Knowledge Representation & Reasoning

Introduction of the knowledge types, reasoning methods, and representation techniques used in multi-agent systems.

### 1. ****Types of Knowledge:****

* **Declarative Knowledge:** Involves representing facts, concepts, and objects. Examples include databases and facts.
* **Procedural Knowledge:** Refers to knowledge on how to perform tasks or processes, such as rules, strategies, and procedures.
* **Meta-Knowledge:** Knowledge about knowledge, used in planning, modeling, and tagging.
* **Heuristic Knowledge:** Knowledge derived from experience, often used as rules of thumb. Good but not guaranteed.
* **Structural Knowledge:** Represents how various pieces of knowledge interrelate, for example, understanding relationships like "part of" or "kind of."

### 2. ****Types of Reasoning:****

* **Deductive Reasoning:** Involves applying general theories to specific observations to infer new facts (e.g., all men are mortal, Socrates is a man, therefore Socrates is mortal).
* **Inductive Reasoning:** Builds general theories from specific observations or facts (e.g., building a classification dataset for cats and dogs).
* **Abductive Reasoning:** Uses incomplete observations and general theories to deduce the most plausible conclusion (e.g., maze navigation using reinforcement learning).

How do I represent it?

### 3. ****Knowledge Representation:****

The field is concerned with using formal symbols to represent what is known by an agent. Concepts like **propositional attitudes** (knows,belief, hope, fear) and **propositional contexts** (e.g., John knows Mary will come to the party) are emphasized.

Proposition is essentially a statement that can be either true or false (e.g., "Mary will come to the party").

Propositional Attitudes refer to the mental stance or attitude an agent holds towards a proposition,

### Screenshot 2024-10-11 144716

### The propositional context refers to the specific circumstances or conditions under which the agent holds their belief or attitude toward the proposition.

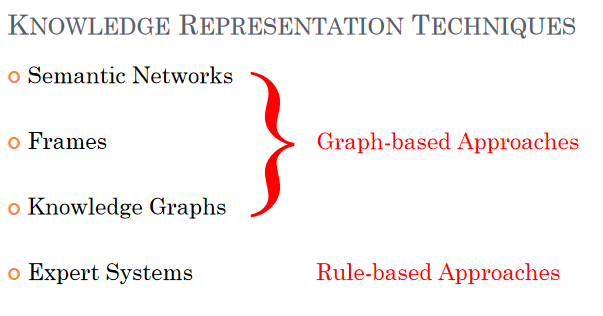
· In the sentence **"John knows that Mary will come to the party"**, the context is the knowledge that John holds based on his prior information (e.g., maybe Mary confirmed to John directly).

· In the sentence **"John fears that Mary will not come to the party"**, the context is driven by John's emotions and the uncertainty surrounding the situation.

While propositional attitudes describe how an agent engages with a proposition (e.g., knowing, believing, hoping),

The propositional context provides the background or mental framework of this engagement (e.g., prior information, emotional state).It highlights how that proposition fits into the mental state of the agent.

### Knowledge ****Representation Techniques:****



* **Semantic Networks**: Visualize relationships between concepts using nodes and edges, which have been in use since the 3rd century AD.
* **Frame-Based Systems:** Used to represent stereotyped situations with slots that hold data or procedures (e.g., being in a living room).
* **Knowledge Graphs**: Accumulate and convey knowledge using nodes representing entities and edges denoting relationships.
* **RDF Graphs (Resource Description Framework)**: A standardized model for web-based data, using subjects, predicates, and objects in a triple format to describe entities.

### 5. ****Expert Systems:****

These systems rely on procedural knowledge and rules to infer new knowledge. Early successes include the SHRDLU and Shakey projects, which could understand language or navigate environments. They mark significant progress in using logic and reasoning in AI, showcasing the early symbolic systems for intelligent action.

### 6. ****Physical Symbol System Hypothesis (PSSH):****

This hypothesis argues that symbol manipulation systems can exhibit general intelligent action. While it gained popularity in the early AI field, it has been criticized, especially with the rise of connectionism, which focuses on sub-symbolic approaches.

Logic & Domain Modelling

Fundamental concepts of logic, reasoning, and domain modeling in the context of multi-agent systems. These are the basis for Astra modelling

Different type of Logics

### 1. ****Rules to Logic:****

* **Rule-based Systems**: These systems are a special case of logical reasoning systems. They consist of facts (propositions) and rules (implications). For example:
  + **Fact**: A, B
  + **Rule**: A ⇒ B (if A, then B)
* **Inference**: This is a logical process that derives new facts from a set of logical statements based on inference rules. A common inference rule is **Modus Ponens**: If A is true and A implies B (A ⇒ B), then B is also true.

### ****Propositional Logic:****

* Includes roles to logic
* This type of logic is limited as it only allows questions about the truth or falsity of simple propositions. An example is using logical statements to derive conclusions about weather:
  + **Logical Statements**:
    - LOW\_PRESSURE
    - CLOUDY
    - LOW\_PRESSURE & CLOUDY ⇒ RAIN\_LIKELY
  + **Proof**: By applying logical rules like conjunction and modus ponens, we can prove that RAIN\_LIKELY is true.

