Artificial Intelligence

Intelligent Agents

Banerjee & Rathi
(based on slides by Jonathan Voris & Sal Stolfo)

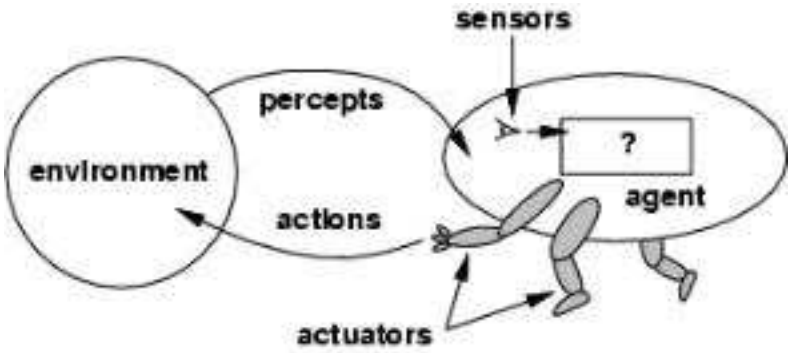
Outline

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

Agents

- An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators
- Human agent:
  - Sensors: Eyes, ears, and other
  - Actuators: Hands, legs, mouth, and other body
- Robotic agent:
  - Sensors: Cameras and infrared range finders
  - Actuators: Various motors
- Agents vs “non-agents” ?

Agents and Environments

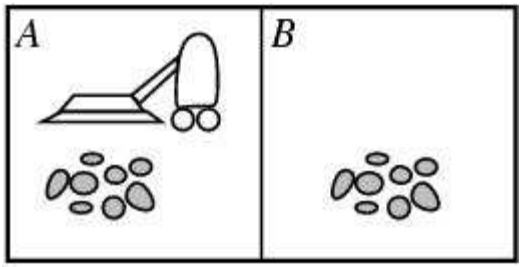


- The agent function maps from percept histories to actions:

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

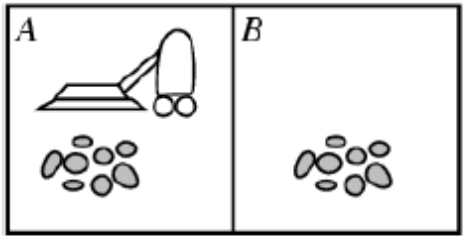
- The agent program runs on the physical architecture to produce the agent function
  - Agent function: Mathematical abstraction
  - Agent program: Concrete implementation
- Agent = architecture + program

Vacuum-cleaner world



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

Vacuum-cleaner world



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

Percept sequence	Actions
[A,Clean]	Right
[A, Dirty]	Suck
[B,Clean]	Left
[B,Dirty]	Suck
[A,Clean],[A,Clean]	Right
[A,Clean],[A,Dirty]	Suck
…	…
[A,Clean],[A.Clean],[A,Clean]	Right
[A,Clean],[A,Clean],[A,Clean]	Suck

One simple function is :

if the current square is dirty then suck, otherwise move to the other square

Rational Agents

- Recall: Rational = Doing the right thing
  - Okay, but what's that?
- What are the *consequences* of the agent's actions on the *state* of its *environment?*
  - Sequence of agent actions -> sequence of environmental states
- How to control agent?
  - Introduce *performance measure*
  - An objective criterion for success of an agent's behavior
- Possible performance measures for vacuum world?
- Evaluate performance of environment or that of agent?

Rational Agents

- Rational Agent: 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.

