ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Intelligenza Artificiale

Intelligent Agents

Part I

Ivan Heibi
Dipartimento di Filologia Classica e Italianistica (FICLIT)
Ivan.heibi2@unibo.it

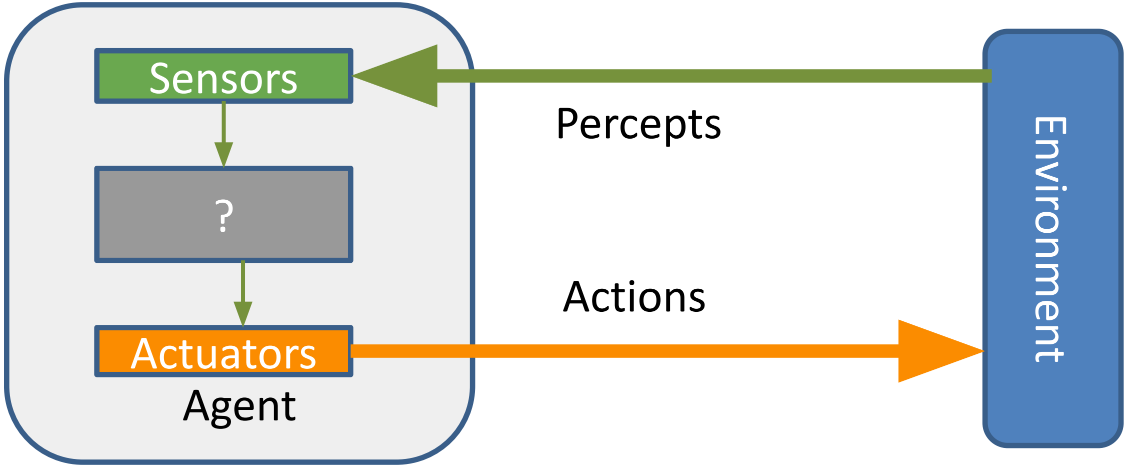
What is an agent?

An **agent** is anything that can be viewed as:

- **perceiving** its environment through sensors
- **acting** upon that environment through actuators

For instance: Human / Robot / Software

An agent's choice of action at any given instant can depend on the entire percept sequence observed to date.



Agent Function

Mathematically speaking, an agent behaviour can be described by a **function** that maps any given percept sequence to an action.

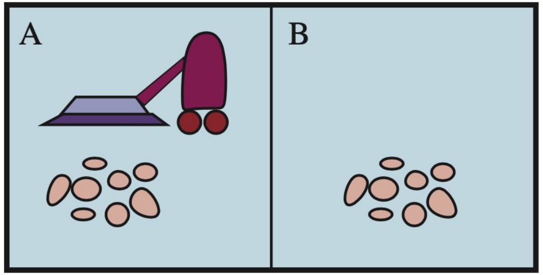
We can imagine this function as a table with (possibly infinite) number of rows

Example: Vacuum agent

Environment: *Square A*, *Square B*

Perceptions: *Square Dirty*, *Square Clean*

Actions: Move left, Move right, Draw up the dirt



Percept sequence	Action
[A, Clean]	Right
[A, Dirty]	Draw up
[B, Clean]	Left
[B, Dirty]	Draw up
[A, Clean][A, Clean]	Right
[A, Clean][A, Dirty]	Draw up
....	
[A, Clean][A, Clean] [A, Clean]	Right
[A, Clean][A, Clean] [A, Dirty]	Draw up
...	

Rational agent

A **rational agent** is one that does the right thing (every entry of the table is filled out correctly).

The action sequence of the agent causes the environment to go through a sequence of states. If the sequence (of environment states) is desirable, then the agent has performed well.

The notion of **desirability** is captured by a **performance measure** that evaluates any given sequence of **environment states** (not agent states!).

There is no fixed performance measure for all tasks and agents, typically a designer will devise one appropriate to the circumstances.

The performance measures are defined according to what the designer actually wants from the environment, rather than how one thinks the agents should behave.

Rationality

What is rational at any given time depends on four things:

- The **performance measure** that defines the criterion of success
- The agent’s **prior knowledge** of the environment
- The **actions** that the agent can perform
- The agent’s **percept sequence** to date

For each possible 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 agent – vacuum example

- The **environment** is known a priori but the dirt distribution and the initial location of the agent are not, also, clean squares stay clean and sucking cleans the current square.
- The agent correctly **perceives** its location and whether that location contains dirt.
- The only available **actions** are Right, Left, and Draw up. The Right and Left actions move the agent one square except when this would take the agent outside the environment, in which case the agent remains where it is.
- The **performance measure** awards one point for each clean square at each time step.

Under these circumstances the **agent is indeed rational**; its expected performance is at least as good as any other agent's.

Omniscience, learning and autonomy

We distinguish **rationality** and **omniscience**.

An omniscient agent knows the actual outcome of its actions and can act accordingly, but omniscience is impossible in reality.

Rationality maximizes expected performance.

Doing an action in order to modify future percepts or explore the environment is called **information gathering**.

The rational agent is also able to **learn** from what it perceives and act autonomously.

