COMP111: Artificial Intelligence

Section 3. Intelligent Agents

Frank Wolter

Content

- The notion of intelligent agents (following Russell and Norvig);
- Task Environments (PEAS): Performance measure, environment, actuators, sensors;
- Task environment classification: fully observable vs partially observable, deterministic vs stochastic, episodic vs sequential, static vs dynamic, discrete vs continuous;
- Agent classification: simple reflex agents, model-based reflex agents, goal-based agents, utility-based agents, learning agents

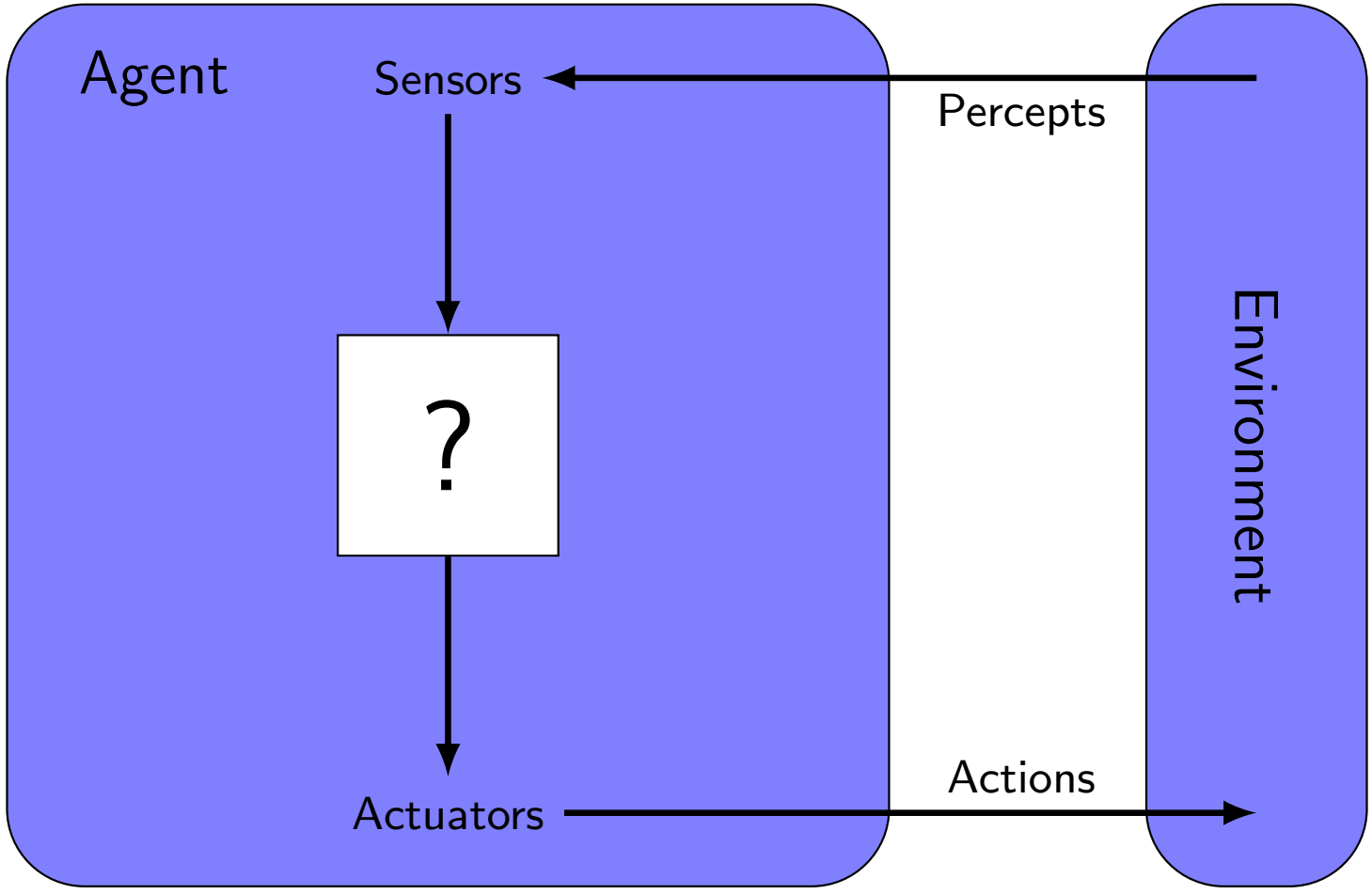
Intelligent Agents

- The intelligent entities that we engineer in AI are known as agents.
- A popular characterisation by Russell and Norvig is:

  > *An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators.*

- A human agent has eyes, ears and other organs as sensors and hands, legs and other body parts as actuators.
- A robotic agent might have cameras as sensors and various motors as actuators.
- A software agent might receive keystrokes and file contents as sensor inputs and might act on the environment by writing files.
- It is reasonable to say that this module is about tools for designing and implementing rational agents.

Agent



Task Environments

Agents are situated within task environments, which differ in accordance with the particular problem area that the agent is designed to address.
When we design agents to solve particular problems, we must specify the task environment as fully as possible. Four elements (PEAS) to take into account (from Russell and Norvig):

- Performance measure: the criteria by which we can measure the success of an agent's behaviour.
- Environment: the external environment that the agent inhabits.
- Actuators: the means by which the agent acts within its environment.
- Sensors: the means by which the agent senses its environment.

Example 1

Consider a taxi driver agent. Its PEAS description might be as follows:

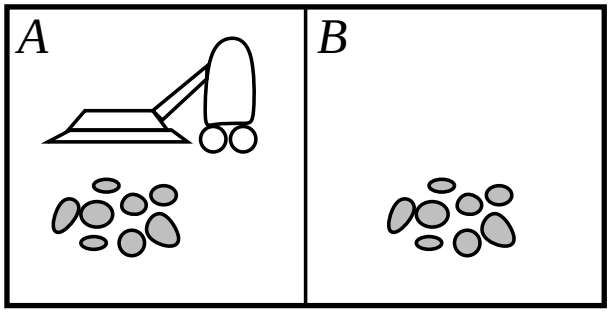
- Performance measure: safe, fast, legal, comfortable, etc.;
- Environment: roads, other traffic, pedestrians, customers;
- Actuators: steering, accelerator, brake, signal, display, etc.;
- Sensors: cameras, sonar, speedometer, GPS, engine sensors, keyboard.

Example 2

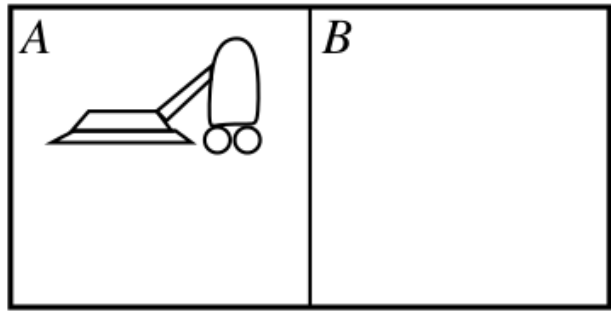
Consider an agent used for medical diagnosis. Its PEAS description might be as follows:

- Performance measure: health of patient, costs of treatment.
- Environment: patient, hospital, staff.
- Actuators: display questions, tests, diagnoses and treatments.
- Sensors: keyboard entry of patient's symptoms, responses to questions and findings.

Example 3: Vacuum cleaner world



Dirt in both rooms



No dirt in any room

- Performance measure: clean all rooms quickly;
- Environment: a vacuum cleaner located in one of two rooms, each possibly containing dirt;
- Actuators: move left/right, suck up the dirt, do nothing;
- Sensors: in which room is the vacuum cleaner? Is there dirt in that room?

Task Environments

The properties of the task environment that the agent inhabits may differ greatly, depending upon the particular application area. Russell and Norvig have given a classification of the different types of properties of agent environments:

- Fully observable vs partially observable
- Deterministic vs stochastic
- Episodic vs sequential
- Static vs dynamic
- Discrete vs continuous

Fully Observable vs partially observable

Fully observable environment: one in which the agent can fully obtain complete, up-to-date info about the environment's state. Many moderately complex environments are only partially observable.