Rational Agents

- Rationality is relative to a performance measure
- Once you have an agent, judge rationality based on:
  - Performance measure
  - Environmental knowledge
  - Possible actions
  - Sensors and accuracy

Rational Agents

- Rationality is distinct from omniscience
  - Optimizing *expected* performance vs *observed* performance
  - Only relying on observations so far
- Agents can perform actions in order to modify future percepts so as to obtain useful information
  - Information gathering: actions which intend to modify future precepts
  - Exploration
- An agent is autonomous if its behavior is determined by its own experience (with ability to learn and adapt)
  - Agent street smarts vs book smarts

Environments

- Four elements of a task environment: PEAS
  - Performance measure
  - Environment
  - Actuators
  - Sensors
- Must first specify the setting for intelligent agent design

Environments

- Must first specify the setting for intelligent agent design
- Consider, e.g., the task of designing an automated taxi driver:
  - 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

Environments

- 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)

Environments

- 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

Environments

- Agent: Interactive English tutor
- Performance measure: Maximize student's score on test
- Environment: Set of students
- Actuators: Screen display (exercises, suggestions, corrections)
- Sensors: Keyboard

Environment types

- Fully observable (vs. partially observable): An agent's sensors give it access to the complete state of the environment at each point in time.
- Deterministic (vs. stochastic): The next state of the environment is completely determined by the current state and the action executed by the agent. (If the environment is deterministic except for the actions of other agents, then the environment is strategic)
- Episodic (vs. sequential): The agent's experience is divided into atomic "episodes" (each episode consists of the agent perceiving and then performing a single action), and the choice of action in each episode depends only on the episode itself.
  - Next episode not affected by actions in previous episode

Environment types

- Static (vs. dynamic): The environment is unchanged while an agent is deliberating. (The environment is semidynamic if the environment itself does not change with the passage of time but the agent's performance score does)
- Discrete (vs. continuous): A limited number of distinct, clearly defined percepts and actions.
- Single agent (vs. multiagent): An agent operating by itself in an environment.

Environment types

	Chess with a clock	Chess without a clock	Taxi driving
Fully observable	Yes	Yes	No
Deterministic	Strategic	Strategic	No
Episodic	No	No	No
Static	Semi	Yes	No
Discrete	Yes	Yes	No
Single agent	No	No	No

- The environment type largely determines the agent design
- The real world is (of course) partially observable, stochastic, sequential, dynamic, continuous, multi-agent

Agent Functions and Programs

- An agent is completely specified by the *agent function* mapping percept sequences to actions
  - Describes agent's behavior
- Job of AI: design an *agent program* which implements the agent function
  - Describes how the agent works
- Program must suit architecture
- Inputs:
  - Agent Program: Current percept
  - Agent Function: Percept history

Agent Functions and Programs

- One agent function (or a small equivalence class) is *rational*
- Aim: find a way to implement the rational agent function concisely

Table Lookup Agent

- Drawbacks:
  - Huge table
  - Take a long time to build the table
  - No autonomy
  - Even with learning, need a long time to learn the table entries