### 3. ****Predicate Logic:****

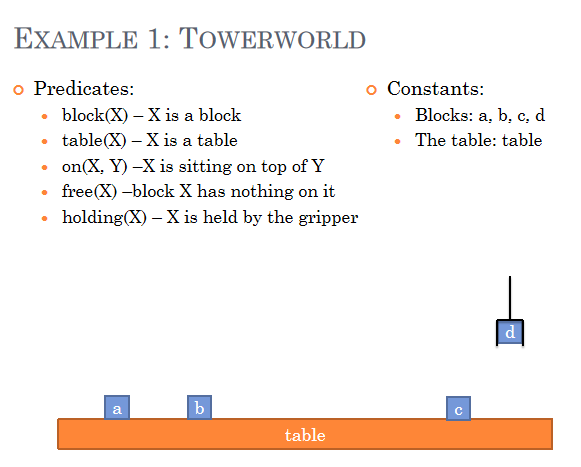
* **Beyond Propositional Logic**: Propositional logic is extended by predicate logic, which allows for more detailed representations. Instead of simple propositions, it uses **predicates** to describe relationships between objects. For example:
  + Constants: “Rem”, “Socrates”, “man”
  + Predicate: is\_a(Socrates, man) means “Socrates is a man.”
* **Quantification**: Predicate logic introduces quantifiers to generalize statements:
  + **Universal Quantification (∀)**: Applies to all objects (e.g., “all men are mortal”).
  + **Existential Quantification (∃)**: There exists at least one object that satisfies a condition (e.g., “there exists a man who is mortal”).

### 4. ****Domain Modelling with Predicate Logic:****

* **Domain Modelling**: This involves identifying objects in an environment and defining relationships between them using predicates. It provides a structured way of representing knowledge about the environment. This method is crucial in multi-agent systems. ASTRA uses predicate logic.

Domain modelling with predicate logic is “simple”

* Identify the kinds of objects that exist in the environment and develop naming conventions for them.
* Specify the relationships that can exist between these objects as predicates.
* In logic, this is referred to a a theory, in MAS it is often considered an ontology.
* **Example 1 – Towerworld**:



* + The logic system models the positioning and status of objects like blocks on a table and whether they are free or held.
* **Example 2 – Cops & Robbers**:
  + A grid world scenario with cops, robbers, and gold on different squares.
  + Predicates:
    - at(X, Y): Location of the cop/robber.
    - cop(X): Cop X is on a square.
    - robber(X): Robber X is on a square.
    - gold(X): There is X amount of gold on the square.

### 5. ****Modeling with Partial Beliefs:****

* **Cops & Robbers Example**: The system models not only the environment but also the beliefs of different agents (cops and robbers) about their positions and the state of the environment. For example:
  + **Cop 1** believes they are at position (1,1).
  + **Robber 2** believes they are at position (6,2) and that Cop 3 is also at (6,2).

### 6. ****Logical Proofs and Resolution:****

* **Resolution Rule**: This is an inference rule that allows the system to infer new facts by resolving contradictions in statements, simplifying logical reasoning in more complex systems.

### 7. ****Applications in Multi-Agent Systems:****

* Logic and domain modeling are applied in systems where agents need to reason about their environment, interact with other agents, and make decisions based on incomplete information. The use of predicate logic in such systems allows for more complex representations and decision-making processes.

Practical Reasoning

### ****1. Introduction to Practical Reasoning and Intentional Stance****

* Practical reasoning involves making decisions on how to act in various situations, balancing competing options based on desires, values, and beliefs.
* The lecturer introduces the **Intentional Stance**, a philosophical theory developed by Daniel Dennett in 1989, which explains how humans tend to describe and predict the behavior of other entities (including people, animals, and machines) by attributing mental states such as beliefs, desires, and intentions to them.
* This stance is applied to describe and predict the behavior of agents in complex systems.It allows developers to model agents as systems that have goals, beliefs, and intentions, making it easier to design their behavior and interaction.

### ****2. The Intentional Stance and System Behavior****

* The **Intentional Stance** is one of three approaches to understanding behavior, alongside the **Physical Stance** (focused on physical laws) and the **Design Stance** (focused on function and design).

Physical Stance: This stance is focused on understanding behavior based on the laws of physics, chemistry, and biology.=> Example: Predicting where a ball will land based on trajectory.

Design Stance: This stance is concerned with understanding objects or systems in terms of their purpose, function, or design. => Example: Predicting that a bird will fly when flapping its wings because this is what wings are for.

Intentional Stance: This stance is about interpreting behavior by attributing mental states, such as beliefs, desires, and intentions, to the entity in question => Example: Predicting the bird will fly away because it knows the cat is coming and is afraid of being eaten.

**Intentional Systems** are described using beliefs, desires, and intentions. This abstraction allows developers to build complex software by applying agent theory.

