INFORMATICS INSTITUTE OF TECHNOLOGY

CM 2602 - Artificial Intelligence

LOGICAL AGENTS, PREPOSIONAL LOGIC AND FIRST-ORGER LOGIC (FOL)

Motivation of Logical Agents

- Humans can know "things" and do "reasoning".
    - **Representation:** How are the **things stored**?
    - **Reasoning:** How is the **stored information** used to draw **conclusions** and make **decisions**?
        - To solve a problem
        - To generate more knowledge….

Motivation of Logical Agents

- Knowledge and reasoning are important to artificial agents because they enable successful behaviours difficult to achieve otherwise
    - Useful in partially observable environments

- Can benefit from knowledge in very general forms, combining and recombining information.

Logical Agents (Knowledge-Based Agents)

- The idea is that an **agent can represent knowledge** of its world, **its goals** and the **current situation** by **sentences in logic** and **decide what to do** by **inferring** that a certain action or course of action is **appropriate to achieve** its goals.

John McCarthy in Concepts of logical AI, 2000.

Knowledge-Based Agents

- Central component of a knowledge-based agent is a **Knowledge-Base**
    - A set of sentences in a formal language
        - Sentences are expressed using a knowledge representation language

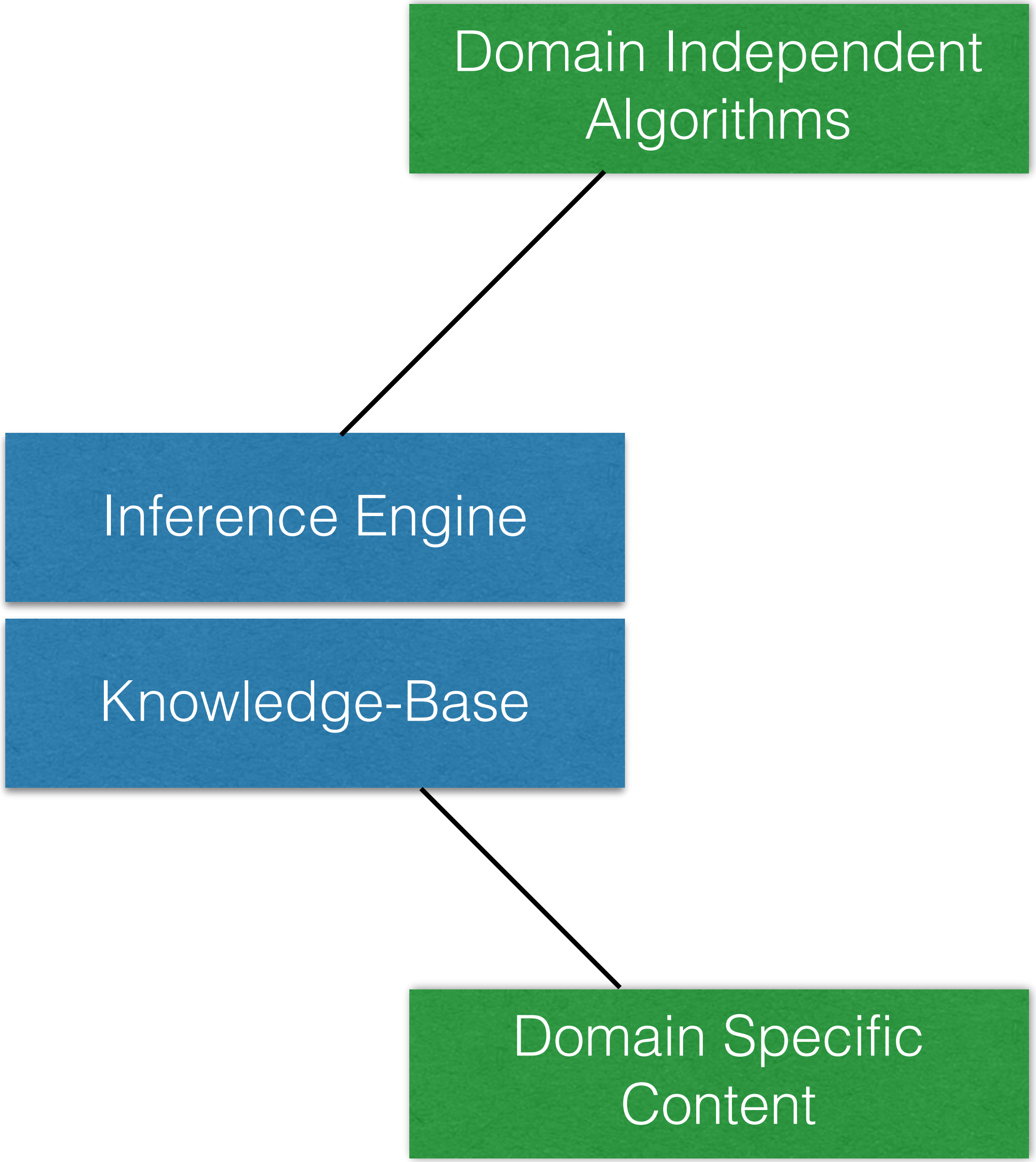
Designing a Knowledge-Based Agents

- **Declarative approach**: We can create a knowledge-based agent by initialising with an empty knowledge base and telling the agent all the sentences with which we want to start with.

