

# AI Week 1 notes

Problem classification based on the assumption

1 Fully Observable

2 Partially Observable

1 Fully observable - An environment is fully observable if it has given you or if you know anything that is needed/required to make a decision.

- Any problem can be solved at some level of abstraction
- Deterministic vs Stochastic

Stochastic meaning the probability distribution function for any action to take place or say the chance of an action to happen.

- Episodic vs Sequential
- Static vs Dynamic

Static means the environment never changes however dynamic means the environment keeps on changing.

Semi dynamic- performance changes based on the particular action.

- Discrete vs Continuous  
 $\{1,2,3\}$  is the example of discrete  
 $[1,3]$  is the example of continuous

- Single Agent vs Multiple Agent-  
Like your computer program(single)  
Like you (can do multiple tasks)

Multiple agents are further classified into two categories -

Cooperative agent

Competitive agent(on the basis of agent)

Competitive agent- A system is competitive , when both the agents are having different goals and different performance measures.

Cooperative agent- A system is cooperative , when both the agents are having exactly the same objectives and performance measures too.

- Multi-agent systems(on the basis of communication)

- 1 Direct Communication (when an agent is connected with wifi , LAN , BT etc)
- 2 No direct communication( when an agent can see, hear each other but can't communicate directly)

Randomization- An action needs to have random elements

The screenshot shows a YouTube video player with a handwritten diagram titled "AI AGENT". The diagram illustrates the interaction between an agent and its environment. On the left, a circle labeled "AGENT" has an arrow labeled "Sensors" pointing to a circle on the right labeled "ENVIRONMENT (SIMULATED)". An arrow labeled "Actuators" points from the agent circle back to the environment circle. To the left of the agent circle, the text "ENVIRONMENT" is written vertically, followed by "N(SENSOR)" and "S ACT". To the right of the environment circle, the text "while (true)" is written. Below the environment circle, there is a red handwritten note: "class ENV { true T(S) Ready-...". The video player interface includes a search bar, a list of related videos on the right, and a video player control bar at the bottom.

This screenshot is similar to the one above, but with additional handwritten code snippets. The diagram and red note "class ENV { true T(S) Ready-..." are present. To the right of the environment circle, there is a code block:
 

```
while (true)
{
    S ← GETREAD OF
    UR. SIMULATOR
    A ← AG. AGFUN(S)
    IMPLEMENT A
}
```

 Below the environment circle, there is another red handwritten note: "AGENT - AG". The video player interface and related videos on the right are also visible.

YouTube video player showing a lecture titled "AI L1 P5 Introduction and Agents: Reflexive Agents" by rahul kala. The video displays a hand-drawn diagram of a reflexive agent's internal structure.

**Diagram: REFLEXIVE AGENT**

- AGENT** (Overall structure)
- SENSOR** (Input to the agent)
- SIGNAL PROCESSING** (Understand of what the sensors say?)
- RULES** (Connected to Rule Engine)
- RULE ENGINE** (Connected to Rules and Act)
- act** (Output of the agent)
- ACTUATOR** (Output of the agent)

The video player interface includes a search bar, a list of related videos, and a video player with a progress bar.

YouTube video player showing a lecture titled "AI L1 P6 Introduction and Agents: Model-based and Goal based Agents" by rahul kala. The video displays a hand-drawn diagram of a model-based reflexive agent's internal structure.

**Diagram: MODEL-BASED REFLEXIVE AGENT**

- AGENT** (Overall structure)
- SENSOR** (Input to the agent)
- SIGNAL PROCESSING** (Understand of what the sensors say?)
- RULES** (Connected to Rule Engine)
- RULE ENGINE** (Connected to Rules and Act)
- act** (Output of the agent)
- ACTUATOR** (Output of the agent)

The video player interface includes a search bar, a list of related videos, and a video player with a progress bar.

Exploration - It means gathering of information ( you go to washroom during exams just to check if your friend is there)

YouTube video player interface showing a lecture titled "AI L1 P6 Introduction and Agents: Model-based and Goal based Agents" by rahul kala. The video content displays a hand-drawn diagram of an AGENT. The diagram includes components: SENSORS, AGENT (containing MODEL, PROC, GOAL, PLAN, RULE, and RECEPTION), and ACT. Handwritten notes explain the process: "1st iteration" and "2nd iteration onwards" are marked. A note states: "RE-PLAN IF GOAL CHANGES STATE IS DIFF FROM EXPECTED". A small video inset shows the presenter, rahul kala.

