

University of Warwick
Department of Computer Science

CS255

Artificial Intelligence



Cem Yilmaz
January 3, 2023

Contents

1	Agents	2
1.1	Inputs to an agent	2
1.2	Dimensions of Complexity	2
1.2.1	Modularity	2
1.2.2	Planning Horizon	3
1.2.3	Representation	3
1.2.4	Learning from experience	3
1.2.5	Uncertainty	3
1.2.6	Sensing Uncertainty	3
1.2.7	Effect Uncertainty	3
1.2.8	Preferences	3
1.2.9	Number of Agents	3
1.2.10	Interaction	3
2	Representation	3
2.1	Defining a solution	4
2.2	Decisions and outcome	4
2.3	Physical Symbol System Hypothesis	4
2.4	Knowledge and Symbol Levels	4
2.5	Agent System Architecture	5
2.6	Controller	5
2.7	Belief States	5
2.8	Agent Types	7

1 Agents

Definition 1.1. Agent

An agent is an entity that perceives and acts. An agent can be viewed as a function from percept histories to actions, where

$$f : P \rightarrow A$$

Agents typically required at exhibit autonomy. For any given class of environments and tasks we seek the agent(s) with the best performance. Perfect rationality usually impractical given computational restraints, therefore, it is the best that we design the best agent for a given set of resources.

Definition 1.2. Artificial Intelligence

Artificial intelligence is the synthesis and analysis of computational agents that act intelligently. An agent acts "intelligently" if:

- its actions are appropriate for its goals and circumstances
- it is flexible to changing environments and goals
- it learns from experience
- it makes appropriate choices given its perceptual and computational limitations

However, note that artificial intelligence has different goals:

Scientific Goal: to understand the principles that make intelligence behaviour possible in natural or artificial systems. That is, analyse natural and artificial agents, formulate and test hypotheses about what it takes to construct intelligent agents and finally design, build and experiment with computational systems that perform tasks that require intelligence

Engineering Goal: design useful, intelligence artefacts

For this module, the main goal is engineering goal.

1.1 Inputs to an agent

There are several inputs to an agent, that list as following:

1. Abilities - the set of possible actions it can perform
2. Goals/Preferences - what it wants, its desires, its values ...
3. Prior Knowledge - what it comes into being knowing, what it doesn't get from experience, ...
4. History of stimuli - what it has received in the past. However, current stimuli is what it perceives from environment now.

However, note that rational \neq omniscient. An omniscient agent would know the actual outcome of its actions - agents are rarely omniscient since unexpected situations occur in dynamic environments. A rational agent needs only to do its best given the current percepts. Similarly, rational \neq clairvoyant. An agent is not expected to foresee changes in its environment. Lastly, rational \neq successful. Rational action is defined in terms of expected value, rather than actual value; a failed action can still be rational.

1.2 Dimensions of Complexity

1.2.1 Modularity

Agent structure has one level of abstraction: flat. Agent structure has interesting modules that can be understood separately: modular. Agent structures has modules that are decomposed into modules: hierarchical.

1.2.2 Planning Horizon

Planning horizon is how far the agents look into the future when deciding what to do. A static is a non-planning AI. Finite stage are agent reasons about a fixed number of time steps. Indefinite stage are agent reasons about a finite, but not predetermined, number of steps. Infinite stage is the case that agent plans for going on forever.

1.2.3 Representation

Much of modern AI is about finding compact representations and exploiting the compactness for computational gains. An agent can reason in terms of explicit states, where a state is one way the world could be, features and propositions where states can be described using features, and finally, individuals and relations where there is a feature for each relationship on each tuple and individual.

1.2.4 Learning from experience

Whether the model is fully specified a priori: knowledge is given or knowledge is learned from data or past experience.