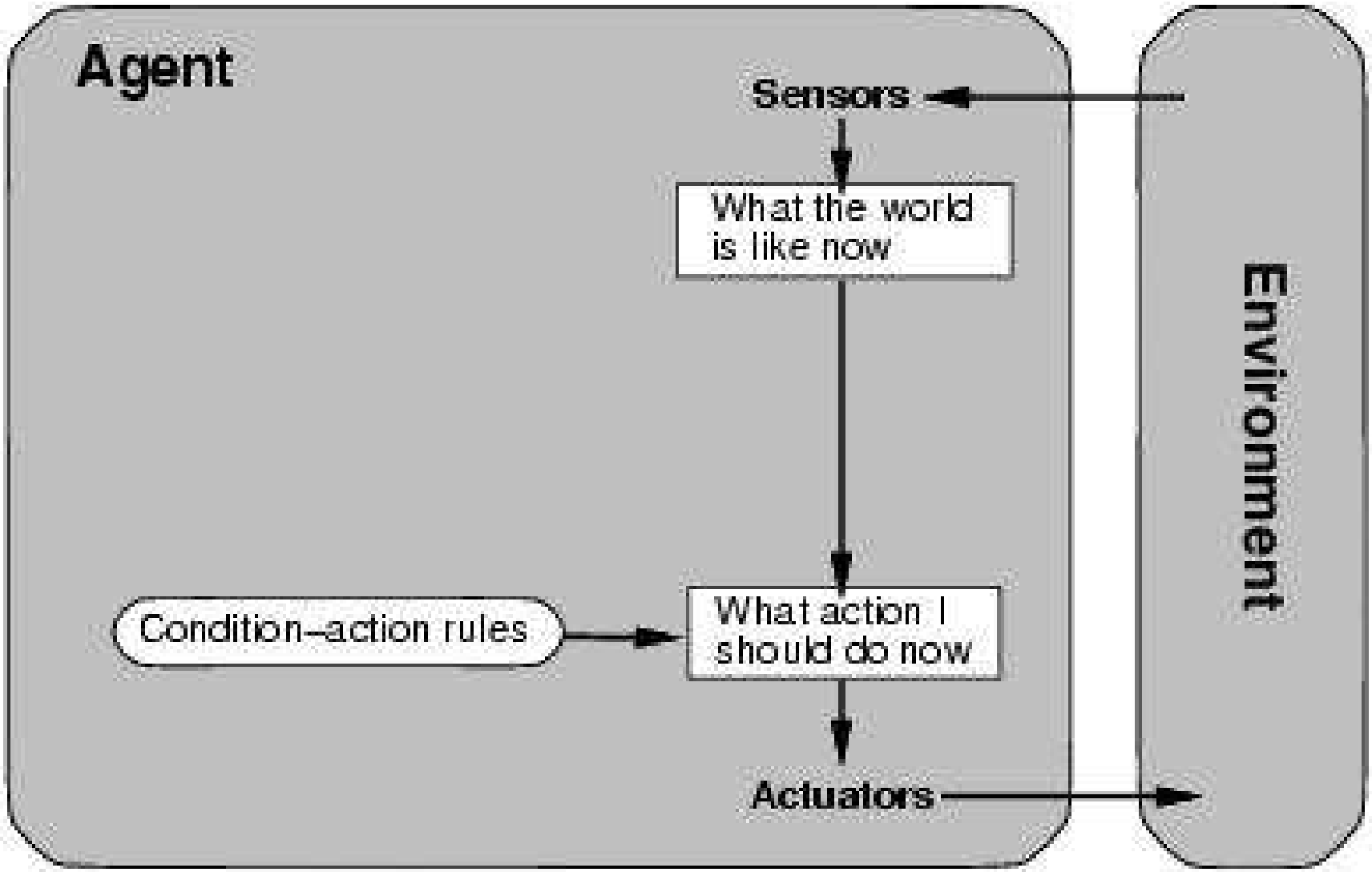
Table Lookup Agent

- Why a chess table agent is a bad idea:
  - Not enough space in universe to store table
  - Would take longer than your life to build
  - Too massive to be learned through observation
  - Where do you start?
- Reminder: The fact that a program can find a solution in principle does not mean that the program contains *any* of the mechanisms needed to find it in practice

Simple Reflex Agents

- Forget the precept history!
- Can’t we just act on the current state?