- Fully observable environments are more convenient as the agent does not need to maintain any internal state (including memory) to keep track of the environment.
- Fully observable environment examples: a crossword puzzle, the game of chess.
- Partially observable environment examples: the everyday physical world (driving a car, playing football), the card game poker.

Deterministic vs Stochastic

Deterministic environment: one in which any action has a single guaranteed effect - there is no uncertainty about the state that will result from performing an action. In stochastic environment, the effect is not guaranteed.
This definition applies from the point of view of the agent.

- If the environment is deterministic except for the actions of other agents, the environment is said to be strategic.
- Deterministic environment examples: a crossword puzzle, the game of chess.
- Stochastic environment examples: medical diagnosis, the card game poker, driving a car (the physical world).

Episodic vs Sequential

Episodic environment: one where the performance of an agent is dependent on a number of discrete episodes with no link between its performance in different episodes. In sequential environments different episodes are linked.

- In episodic environments the agent can decide what action to perform based only on the current episode without having to reason about the interactions between this and future episodes.
- In sequential environments the current decision could affect all future decisions.
- Episodic environment examples: a mail sorting system, defect detection on an assembly line.
- Sequential environment examples: a crossword puzzle, the card game poker, driving a car.

Static vs dynamic

Static environment: one that can be assumed to remain unchanged whilst the agent is deliberating.
Dynamic environment: one that has other processes operating on it, and hence changes whilst the agent is deliberating.
Static environments are easier to deal with. The agent does not need to keep observing the environment whilst deciding how act, nor need it worry about time elapsing.

- Static environment examples: a crossword puzzle, the card game poker, the game of backgammon.
- Dynamic environment examples: medical diagnosis, the physical world.

Discrete vs Continuous

Discrete environment: one that contains a fixed, finite number of distinct states.
The distinction applies to the state of the environment, the way in which time is handled, the percepts and actions of the agent.
Continuous environments provide greater challenges for agent designers.
Discrete environment examples: a crossword puzzle, the game of chess.
Continuous environment exmples: medical diagnosis, driving a car.

Intelligent Agents: a classification

We classify intelligent agents according to how they map percepts to their actions.
Russell and Norvig identify five basic kinds of agents that underpin most intelligent systems:

- Simple reflex agents
- Model-based reflex agents
- Goal-based agents
- Utility-based agents
- Learning agents