1.2.5 Uncertainty

There are two dimensions for uncertainty - sensing and effect. In each dimension an agent can have no uncertainty, disjunctive uncertainty or probabilistic uncertainty. Usually we would choose probability as agents can still act even if they are uncertain. Predictions are needed to decide what to do. Acting is gambling - agents who do not use probability will lose to those that do, and probabilities can be learned from data and prior knowledge.

1.2.6 Sensing Uncertainty

1.2.7 Effect Uncertainty

A deterministic environment is the resulting state is determined from the action and the state. Stochastic is there is uncertainty about the resulting state.

1.2.8 Preferences

An agent has achievement goal to achieve - this can a complex logical formula. Complex preferences may involve trade-offs between various desiderata, perhaps at different times from ordinal and cardinal. Ordinal is where only the order matters, and cardinal is where the values also matter.

1.2.9 Number of Agents

Are there multiple reasoning agents that need to be taken into account? Single agent reasoning - any other agents are part of the environment. Or multiple agents, an agent reasons strategically about the reasoning of other agents. Agents can have their own goals - cooperative, competitive, or goals can be independent of each other

1.2.10 Interaction

When does the agent reason to determine what it do? Reason offline - before acting or reason online - while interacting with the environment

2 Representation

We want a representation to be

- rich enough to express the knowledge needed to solve the problem
- as close to the problem as possible - compact, natural and maintainable
- amenable to efficient computation i.e., able to express features of the problem that can be exploited for computational gain or able to trade off accuracy and computation time and/or space.
- Able to be acquired from people, data and past experiences.

2.1 Defining a solution

Given an informal description of a problem what is a solution? Typically much is left unspecified, but the unspecified parts cannot be filled arbitrarily. Much work in AI is motivated by common-sense reasoning - the computer needs to make common-sense conclusions about the unstated assumptions. What matters too, is the quality of solutions -

- An optimal solution - is a best solution according some measure of quality
- Satisfying solution - is one that is enough, according to some description of which solutions are adequate
- Approximately optimal solution - is one whose measure of quality is closer to the best theoretically possible
- Probable solution - one that is likely to be a solution

2.2 Decisions and outcome

Good decisions can have bad outcomes and equally bad decisions can have good outcomes. Information can be valuable because it leads to better decisions: can sometimes quantify the value of information. We can often trade off computation time and solution quality. An anytime algorithm can provide a solution at any time; given more it can produce better solutions. An agent is not just concerned about finding the right answer, but about acquiring the appropriate information, and computing it in a timely manner.

2.3 Physical Symbol System Hypothesis

A symbol is a meaningful physical pattern that can be manipulated. A symbol system creates, copies, modifies and destroys symbols. A physical symbol system hypothesis has the necessary and sufficient means for general intelligent action.

- The knowledge level is in terms of what an agent knows and what its goals are
- The symbol level is a level of description of an agent in terms of what reasoning it is doing

to map from problem to representation, we need to think about level of abstraction the problem represents, what individuals and relations in the world to present, how can an agent represent the knowledge to ensure that the representation is natural, modular and maintainable, how can an agent acquire the information from data, sensing experience or other agents

2.4 Knowledge and Symbol Levels

Two levels of abstraction seem to be common among biological and computational entities: the knowledge level is in terms of what an agent knows and what its goals are. The symbol level is a level of description of an agent in terms of what reasoning it is doing.

