

---

# AI1001 - Assignment 6

---

Shivram S  
ai24btech11031@iith.ac.in

## 1 Agents and Environments

An **agent** is a system that perceives its environment through sensors and acts upon the environment using actuators. What is perceived by the sensors is called **percept** and the complete history of percepts is called the agent's **percept sequence**. An agent's behaviour is described by an **agent function** that maps percepts to actions. This is implemented by an **agent program**.

A rational agent is one that does the **right thing**. There are different notions of what the "right thing", but AI generally uses the notion of **consequentialism**: each action sequence of the agent is assigned some **desirability** which is used as a performance measure.

It can be quite hard to formulate a performance measure correctly. For a vacuum cleaner, we might measure performance by the amount of dirt cleaned, but an agent might exploit it by creating more dust to maximize its performance metric.

Rationality depends on four factors - the performance measure, the agent's prior knowledge, the allowed actions, and the agent's percept sequence. Rationality maximizes an agent's expected performance, but the actual outcome of actions can not be predicted, i.e., rationality is not omniscience. In order to reduce risk of accident, an agent should gather information through exploration, and learn as much as possible from what it perceives.

If an agent relies on the designer's prior knowledge rather than its own percepts and learning processes, we say that the agent lacks autonomy. A rational agent should be autonomous, but does not require complete autonomy from the start. It may be provided with some initial knowledge as well as an ability to learn. After sufficient experience of its environment, the behaviour of a rational agent can become effectively independent of its prior knowledge.

## 2 Nature of Environments

When designing an agent, we must consider four attributes - the performance measure, the environment, the actuators and the sensors. For example, a self-driving car might be expected to get to the correct destination, minimize traffic violations, minimize trip time and cost, and maximise safety and passenger comfort. Some of these goals conflict, and the agent will have to make tradeoffs.

The environment for a self-driving car may vary from small lanes to freeways, and the roads may contain obstacles. The actuators might be an accelerator, a steering, breaking, and a screen to communicate with passengers. The sensors for the car might include video cameras, lidar and ultrasound sensors.

Environments may be classified into several categories:

- In a **fully observable** environment, the agent has access to the complete state of the environment, but not in a **partially observable** environment. If an agent has no sensor, then the environment is unobservable.
- If one agent maximizing its performance measure leads to another agent minimizing its performance measure, then the environment is said to be **competitive**. If some action collectively maximizes the performance measure then the environment may be **co-operative**.

- If the agent can determine the next state of the environment from the current state and its action, then the environment is said to be **deterministic** otherwise it is said to be **non-deterministic**. A non-deterministic environment is said to be **stochastic** if the probabilities for each outcome can be quantified.
- Environments with one agent are called **single-agent** and those with more than one agent are said to be called **multi-agent** environments.
- An environment which has a finite number of discrete states is **discrete** and an environment whose state sweeps over values smoothly over time is said to be **continuous**.
- If the “laws of physics” of the environment are known, then it is said to be **known** otherwise it is said to be **unknown**.
- If the environment can change while the agent decides on an action, then it is said to be **dynamic** otherwise it is said to be **static**.