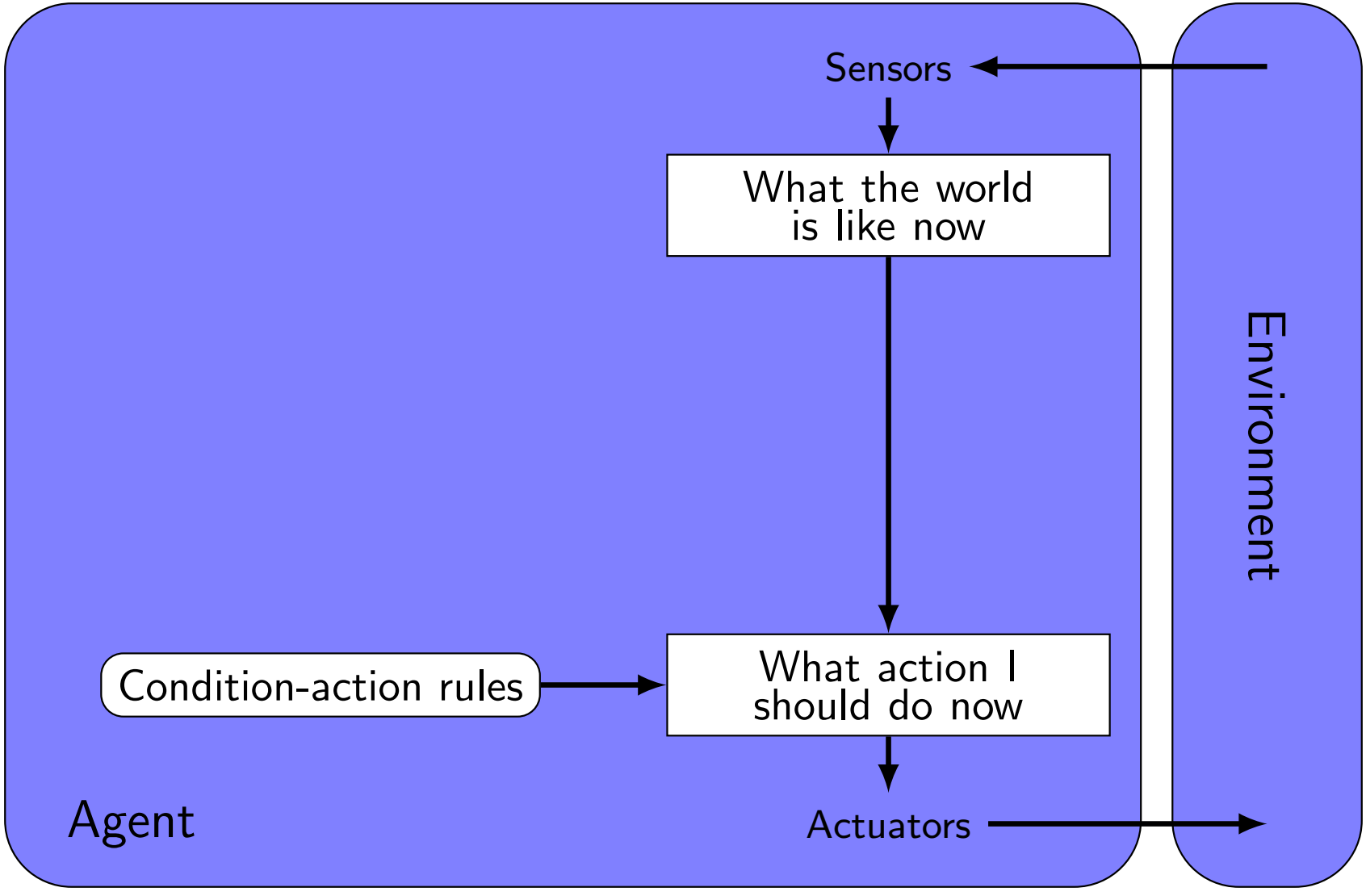
Simple Reflex Agents

- Simple reflex agents: select actions to execute based upon the current percept.
- Do not take the percept history into account.
- Implemented using condition-action rules such as:

  If car in front is breaking, then initiate breaking

- Such agents are simple to implement, but of very limited intelligence.