Simple Reflex Agents



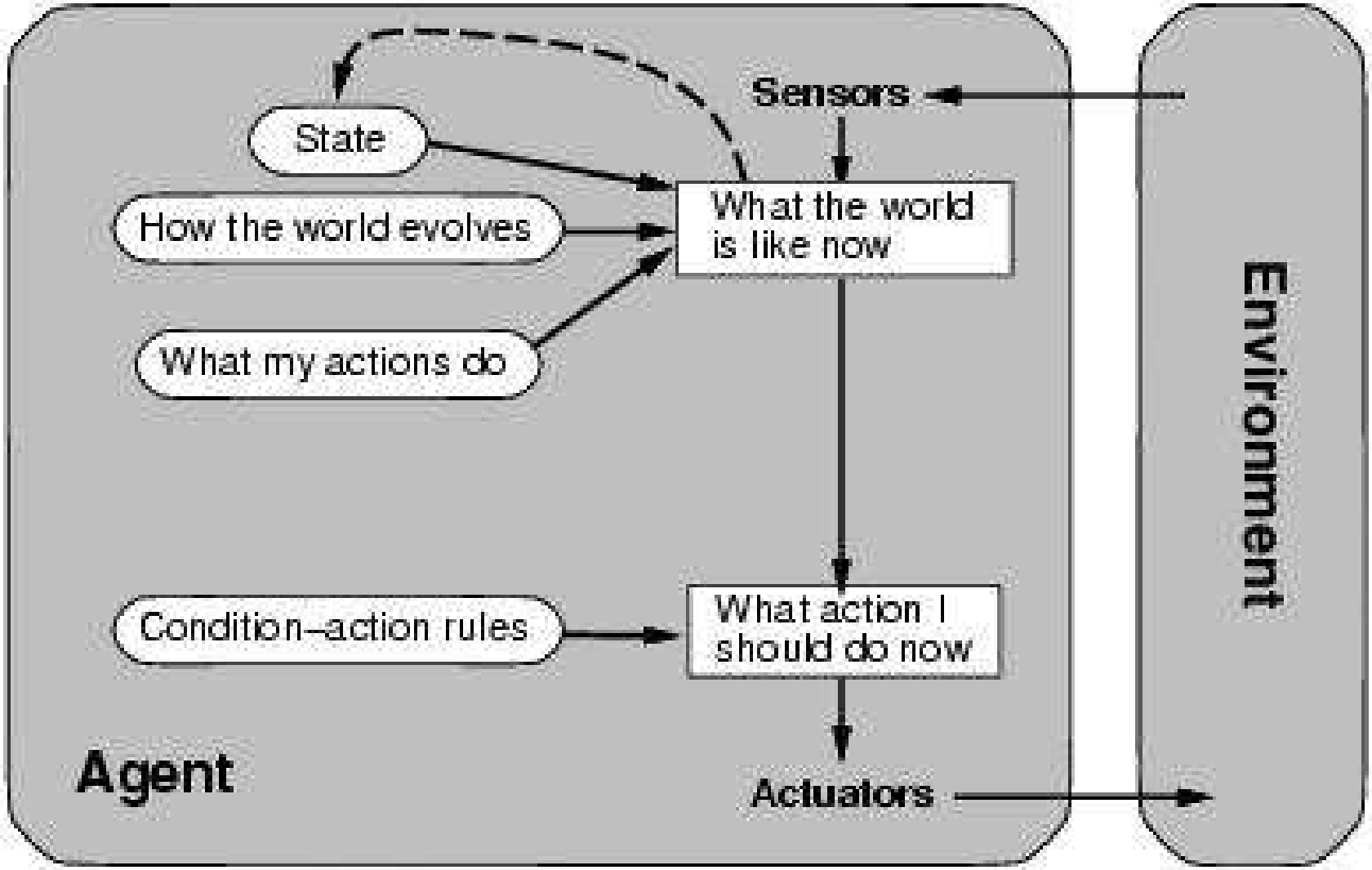
Simple Reflex Agents

- Pros:
  - Simple
- Cons:
  - For this to produce rational agents the environment must be…?
  - Often yield infinite loops
    - Randomization can help but only so much

Model-based Reflex Agents

- That didn't turn out so well
- Let's try keeping track of what we can't currently observe
- Internal state to track
  - Is that car braking?
  - Is there a car in my blind spot??
  - Where did I park???

