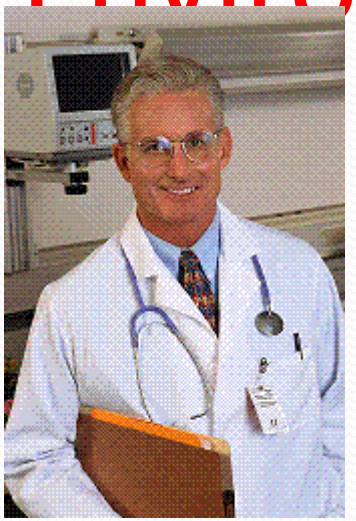
Environment Examples



Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis						

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

Environment Examples



Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single

Fully observable vs. partially observable
Deterministic vs. stochastic / strategic
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis						

Fully observable vs. partia
Deterministic vs. stochast
Episodic vs. sequential
Static vs. dynamic
Discrete vs. continuous
Single agent vs. multiagent

Environment Examples

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single

Fully observable vs. partially observable

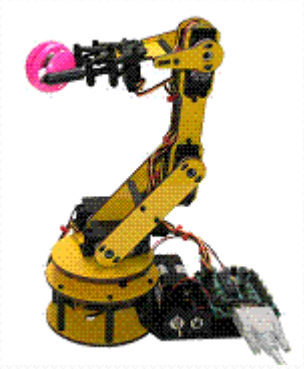
Deterministic vs. stochastic / strategic

Episodic vs. sequential
Static vs. dynamic

Discrete vs. continuous

Single agent vs.

Environment Examples



Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking						

Fully observable vs. partially observable

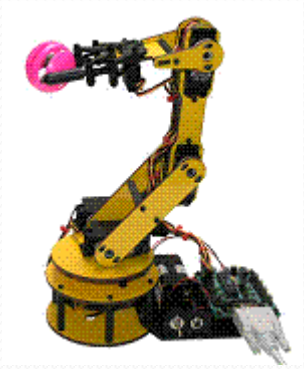
Deterministic vs. stochastic / strategic

Episodic vs. sequential
Static vs. dynamic

Discrete vs. continuous

Single agent vs.

Environment Examples



Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking	Fully	Deterministic	Episodic	Semi	Discrete	Single

Fully observable vs. partially observable

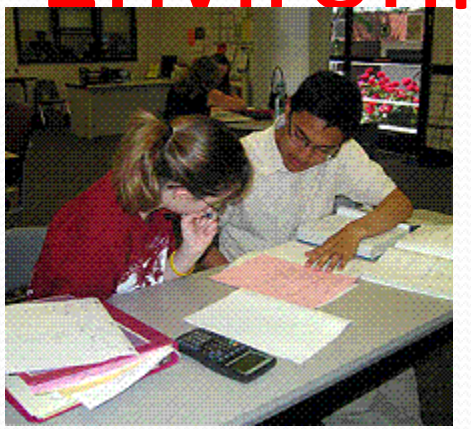
Deterministic vs. stochastic / strategic

Episodic vs. sequential
Static vs. dynamic

Discrete vs. continuous

Single agent vs.

Environment Examples



Fully observable vs. partially observable

Deterministic vs. stochastic / strategic

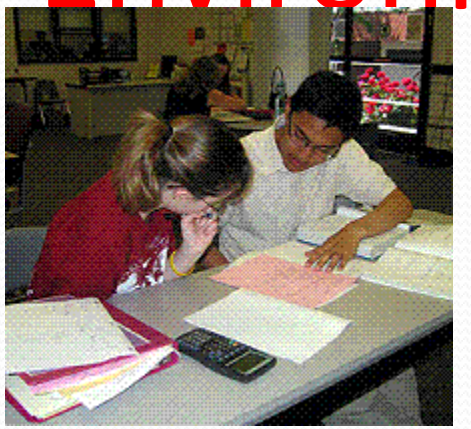
Episodic vs. sequential
Static vs. dynamic

Discrete vs. continuous

Single agent vs.

Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking	Fully	Deterministic	Episodic	Semi	Discrete	Single
Interactive English tutor						

Environment Examples



Fully observable vs. partially observable

Deterministic vs. stochastic / strategic

Episodic vs. sequential
Static vs. dynamic

Discrete vs. continuous

Single agent vs.