Simple Reflex Agent



Simple Reflex Agents: limitations

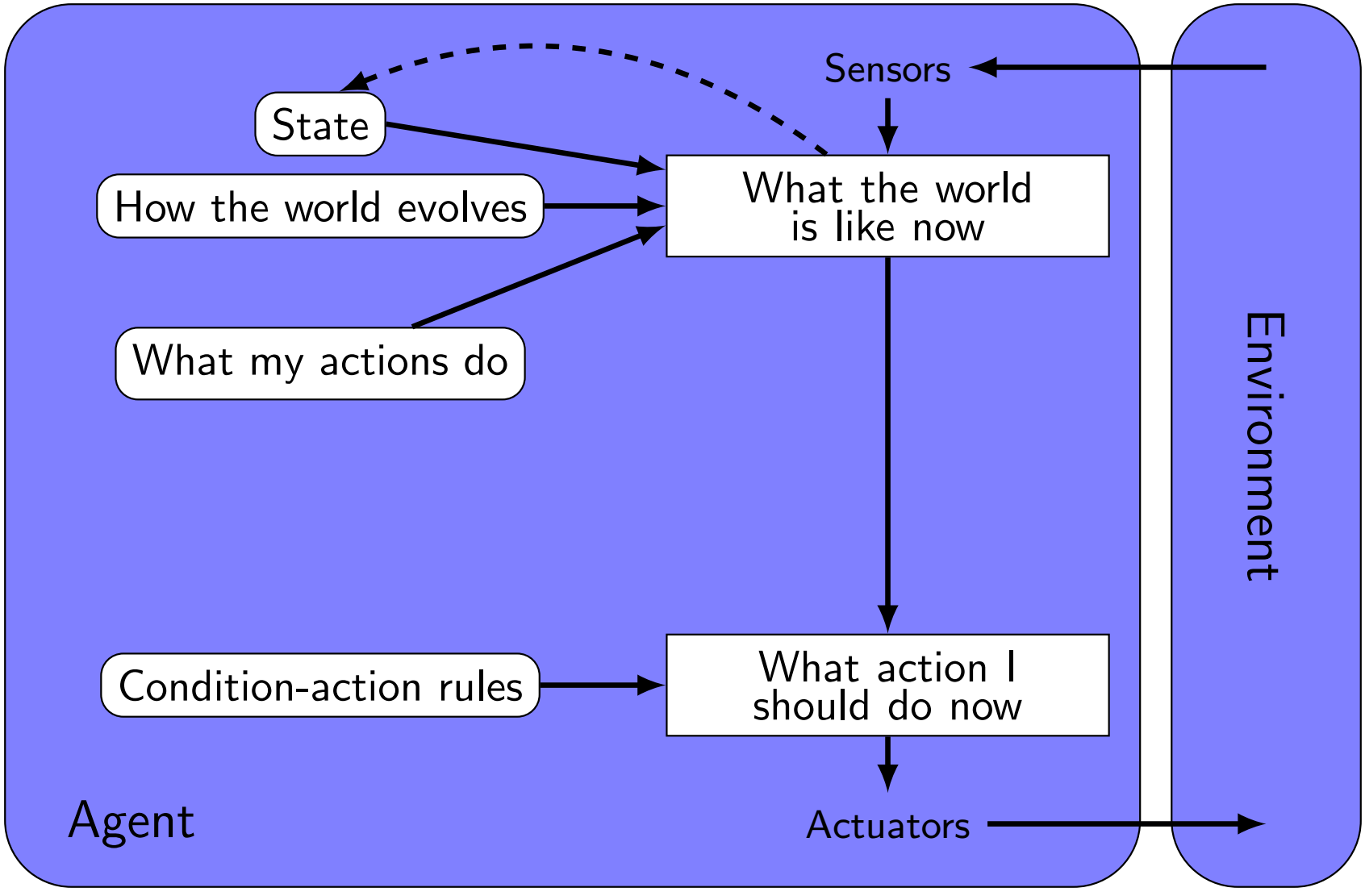
Simple reflex agents have no history and do not know how their actions influence the world in different situations. They cannot

- play football: to decide the next move, a striker has to remember that a second ago there was a defender 4 meters away.
- play poker: to decide the next move, the player has to remember the history of the current game.
- drive a car: to decide whether to overtake a car, the driver has to remember whether a couple of seconds ago there was another car overtaking the driver's car.
- and so on

Model-based Reflex Agents

- Model-based reflex agents: maintain an internal state (memory, etc) that depends upon the percept history.
- Current percept combined with previous internal state to update description of current state.
- Helps to deal with partial observability.
- Model-based agents have knowledge (a model) about 'how the world works': how does the world change independently from the agent's actions. How does the world change due to the agent's actions.