Model-based Reflex Agents



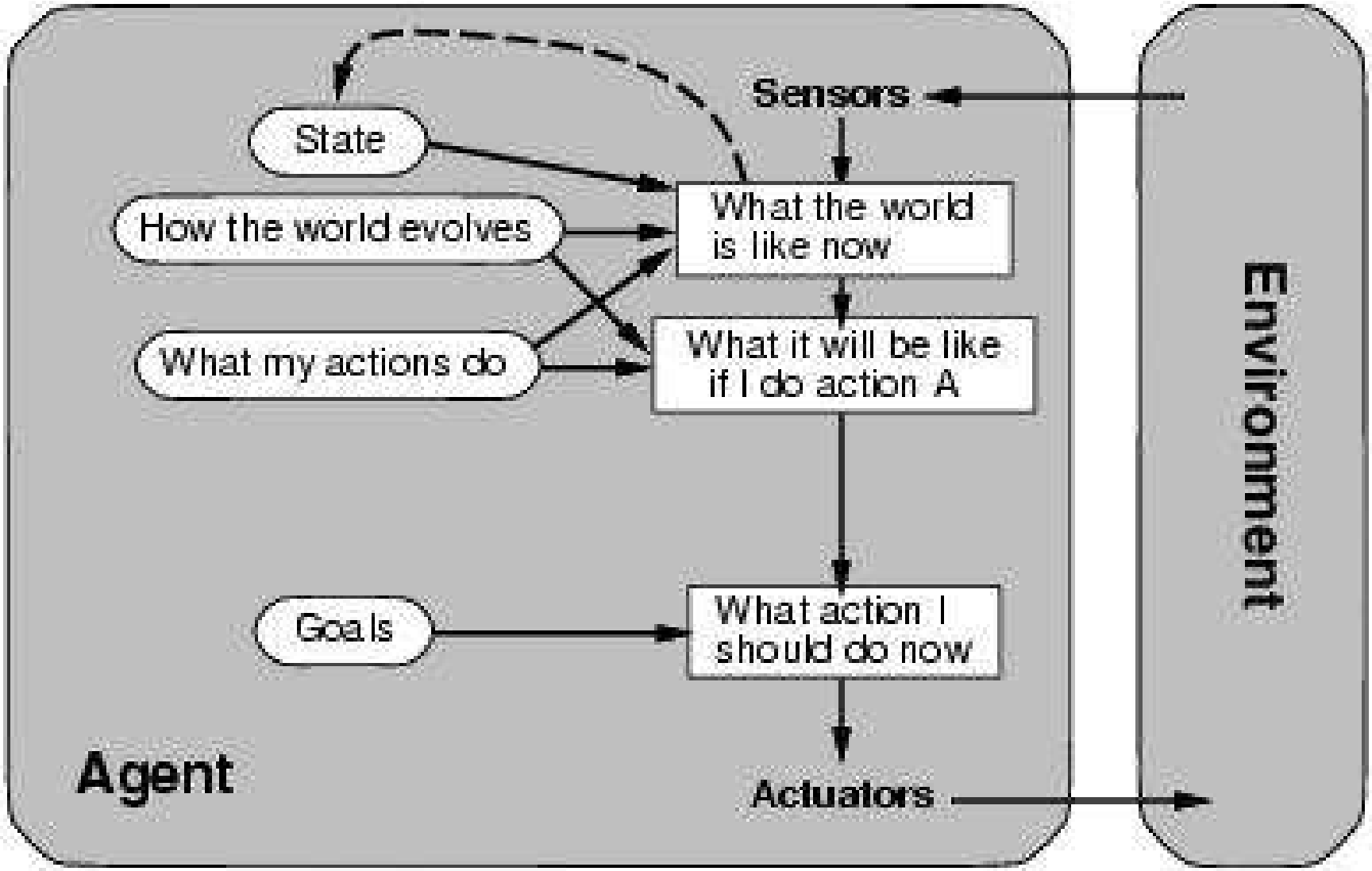
Model-based Reflex Agents

- Getting better!
- But sometimes model is insufficient:
  - Impossible to model hidden world accurately and reliably
  - What to do when environment doesn't imply action?
    - i.e., fork in the road

Goal-based agents

- Let's introduce a *goal* concept
  - Describe desirable situations
- Agent will account for the future