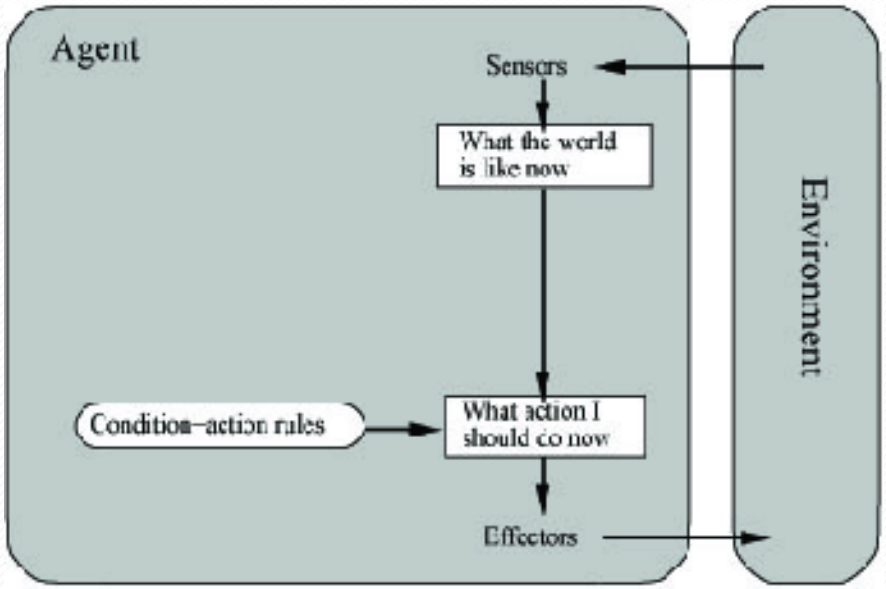
Environment	Observable	Deterministic	Episodic	Static	Discrete	Agents
Chess with a clock	Fully	Strategic	Sequential	Semi	Discrete	Multi
Chess without a clock	Fully	Strategic	Sequential	Static	Discrete	Multi
Poker	Partial	Strategic	Sequential	Static	Discrete	Multi
Backgammon	Fully	Stochastic	Sequential	Static	Discrete	Multi
Taxi driving	Partial	Stochastic	Sequential	Dynamic	Continuous	Multi
Medical diagnosis	Partial	Stochastic	Episodic	Static	Continuous	Single
Image analysis	Fully	Deterministic	Episodic	Semi	Discrete	Single
Robot part picking	Fully	Deterministic	Episodic	Semi	Discrete	Single
Interactive English tutor	Partial	Stochastic	Sequential	Dynamic	Discrete	Multi

Agent types

- Four basic types in order of increasing generality:
  - Simple reflex agents
  - Model-based reflex agents
  - Goal-based agents
  - Utility-based agents

Simple Reflex Agent

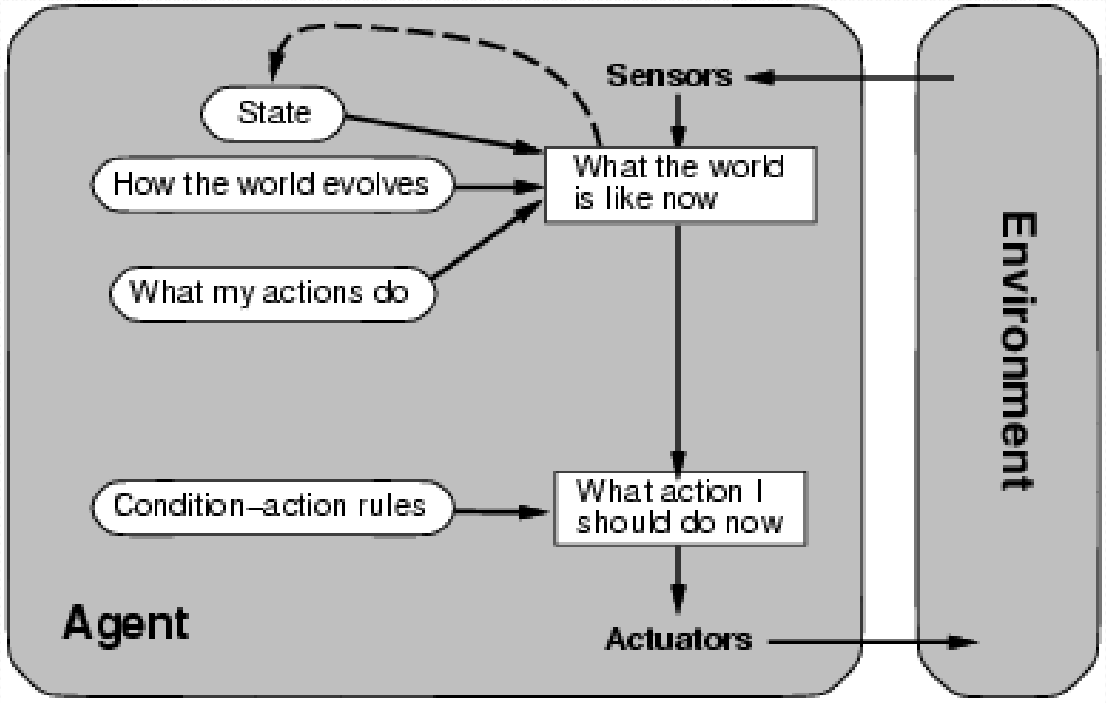
- Use simple "if then" rules
- Can be short sighted
- *Will only work if the environment is fully observable otherwise infinite loops may occur.*




```
SimpleReflexAgent(percept)
  state  = InterpretInput(percept)
  rule   = RuleMatch(state, rules)
  action = RuleAction(rule)
  Return action
```