AI L1 P6 Introduction and Agents: Model-based and Goal based Agents  
134 views • Aug 2, 2020

rahul kala

Artificial Intelligence Lectures  
rahul kala - 6 / 21

- AI L1 P6 Introduction and Agents: Model-based and Goal based Agents
- AI L1 P7 Introduction and Agents: Utility and Learning Based Agents
- AI L2 P1 Searching: When to use a search?
- AI L2 P2 Searching: States and Abstraction
- AI L2 P3 Searching: Actions & More Examples
- AI L2 P4 Searching: Breadth First Search (BFS)

Data Analysis with Python: Part 1 of 5 (Live Course)  
freeCodeCamp.org  
2.4K watching  
LIVE NOW

AI L2 P2 Searching: States and Abstraction

YouTube video player interface showing a lecture titled "AI L1 P6 Introduction and Agents: Model-based and Goal based Agents" by rahul kala. The video content displays a hand-drawn diagram of a GOAL-BASED AGENT. The diagram includes components: SENSORS, AGENT (containing MODEL, PROC, GOAL, PLAN, RULE, and RECEPTION), and ACT. Handwritten notes explain the process: "1st iteration" and "2nd iteration onwards" are marked. A note states: "RE-PLAN IF GOAL CHANGES STATE IS DIFF FROM EXPECTED". A small video inset shows the presenter, rahul kala.

GOAL-BASED AGENT

AI L1 P6 Introduction and Agents: Model-based and Goal based Agents  
134 views • Aug 2, 2020

rahul kala

Artificial Intelligence Lectures  
rahul kala - 6 / 21

- AI L1 P6 Introduction and Agents: Model-based and Goal based Agents
- AI L1 P7 Introduction and Agents: Utility and Learning Based Agents
- AI L2 P1 Searching: When to use a search?
- AI L2 P2 Searching: States and Abstraction
- AI L2 P3 Searching: Actions & More Examples
- AI L2 P4 Searching: Breadth First Search (BFS)

Data Analysis with Python: Part 1 of 5 (Live Course)  
freeCodeCamp.org  
2.4K watching  
LIVE NOW

AI L2 P2 Searching: States and Abstraction

YouTube video player showing a lecture titled "GOAL-BASED AGENT" by rahul kala. The video is at 16:22. The main content is a hand-drawn diagram of a goal-based agent. The diagram shows a large circle labeled "AGENT" containing a smaller circle labeled "MODEL". Inside the "MODEL" circle, there are boxes for "GOAL", "PLAN", "PROC", and "RULE". Arrows indicate a flow from "GOAL" to "PLAN" to "PROC" to "RULE". Outside the "AGENT" circle, there are labels "SENSORS" and "ACT". A red handwritten note on the left says "What is I? ALL INFO ABOUT THE WORLD (MAP)". The video player interface includes a search bar, a list of related videos, and a Windows taskbar at the bottom.

GOAL-BASED AGENT

AGENT

SENSORS

GOAL

PLAN

PROC

RULE

RULEBASE

ACT

What is I?  
ALL INFO  
ABOUT  
THE WORLD  
(MAP)

AI L1 P6 Introduction and Agents: Model-based and Goal based Agents

134 views · Aug 2, 2020

rahul kala

Artificial Intelligence Lectures

rahul kala · 6 / 21

- 16:22 rahul kala
- 25:00 AI L1 P6 Introduction and Agents: Model-based and Goal based Agents
- 22:41 AI L1 P7 Introduction and Agents: Utility and Learning Based Agents
- 7:44 AI L2 P1 Searching: When to use a search?
- 25:24 AI L2 P2 Searching: States and Abstraction
- 29:08 AI L2 P3 Searching: Actions & More Examples
- 16:18 AI L2 P4 Searching: Breadth First Search (BFS)

AI L1 P7 Introduction and Agents: Utility and Learning...

rahul kala · 142 views · 1 week ago

22:41

Video Results of the paper Decentralized Multi-Robot...

rahul kala · 92 views · 2 years ago

3:50

Data Analysis with Python: Part 1 of 5 (Live Course)

freeCodeCamp.org

2.4K watching

LIVE NOW

AI L2 P2 Searching: States and Abstraction

YouTube video player showing a lecture titled "GOAL-BASED AGENT" by rahul kala. The video is at 16:22. The main content is a hand-drawn diagram illustrating the relationship between the agent's model and the real world. The diagram is divided into two sections by a vertical line. The left section is labeled "MODEL (WHAT AGENT BELIEVES?)" and contains the text "S IS INTELLIGENT". The right section is labeled "REAL WORLD" and contains the text "S IS NOT INTELLIGENT". A large 'X' is drawn across the entire diagram, indicating a discrepancy or a challenge in the agent's perception. The video player interface includes a search bar, a list of related videos, and a Windows taskbar at the bottom.

MODEL (WHAT AGENT BELIEVES?)