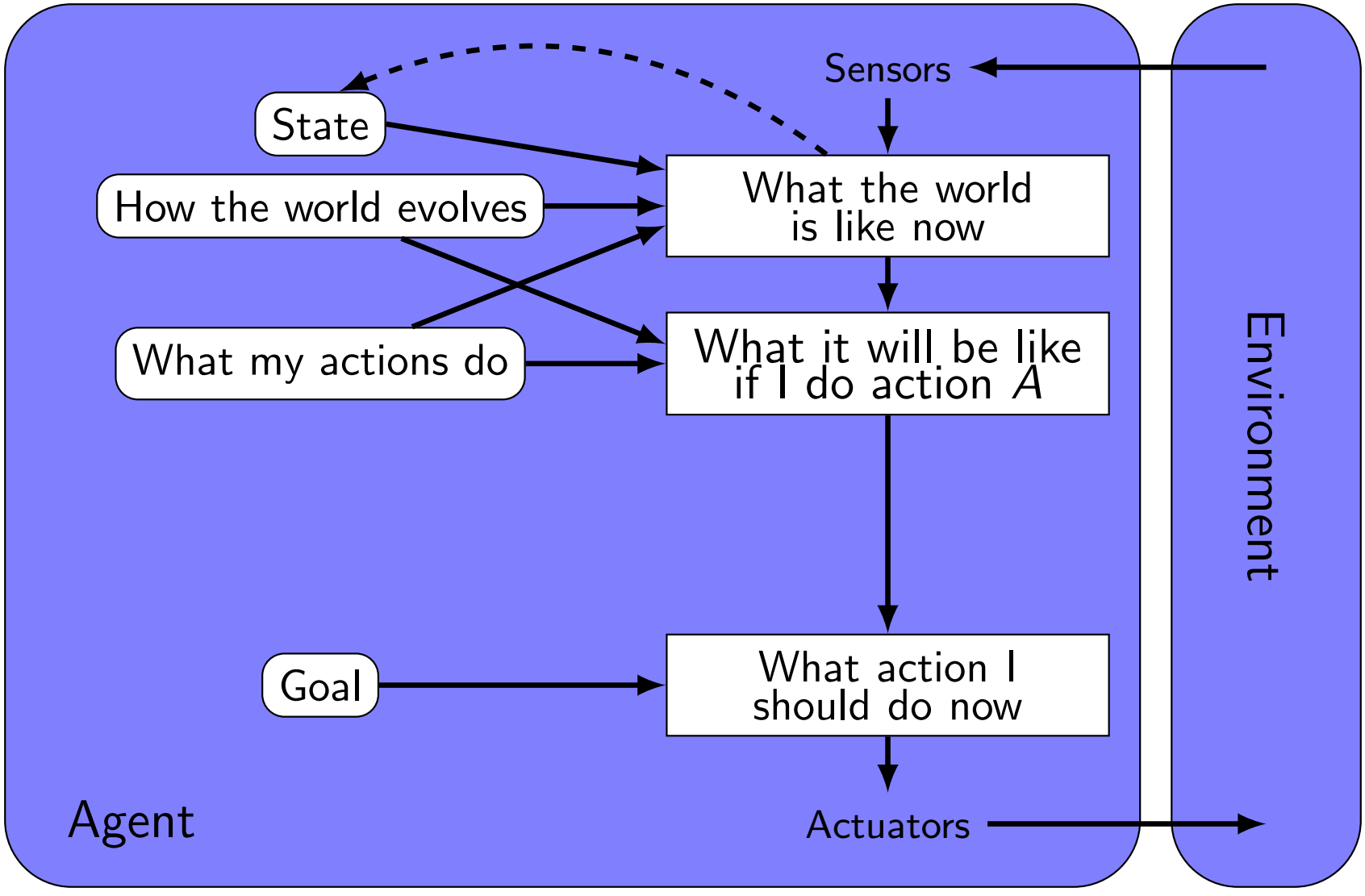
Model-Based Reflex Agent



Goal-based agents

- Goal-based agents: select appropriate actions to achieve desirable states of the environment: goals.
- Knowledge of the current state does not automatically mean that the agent knows what to do.
- Decision making may become complicated when dealing with long sequences of actions to achieve a goal. Search and planning may be required.

