(AI Important Questions)

Sessional 1

Agents and Environments

What is an Agent?

An agent is an entity that perceives its environment through sensors and acts upon that environment through actuators to achieve specific goals.

What is an Environment?

An environment is the external context or space in which an agent operates. It includes everything that the agent can interact with and is influenced by the agent's actions.

Actuators and Effectors

- **Actuators**: Devices through which an agent takes actions within its environment.
- **Effectors**: The components of the actuators that directly cause changes in the environment.

Rational Agent

A rational agent is one that acts to achieve the best expected outcome based on its knowledge and capabilities. It chooses actions that maximize its performance measure, given its percept sequence.

Ideal Rational Agent

An ideal rational agent is a theoretical construct that always takes the best possible action for any given situation to maximize its performance measure. It has complete knowledge and infinite computational resources.

Difference: A rational agent operates within its knowledge and resource constraints, while an ideal rational agent operates under the assumption of perfect knowledge and unlimited resources.

Types of Agents

1. **Simple Reflex Agents**: Act based on the current percept, ignoring the rest of the percept history.
2. **Model-Based Reflex Agents**: Use a model of the world to handle partial observability and maintain an internal state.

3. **Goal-Based Agents**: Take actions to achieve specific goals.
4. **Utility-Based Agents**: Choose actions based on a utility function to maximize happiness or satisfaction.
5. **Learning Agents**: Improve their performance over time through learning from their experiences.

Key Differences: The primary difference lies in their complexity and the way they process percepts to make decisions—from simple reflexes to complex learning mechanisms.

Model vs. Utility

- **Model**: Represents the agent's understanding of the environment, including how actions affect the world.
- **Utility**: A measure of the agent's satisfaction or happiness with a particular state or outcome.

Environments

Types of Environments

1. **Fully Observable vs. Partially Observable**: Whether the agent can access the complete state of the environment at each point in time.
2. **Deterministic vs. Stochastic**: Whether the next state of the environment is determined by the current state and the action executed by the agent.
3. **Episodic vs. Sequential**: Whether the agent's actions are divided into discrete episodes with no dependence on previous episodes.
4. **Static vs. Dynamic**: Whether the environment can change while the agent is deliberating.
5. **Discrete vs. Continuous**: Whether there are a finite number of distinct states and actions or a continuum of them.

Examples:

- **Fully Observable**: Chess
- **Partially Observable**: Poker
- **Deterministic**: Puzzle solving
- **Stochastic**: Backgammon
- **Episodic**: Image recognition tasks
- **Sequential**: Robot navigation
- **Static**: Crossword puzzle

- **Dynamic**: Taxi driving
- **Discrete**: Board games
- **Continuous**: Robotic control

Intelligent Problem Solving Steps

1. **Problem Definition**
2. **Problem Formulation**
3. **Search for Solutions**
4. **Selection of the Best Solution**
5. **Execution and Monitoring**

Constraints

Constraints limit the possible solutions to a problem and can include factors like resources, time, and the rules governing the environment.

Searches and Optimization

Local Search and Optimization

1. **Hill Climbing**: Moves to the neighbor with the highest value until no better neighbors are found.
2. **Simulated Annealing**: Introduces randomness to avoid local maxima by occasionally accepting worse solutions.

Differences: Simulated annealing reduces the probability of getting stuck in local minima by accepting worse solutions at the beginning and decreasing this likelihood over time.

Getting Stuck in Local Minima

To avoid getting stuck in local minima in hill climbing:

- Use random restarts.
- Employ simulated annealing.

Uninformed Searches

Definitions and Failures

1. **Breadth-First Search (BFS)**: Explores all nodes at the present depth level before moving on to nodes at the next depth level. Can fail due to high memory usage.
2. **Depth-First Search (DFS)**: Explores as far down a branch as possible before backtracking. Can fail by getting stuck in deep but unfruitful branches.

3. **Uniform Cost Search (UCS)**: Expands the least costly node first. Can fail due to high memory requirements and long computation times.

Why Use A*: A* uses heuristics to guide the search, significantly improving efficiency over uninformed methods.

Informed Searches

A*

A* is an informed search algorithm that uses a combination of actual cost from the start node (g) and estimated cost to the goal (h) to find the optimal path efficiently.

Adversarial Search

Minimax

Minimax is used in two-player games to minimize the possible loss in a worst-case scenario. It assumes that the opponent is playing optimally.

Uninformed vs. Informed Search

- **Uninformed Search**: Lacks additional information about the problem domain (e.g., BFS, DFS).
- **Informed Search**: Uses heuristics to improve search efficiency (e.g., A*).

Turing Test

The Turing Test evaluates a machine's ability to exhibit intelligent behavior indistinguishable from that of a human. A machine passes the test if a human evaluator cannot reliably tell the machine apart from a human based on their responses to questions.

This overview covers fundamental concepts in artificial intelligence, from agent and environment types to specific search algorithms and problem-solving strategies.