REAL WORLD

S IS INTELLIGENT

S IS NOT INTELLIGENT

AI L1 P6 Introduction and Agents: Model-based and Goal based Agents

134 views · Aug 2, 2020

rahul kala

Artificial Intelligence Lectures

rahul kala · 6 / 21

- 22:09 Problem Types
- 25:12 AI L1 P4 Introduction and Agents: Single and Multi-Agent Systems
- 16:22 AI L1 P5 Introduction and Agents: Reflexive Agents
- 25:00 AI L1 P6 Introduction and Agents: Model-based and Goal based Agents
- 22:41 AI L1 P7 Introduction and Agents: Utility and Learning Based Agents
- 7:44 AI L2 P1 Searching: When to use a search?
- 25:24 AI L2 P2 Searching: States and Abstraction

AI L1 P7 Introduction and Agents: Utility and Learning...

rahul kala · 142 views · 1 week ago

22:41

Video Results of the paper Decentralized Multi-Robot...

rahul kala · 92 views · 2 years ago

3:50

Data Analysis with Python: Part 1 of 5 (Live Course)

freeCodeCamp.org

2.4K watching

LIVE NOW

AI L2 P2 Searching: States and Abstraction

Reflexive Agent (immediately decides what to do and take the required decision)

Classical AI

Model Based or Goal Based agent

Utility and learning based agents(select actions that maximize utility)

Learning Algo(critic) - (what was to be done)



YouTube video player showing a lecture titled "AI L1 P7 Introduction and Agents: Utility and Learning Based Agents" by rahul kala. The video displays a hand-drawn diagram of a "FEEDBACK LEARNING-BASED AGENT".

The diagram illustrates the components and flow of the agent:

- AGENT** (Central Entity):
  - MODEL**: Receives input from the **SENSOR** and provides input to the **RULE ENGINE**.
  - UNDERSTANDING**: Receives input from the **SENSOR** and provides input to the **RULE ENGINE**.
  - RULE ENGINE**: Receives input from the **MODEL** and **UNDERSTANDING**, and provides output to the **ACT** component.
  - ACT**: Receives input from the **RULE ENGINE** and provides output to the **ACTUATOR**.
  - LEARNING ALGO.**: Receives input from the **CRITIC** and provides input to the **RULE ENGINE**.
  - CRITIC**: Receives input from the **ACTUATOR** and provides output to the **LEARNING ALGO.**
  - ACTUATOR**: Receives input from the **ACT** component and provides output to the **ENVIRONMENT**.
  - ENVIRONMENT**: Receives input from the **ACTUATOR** and provides output to the **SENSOR**.

The video player interface shows the video title, duration (14:21), and a list of related videos on the right side.

YouTube video player showing a lecture titled "AI L1 P7 Introduction and Agents: Utility and Learning Based Agents" by rahul kala. The video displays a hand-drawn diagram of a "FEEDBACK LEARNING-BASED AGENT".

The diagram illustrates the components and flow of the agent:

- AGENT** (Central Entity):
  - MODEL**: Receives input from the **SENSOR** and provides input to the **RULE ENGINE**.
  - UNDERSTANDING**: Receives input from the **SENSOR** and provides input to the **RULE ENGINE**.
  - RULE ENGINE**: Receives input from the **MODEL** and **UNDERSTANDING**, and provides output to the **ACT** component.
  - ACT**: Receives input from the **RULE ENGINE** and provides output to the **ACTUATOR**.
  - LEARNING ALGO.**: Receives input from the **CRITIC** and provides input to the **RULE ENGINE**.
  - CRITIC**: Receives input from the **ACTUATOR** and provides output to the **LEARNING ALGO.**
  - ACTUATOR**: Receives input from the **ACT** component and provides output to the **ENVIRONMENT**.
  - ENVIRONMENT**: Receives input from the **ACTUATOR** and provides output to the **SENSOR**.

The video player interface shows the video title, duration (14:21), and a list of related videos on the right side.

YouTube video player interface showing a lecture titled "AI L1 P7 Introduction and Agents: Utility and Learning Based Agents" by rahul kala. The video content displays a hand-drawn diagram of a Learning-Based Agent architecture.

The diagram illustrates the components and flow of a Learning-Based Agent:

- AGENT** (Central Entity):
  - MODEL**: Connected to **UNDERSTANDING**.
  - UNDERSTANDING**: Connected to **CRITIC** and **ACT**.
  - CRITIC**: Connected to **LEARNING ALGO.** and **ACT**.
  - LEARNING ALGO.**: Connected to **MODEL** and **UNDERSTANDING**.
  - RULE BASE**: Connected to **UNDERSTANDING** and **ACT**.
  - RULE ENGINE**: Connected to **RULE BASE** and **ACT**.
  - ACT**: Connected to **ACTUATOR** and **SENSOR**.
  - ACTUATOR**: Connected to **ACT** and **ACTUATORS**.
  - SENSOR**: Connected to **UNDERSTANDING** and **ENVIRONMENT**.
  - ENVIRONMENT**: Represented by a red circle labeled **ENVIRONMENT**.

The video player interface includes a search bar, a list of related videos, and a Windows taskbar at the bottom.