If status=Dirty then return Suck
else if location=A then return Right
else if location=B then right Left

Model-based reflex agents

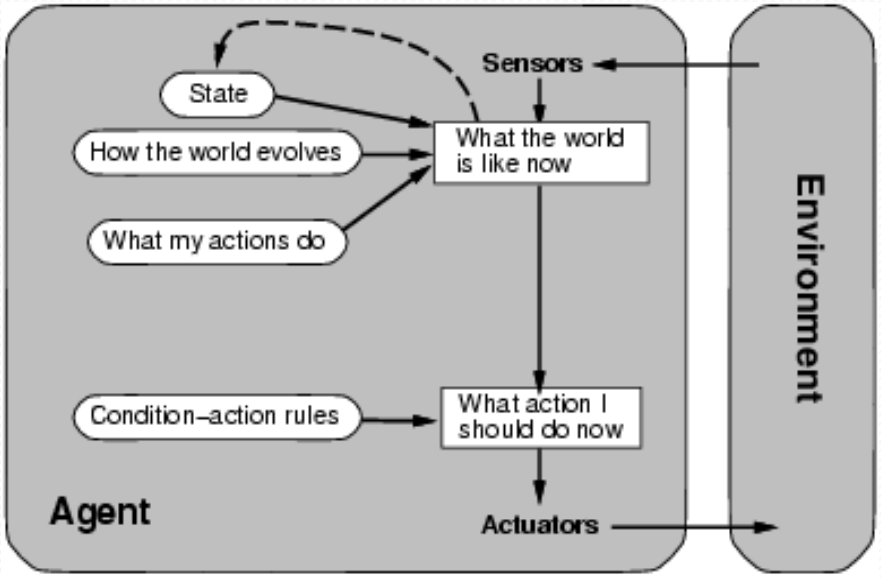


- To tackle *partially observable* environments.
  - Maintain internal state
- Over time update state using world knowledge
  - How does the world change.
  - How do actions affect world.

  $\Rightarrow$ *Model of World*

Model-based reflex agents

- Simple Reflex Agent with Internal State
- Store previously-observed information
- Can reason about unobserved aspects of current state