PEAS description

The **PEAS description** is an acronym for the elements needed for specifying the problem:

- Performance measure
  It defines what the agent aspires.
- Environment
  It is where the agent operates.
- Actuators
  The “devices” that allow the agent to operate
- Sensors
  The “devices” that allow the agent to percept

Agent type	Performance measure	Environment	Actuators	Sensors
Automated Taxi Driver	Safe, fast, legal, comfortable trip, maximize profits	Roads, pedestrian, customers	Steering, accelerator, brake, signal, horn, display	Cameras, sonar, speedometer, GPS

Properties of the task environment: Fully observable vs. partially observable

If an agent's sensors give it access to the complete state of the environment at each point in time, then we say that the task environment is **fully observable**.

A task environment is effectively fully observable if the sensors detect all aspects that are relevant to the choice of action; relevance, in turn, depends on performance measure.

Fully observable environments are convenient because the agent need not maintain any internal state to keep track of the world.

If the agent has no sensors at all the environment is **unobservable**.

Properties of the task environment: Single agent vs. multiagent

This distinction depends on the **number of agents** acting within the environment.

An agent solving crossword puzzle operates in a **single-agent environment,** whereas an agent playing chess operates in a **two-agent (multi agent) environment**.

The environment may be **competitive** (e.g. chess) or **cooperative** (e.g. taxi-driver, since other agents intend to avoid collisions).

Properties of the task environment: Deterministic vs. non deterministic

If the next state of the environment is completely determined by the current state and the action executed by the agent, then we say the environment is deterministic; otherwise, stochastic.

If the environment is partially observable, then it could appear to be stochastic.

We say that an environment is uncertain if it is not fully observable or not deterministic.

Most real environments are so complex that is impossible to keep track of all the unobserved aspects; and, for practical purposes, the must be treated as stochastic.

An **environment is stochastic** when possibilities are attached to its states (e.g., 25% chance of raining)

Properties of the task environment: Episodic vs. sequential

In an episodic task environment, the agent's experience is divided into atomic episodes. In each episodes, the agent receives a percept and then performs a single action.

The next episode does not depend on the actions taken in previous episodes.

(e.g. question answering, classification)

In sequential environments, the current decision could affect all future decisions (e.g. chess, driving)

Properties of the task environment: Static vs. Dynamic

If the environment can change while an agent is deliberating then we say that the environment is **dynamic** for that agent; otherwise, it is **static**.

If the environment itself does not change with the passage of time but the **agent’s performance score does**, the environment is **semi-dynamic**. eg. Chess with clock

Properties of the task environment: Discrete vs. Continuous

The **discrete vs continuous** distinction applies to the state of the environment, to the way the time is handled, and to the precepts and actions of the agent.

For example, chess environment has a finite number of distinct states (excluding the clock).

Taxi driving is a continuous-state and continuous-time.

Properties of the task environment: Known vs. Unknown

This distinction refers not to the environment itself but to the agent's state of knowledge about the "law of physics" of the environment.

In a known environment, the outcomes for all the action are given.

If the environment is unknown, the agent will have to learn how it works in order to make good decisions.

Known environments can be partially observable (e.g. solitaire card games, rules are known but the cards not).

Examples

	Chess w clock	***Object recognition***	***Taxi***
Fully / Partially Observable	Fully	Fully	Partially
Single / Multi Agent	Multi	Single	Multi
Deterministic / Stochastic	Deterministic	Deterministic	Stochastic
Episodic / Sequential	Sequential	Episodic	Sequential
Static / Dynamic	Semi-Dynamic	Static	Dynamic
Discrete / Continuous	Discrete	Continuous	Continuous

The real world is partially observable, multiagent, stochastic, sequential, dynamic, continuous, unknown

Exercise – What about Poker?

	Poker
Fully / Partially Observable	
Single / Multi Agent	
Deterministic / Stochastic	
Episodic / Sequential	
Static / Dynamic	
Discrete / Continuous	

Exercise

	Poker
Fully / Partially Observable	Partially
Single / Multi Agent	
Deterministic / Stochastic	
Episodic / Sequential	
Static / Dynamic	
Discrete / Continuous	

Exercise

	Poker
Fully / Partially Observable	Partially
Single / Multi Agent	Multi
Deterministic / Stochastic	
Episodic / Sequential	
Static / Dynamic	
Discrete / Continuous	

Exercise

	Poker
Fully / Partially Observable	Partially
Single / Multi Agent	Multi
Deterministic / Stochastic	Stochastic
Episodic / Sequential	
Static / Dynamic	
Discrete / Continuous	

Exercise

	Poker
Fully / Partially Observable	Partially
Single / Multi Agent	Multi
Deterministic / Stochastic	Stochastic
Episodic / Sequential	Sequential
Static / Dynamic	
Discrete / Continuous	

Exercise

	Poker
Fully / Partially Observable	Partially
Single / Multi Agent	Multi
Deterministic / Stochastic	Stochastic
Episodic / Sequential	Sequential
Static / Dynamic	Static
Discrete / Continuous	

Exercise

	Poker
Fully / Partially Observable	Partially
Single / Multi Agent	Multi
Deterministic / Stochastic	Stochastic
Episodic / Sequential	Sequential
Static / Dynamic	Static
Discrete / Continuous	Discrete