2.5 Agent System Architecture

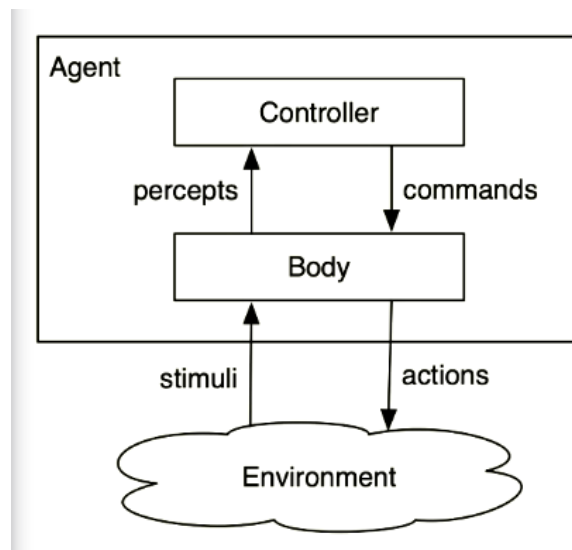


Figure 1: Agent System Architecture

An agent interacts with the environment through its body. The body is made up of sensors that interpret stimuli and actuators that carry out actions. The controller receives percepts from the body and the controller sends commands to the body. The body can also have reactions that are not controlled.

2.6 Controller

A controller is the brains of the agent. Agents are situated in time, they receive sensory data in time and do actions in time. Controllers have limited memory and limited computational capabilities. The controller specifies the command at every time. The command at any time can depend on the current and previous concepts. For agent of functions, let T be the set of time points. A percept trace is a sequence of all past, present and future percepts received the controller. A command trace is similar but with commands except of percepts. A transduction is a function from perept traces into command traces. A transduction is causal if the command trace up up to time t depends only on percepts up to t . A controller is an implementation of the causal transduction. An agent's history at time t is a sequence of past and present percepts and past commands. A causal transduction specifies a function from an agent's history at time t into its action at time t .

2.7 Belief States

An agent does not have access to its entire history, it only has access to what it has remembered. The memory or belief state of an agent at time t encodes all of the agent's that it has access to. The belief state of an agent encapsulates the information about its past it can use for current and future actions. At every time a controller has to decide what it should do, what should it remember and how should it update its memory. This is a function of its percepts and its memory. For discrete time, a controller implements

- belief state function - remember(belief state, percept). Returns the next belief state
- command function - do(memory, percept). Returns the command for an agent

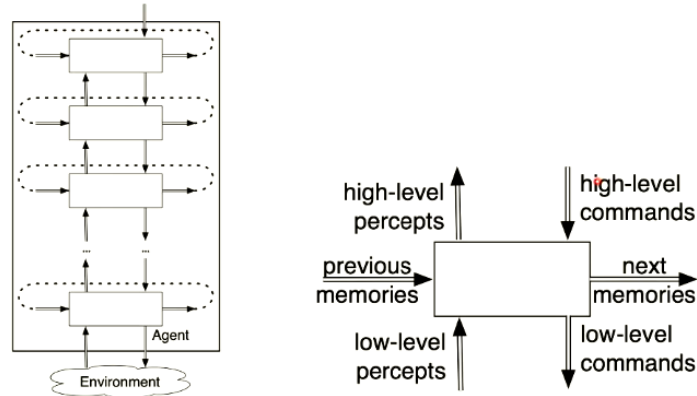


Figure 2: Hierarchical Robotic System Architecture

The layer above generally begins abstract e.g., travelling to a destination using a car. However, by the time the architecture arrives to the bottom layer. For example, pedestrian avoidance, pedestrian detection, etc. The commands may also change depending on their level of abstraction. The functions implemented a layer are described as

- `remember(memory, percept, command)` - we need to take command that comes down from higher level
- `do(memory, percept, command)` - we may need to modify the command depending on memory and perception that is received
- `higher_percept(memory, percept, command)` - we need to decide which perceptions we need to pass back up depending on our memory and command (which may have changed from the bottom layer and now we are sending it back to the layer above)

Example 2.1. Delivery Robot

The robot has three actions: go forward, right or left.