### ****Practical Reasoning Systems ( no BDI)****

* Practical reasoning is the process of figuring out what to do (reasoning about how to act)
* Practical reasoning involves two main activities:
  1. **Deliberation**: Deciding what the agent wants to achieve( Goals).
  2. **Means-ends Reasoning**: Figuring out how to achieve those goals(i.e., selecting actions or plans).

**Key Components**:

* **Goals**: Represent what the agent wants to achieve (desired states or outcomes).
* **Plans**: Define the sequences of actions to reach those goals.
* **Sub-goals**: Intermediate steps that might need to be addressed before the main goal can be pursued.

**Belief-Desire-Intention (BDI) architecture**

* Beliefs: the current state of the environment
* Desires: the agent ideal future state of the environment
* Intentions: subset of the desires that the agent commits to

*Relationship (4 marks):*

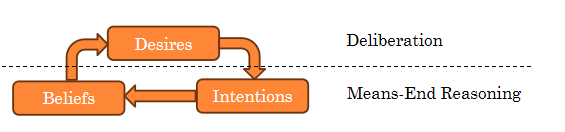
The current state of the environment is represented by the agents beliefs, while the agents desires represent the “ideal” (possibly inconsistent) state of the environment. The agent selects as subset of its desires, which it commits to bringing about and adopts them as intentions. The adopted intentions are mapped to some form of behavior which causes the environment state to change to reflect the intentions (i.e. they become beliefs).

BDI with AgentsSpeak:

* **Beliefs**: the current state
* **Desires**: represented as goals (assumed to be mutually consistent)
* **Intentions**: represented as plans that have been adopted to achieve goals

Intentions are achieved through the application of means-end reasoning.

These guide an agent’s behavior in practical reasoning systems.



### ****4. The Role of Intentions in Practical Reasoning****

* Intentions can be seen as what we are committed to achieving
* Intentions guide agents' actions, requiring them to devote resources to achieve their goals. Intentions also act as filters, preventing the adoption of conflicting goals.
* Agents track their progress toward intentions and can retry or adjust their plans if their first attempt fails.
* Intentions must be achievable and rational, meaning agents must believe they are possible and should not adopt goals that are certain to fail.

### ****Practical Reasoning with BDI****

* 1. **Deliberation**: identifying what desires you are committed to achieving (your intentions)
  2. **Means-ends Reasoning**:adopting plans or actions to achieve your intentions.

### ****4.5. Means-End Reasoning****

The process by we work out the plans that we will use to achieve our intentions:

* Means: how to achieve something (the plan)
* Ends: what we want to achieve (the intention)

A plan can be defined as a specific sequence of actions that the agent should perform to achieve its ends. (very generic definition)

Means-end reasoning strategies:

=>Basic idea is to give an agent in order for him to function and have practical Reasoning:

* A representation of the environment
* A representation of the goal/intention to achieve
* A representation of the actions/plans it can perform
* and have it create/choose a plan to achieve the goal.
* Various planning strategies include:
  + **Planners**: Automatically create plans as goals arise.
  + **Plan Libraries**: Predefined plans are selected based on the situation.
  + **Hybrid Systems**: Combine planning with pre-existing plans.
  + **Reactive Plans**: State-action mappings that define responses to certain states.

### ****6. Planning Systems****

* Planning involves transforming a current state into a goal state by applying a sequence of actions. The presentation introduces **STRIPS (Stanford Research Institute Problem Solver)**, a formal planning system.
* STRIPS uses logic-based descriptions of states and actions. It helps an agent explore possible state transitions, applying actions to move from an initial state to a goal state.

### ****7. STRIPS and Blocksworld Example****

* The Blocksworld problem is used to demonstrate planning using STRIPS. It involves moving blocks to form a desired tower arrangement using actions such as picking up, putting down, stacking, and unstacking blocks.
* The STRIPS framework models actions with **preconditions** (what must be true for the action to happen) and **post-conditions** (how the world changes after the action).
* The planning problem consists of selecting actions that transform the initial state into the goal state while satisfying all preconditions and handling intermediate states.

### ****8. Practical Reasoning Systems and the BDI Model****

* **Practical Reasoning Systems** implement practical reasoning concepts, often using variants of the BDI architecture. These systems combine mental state architectures with algorithms to enable decision-making in agents.
* The **BDI Agent Control Loop** continuously cycles through perception (gathering information), reasoning (updating desires and intentions), and acting (executing actions).

### ****9. Commitment Strategies in Practical Reasoning****

* The presentation explores different commitment strategies for agents, such as:
  + **Blind Commitment**: Agents continue with an intention until it is achieved.
  + **Single-minded Commitment**: Agents maintain intentions until they believe they are either achieved or no longer possible.
  + **Open-minded Commitment**: Agents hold onto intentions as long as they believe the goal is still possible.

In conclusion, the presentation dives into how agents use **practical reasoning** and **planning techniques** to make decisions in multi-agent systems, focusing on the use of the **BDI architecture** and planning systems like **STRIPS** to model and execute complex tasks.