

Agents and Environments in AI

Manuel Mateo Delgado-Gambino López

I. INTRODUCTION

An **agent** perceives its environment via **sensors** and acts through **actuators**. Its behavior is defined by an **agent function**, mapping **percept sequences** to actions. This function is implemented as an **agent program**, operating on current perceptions. Agents are characterized by their components: sensors, rules for decision-making, and actuators.

II. RATIONAL AGENTS

A **rational agent** maximizes its **performance measure** by selecting actions that yield the best expected outcome, given its percept sequence and stored knowledge. Rationality depends on:

- Performance measure (objective success criteria).
- Accumulated environmental knowledge.
- Available actions.
- Percept sequence history.

Rationality differs from omniscience; the former uses available information, while the latter assumes perfect foresight. Rational agents may engage in **information gathering** (e.g., exploration) to reduce uncertainty.

III. TASK ENVIRONMENTS (PEAS FRAMEWORK)

Task environments are defined via the **PEAS** framework:

- **Performance**: Criteria for evaluating agent success based on the environment's objectives.
- **Environment**: Context in which the agent operates (e.g., physical, virtual, social).
- **Actuators**: Tools to execute actions in the environment.
- **Sensors**: Mechanisms to perceive the environment and gather information.

IV. ENVIRONMENT CLASSIFICATION

Environments are classified along seven dimensions:

A. Observability: Fully vs. Partially Observable

- **Fully Observable**: The agent's sensors detect all relevant aspects needed for decision-making.
- **Partially Observable**: Some aspects remain unknown due to sensor limitations.

B. Agents: Single-Agent vs. Multi-Agent

- **Single-Agent**: The agent operates alone without competition.
- **Multi-Agent**: Multiple agents interact, which can be:
 - **Competitive**: Opposing objectives (e.g., adversarial games).
 - **Cooperative**: Shared objectives (e.g., team-based tasks).

C. Determinism: Deterministic vs. Stochastic

- **Deterministic**: The next state is fully determined by the current state and the agent's action.
- **Stochastic**: Outcomes involve probability, introducing uncertainty.
- **Non-Deterministic**: Multiple possible outcomes without explicit probabilities.

D. Episodic vs. Sequential

- **Episodic**: Each perception-action pair is independent of past and future interactions (e.g., image classification).
- **Sequential**: Current actions affect future states, requiring long-term planning (e.g., chess playing).

E. Dynamism: Static vs. Dynamic

- **Static**: The environment does not change while the agent deliberates.
- **Dynamic**: The environment can change during decision-making.
- **Semi-Dynamic**: The environment remains static, but time affects decision consequences.

F. Discrete vs. Continuous

- **Discrete**: Finite states, actions, and time steps (e.g., turn-based games).
- **Continuous**: Infinite states, actions, or real-time decision-making (e.g., robotic motion control).

G. Known vs. Unknown

- **Known**: The agent understands the environment's rules and dynamics.
- **Unknown**: The agent must learn the environment's behavior over time.

V. AGENT ARCHITECTURES

Agents are categorized into four types, increasing in complexity:

- **Simple Reflex**: Uses condition-action rules; limited to fully observable environments.
- **Model-Based Reflex**: Maintains an internal **world model** to handle partial observability.
- **Goal-Based**: Selects actions to achieve specific goals, requiring planning.
- **Utility-Based**: Maximizes a **utility function**, enabling trade-offs between competing objectives.

VI. STATE REPRESENTATIONS

Agents model environments using three representations:

- **Atomic:** States as indivisible entities (e.g., cities in a route-planning problem).
- **Factorized:** States decomposed into attributes (e.g., fuel level, GPS coordinates).
- **Structured:** Entities with explicit relationships (e.g., relational databases).

Expressiveness increases from atomic to structured, but so does computational complexity.

VII. CONCLUSION

Understanding agents and environments is foundational to AI. Rational agents balance performance, knowledge, and uncertainty, while environment classifications (PEAS) guide system design. Advanced agent architectures and state representations enable sophisticated decision-making in real-world applications. Future work may focus on integrating multiple representations to enhance adaptability.