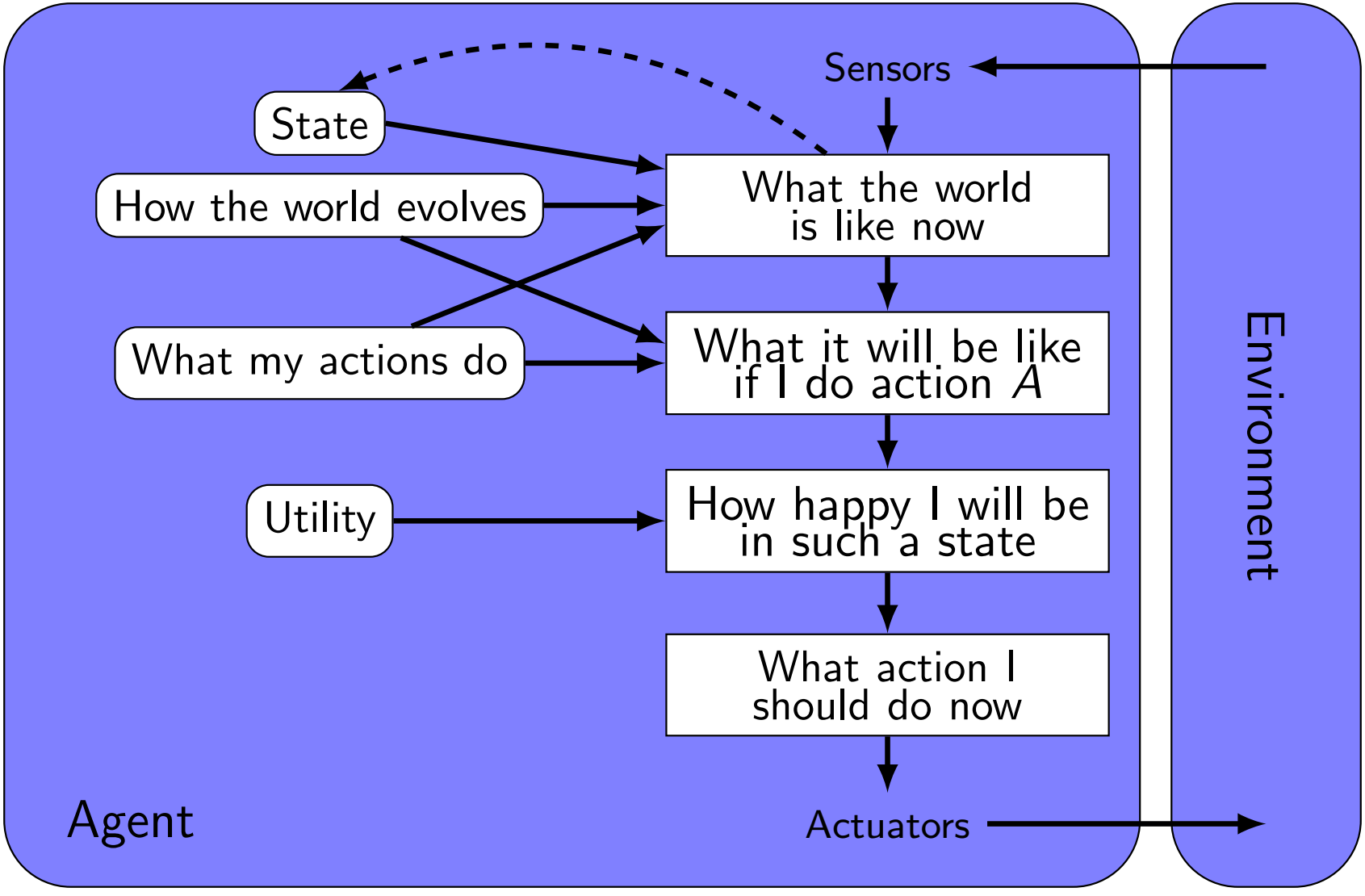
Goal-Based Agent



Utility-based agents

- Utility-based agents: make use of a utility function to compare the desirability of different states that result from actions.
- Many actions may satisfy a goal, but which is the most desirable?
- Utility function maps a state, or sequence of states, onto a number to give the degree of usefulness of the state to the agent.
- Agent tries to maximise the value of its utility function.
- Tradeoffs may need to be made between conflicting goals.

Utility-Based Agent



Learning agents

How does the agent achieve its goal or maximize the value of its utility function? Learning is a fundamental idea. Consider the following learning model (next slide):

- Learning agents: improve the way in which actions are chosen depending on previous experience.
- The performance element denotes the action selection procedure from previous agent models. This is now informed by a learning element.
- The performance standard defines what is a good outcome of an action. It gives the necessary feedback for learning.
- The problem generator supplies problems that should be considered to learn something.

Learning Agent