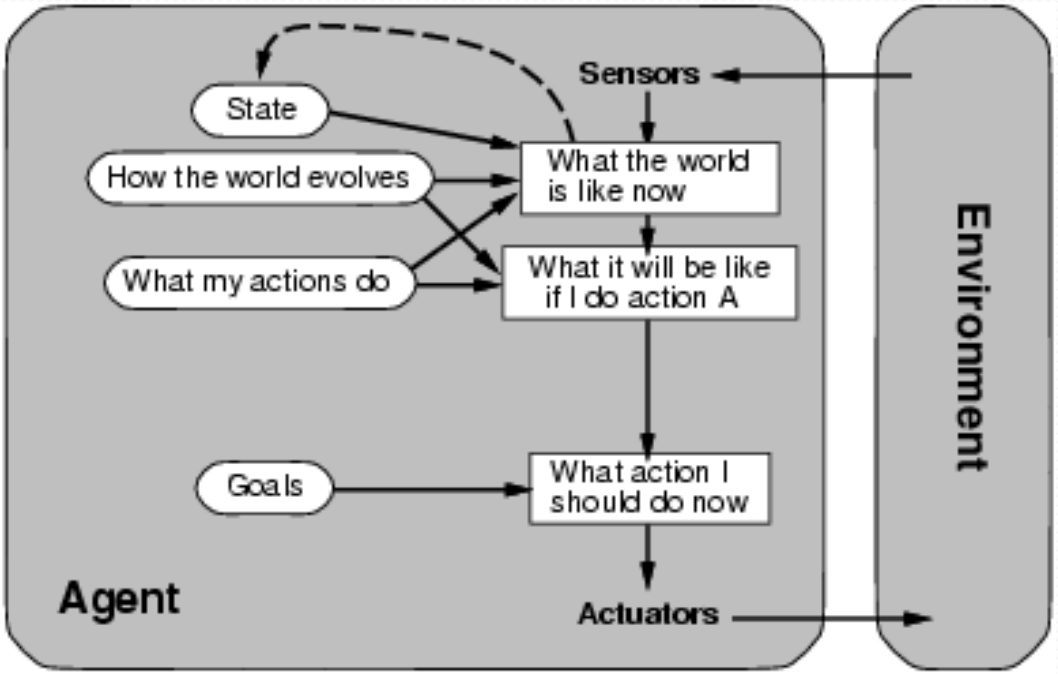



```
ReflexAgentWithState(percept)
  state  = UpdateDate(state,action,percept)
  rule   = RuleMatch(state, rules)
  action = RuleAction(rule)
  Return action
```

If status=Dirty then Suck
else if have not visited other square in >3 time units, go there

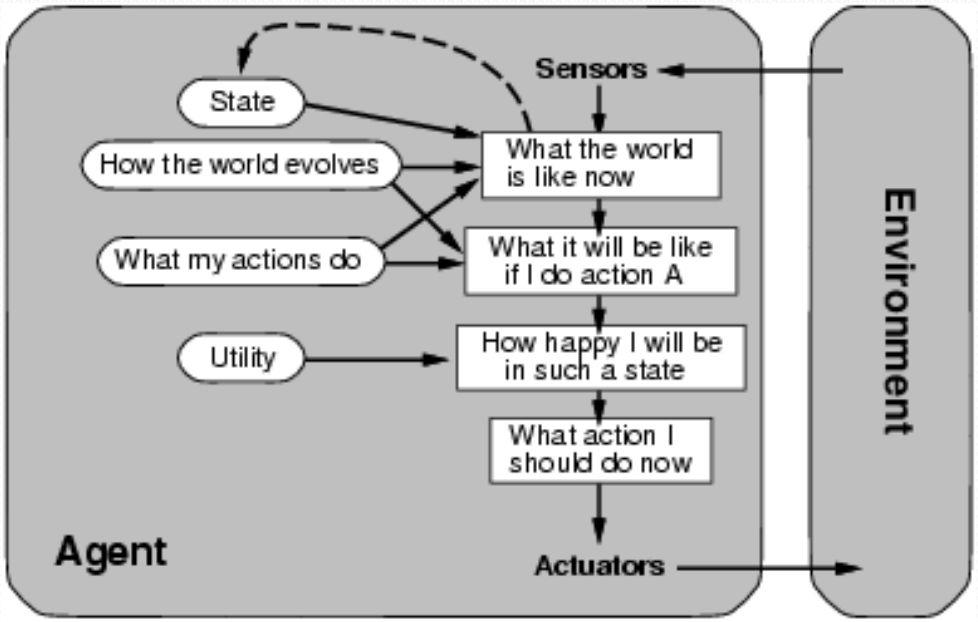
Goal-based agents

- The agent needs a goal to know which situations are *desirable*.
  - Things become difficult when long sequences of actions are required to find the goal.
- Typically investigated in **search** and **planning** research.
- Major difference: future is taken into account
- Is more flexible since knowledge is represented explicitly and can be manipulated.