It can be given a plan consisting of a sequence of named locations for the robot to go to in turn. The robot must avoid obstacles. It has a single whisker sensor pointing forward and to the right. The robot can detect if the whisker hits an object and the robot knows its current location. The obstacles and locations can be moved dynamically; obstacles and new locations can also be created dynamically

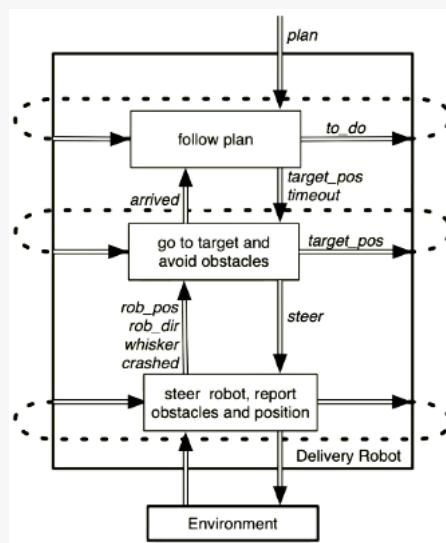


Figure 3: Architecture of the example

To decide what should be in an agent's belief state, we need to first determine what kind of an agent it is. We know that a purely reactive agent does not have a belief state, and a dead reckoning agent does not perceive the state.

Neither work well in complicated domains, but were proven to be useful in simple situations. It is often useful for the agent's belief state to be a model of the world.

Definition 2.1. Ideal Rational Agent

Does whatever action is expected to maximise performance measure on basis of percept sequence and built-in knowledge

Therefore, in theory ideal mapping of percept sequences to actions correspond to an ideal agent. The simplest approach is a lookup table. However, this fails because the size is too big, the time to build is too high, and has no autonomy. Nevertheless, the table suggests a notional ideal mapping. One agent function (or a small equivalence class) is rational. Our aim is to find a way to implement the rational agent function. We need to implement rational agent function concisely. The implementation must be relatively efficient, exhibit autonomy if required and get as close as possible to the ideal mapping.

2.8 Agent Types

Definition 2.2. Simple Reflex Agents

These are condition action rules. More precisely, this is the if then statement.

Definition 2.3. Reflex agents with state

This agent retains the knowledge about the world e.g., the output of the action.

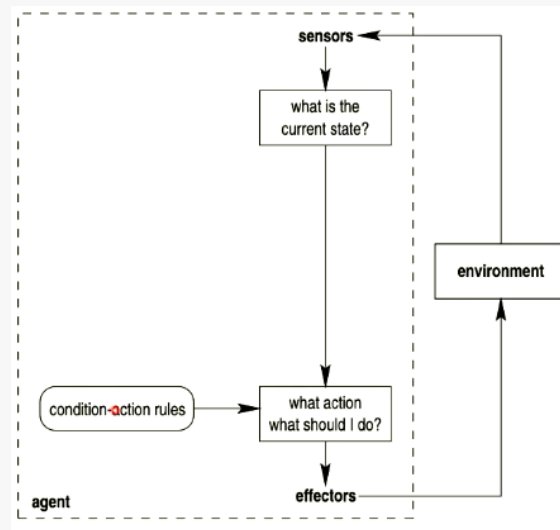


Figure 4: Reflex Agent

Note that sensors are the only thing that show the agent's current state which are determined by the sensors. We can modify the diagram above to include a state that can help it enhance its decisions depending on its percept state.