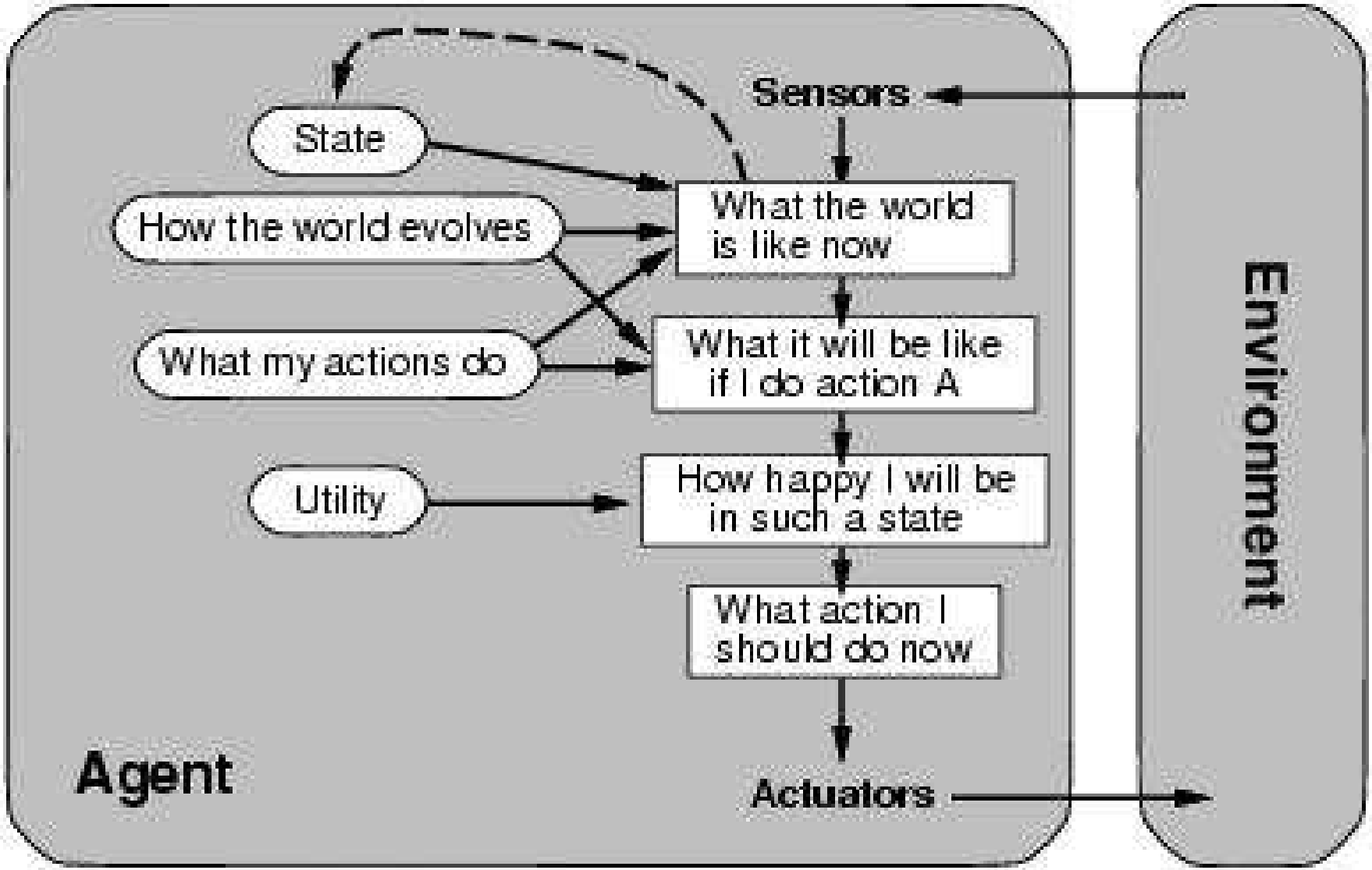
Goal-based agents



Utility-based Agents

- Goals are binary
  - Did you meet it or not?
- What about a relative measure of achievement?
- Economists named this *utility*
- *Utility function*: an agent's internal *performance measure*
- An agent is *rational* if its *utility function* matches its environments *performance measure*

Utility-based Agents



Utility-based Agents

- Utility function is not always needed to be rational
  - But rational agents must act like they have one
- Needed when:
  - Goals conflict
  - Uncertain goals

Learning Agents

- That's nice – but how do you program these things?
- By hand?
  - Turing says: That's for chumps
- Let the agent figure things out by itself!
- Learning element
  - Makes improvements to performance element
- Performance element
  - Selects actions
  - Previously this was the whole agent

Learning Agents

- New components:
- Learning element
  - Makes improvements to performance element
- Performance element
  - Selects actions
  - Previously this was the whole agent
- Critic
  - Gives performance feedback to learning element
  - Needed because precepts don’t capture performance
- Problem generator
  - Suggests innovative actions

Learning Agents