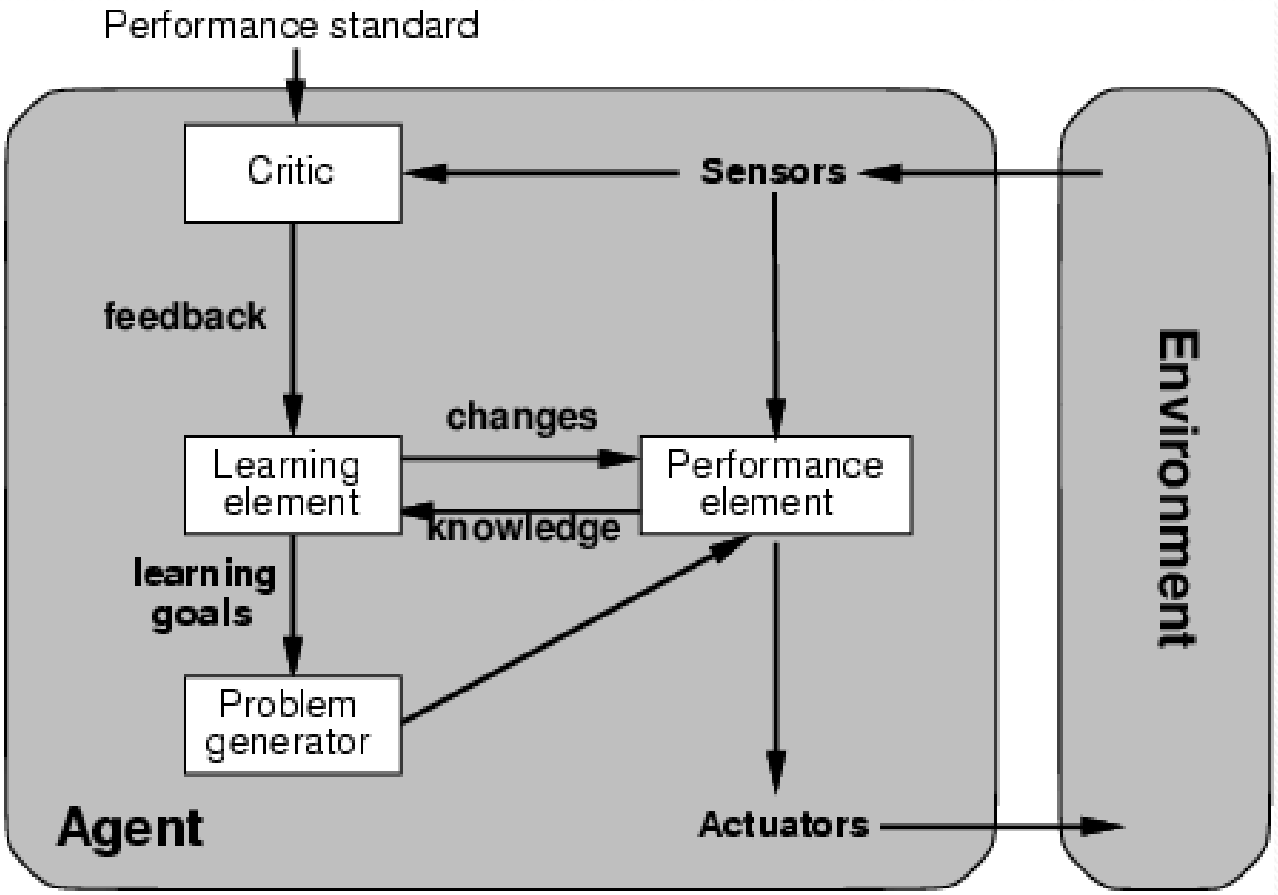


Utility-based agents

- Certain goals can be reached in different ways.
  - Some are better, have a higher utility.
- Utility function maps a (sequence of) state(s) onto a real number.
- Improves on goals:
  - Selecting between conflicting goals
  - Select appropriately between several goals based on likelihood of success.



Learning Agent



Summary: Intelligent Agents

- An **agent** perceives and acts in an environment, has an architecture, and is implemented by an agent program.
- Task environment – **PEAS** (**P**erformance, **E**nvironment, **A**ctuators, **S**ensors)
- The most challenging environments are inaccessible, nondeterministic, dynamic, and continuous.
- An **ideal agent** always chooses the action which maximizes its expected performance, given its percept sequence so far.
- An **agent program** maps from percept to action and updates internal state.
  - **Reflex agents** respond immediately to percepts.
    - simple reflex agents
    - model-based reflex agents
  - **Goal-based agents** act in order to achieve their goal(s).
  - **Utility-based agents** maximize their own utility function.
- All agents can improve their performance through **learning.**