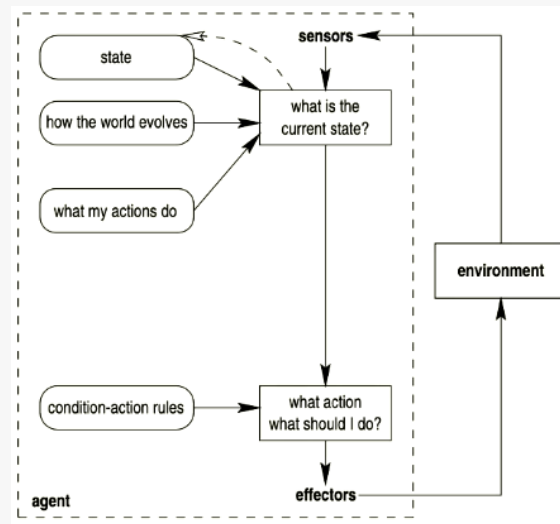


Figure 5: State Reflex Agent

Definition 2.4. Goal-based agents

This has representation of what states are desirable

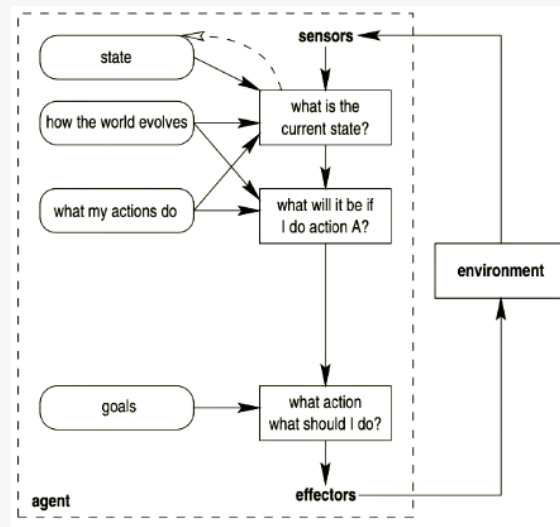


Figure 6: Goal-based agent

The main change of this agent is the question "what will it be if I do action A"? The complexity of this agent increases substantially.

Definition 2.5. Utility-based agents

Ability to discern some useful measure e.g., cost, quality between possible means of achieving some state.

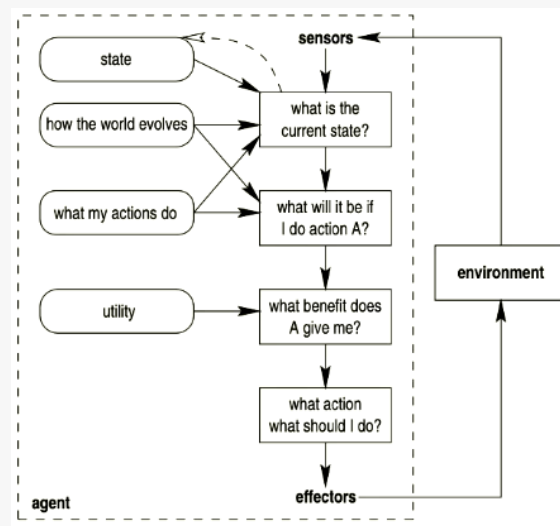


Figure 7: Utility-based agent

We focus on utility and the benefits separately rather than searching for a win state when making our decision. The utility function also enables a choice about which goals to achieve - we select the one with the highest utility. If achievement is uncertain we can measure the importance of goal against likelihood achievement.

Definition 2.6. Learning agents

Able to modify their behaviour based on their performance, or in the light of new information

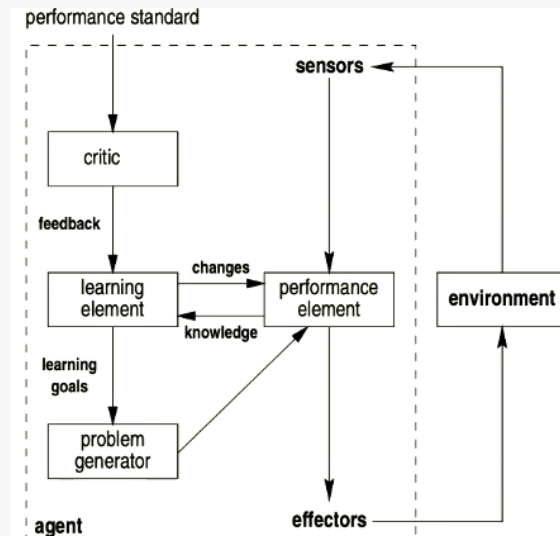


Figure 8: Learning Agent

Learning agent gradually modifies itself. Typically we have objectives that generates problems that feeds into the performance element. We then find out whether our problem solving went well or not. It is then fed back to learning element and looped. The critic and the performance standard tells how well the AI did after using the problem.