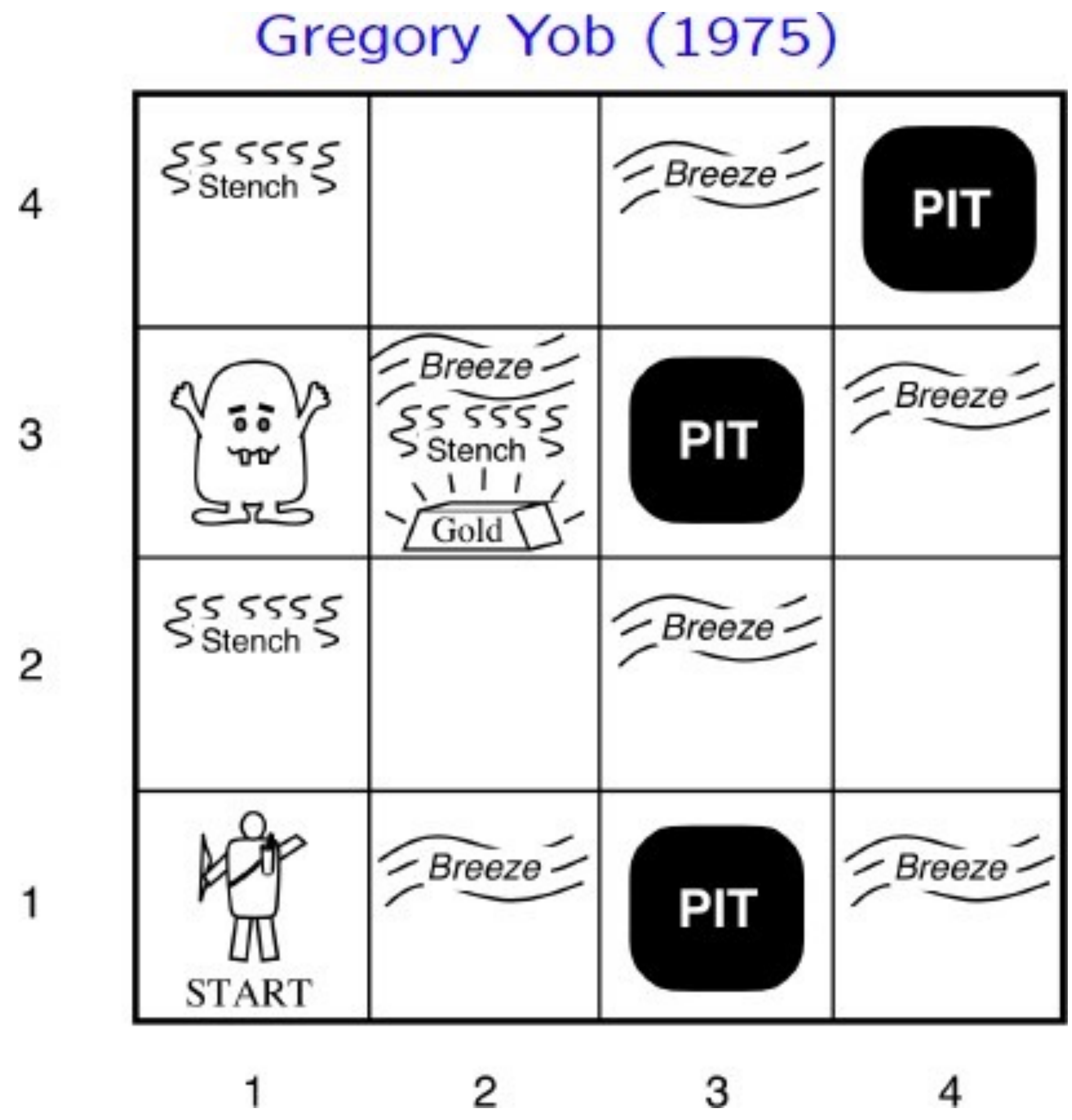
- Two generic functions:
  - **TELL** - add new sentences (facts) to the knowledge base
    - “Tell it what it needs to know”
  - **ASK** - query what is known from knowledge base
    - “Ask what to do next”

Knowledge-Based Agents

- Represent states, actions, etc.
- Incorporate new percepts
- Update internal representations of the world
- Deduce hidden properties of the world
- Deduce appropriate actions



Example: Wumpus World



Wumpus World: PEAS

- **Performance measure**:
  - gold +1000, death (eaten or falling in a pit) -1000, -1 per action taken, -10 for using the arrow.
  - The games ends either when the agent dies or comes out of the cave.
- **Environment**:
  - 4 X 4 grid of rooms
  - Agent starts in square [1,1] facing to the right
  - Locations of the gold, and Wumpus are chosen randomly with a uniform distribution from all squares except [1,1]
  - Each square other than the start can be a pit with probability of 0.2

Wumpus World: PEAS

- **Actuators**:
    - Left turn, Right turn, Forward, Grab, Release, Shoot

- **Sensors**:
  - Stench, Breeze, Glitter, Bump, Scream
  - Represented as a 5-element list
  - Example: [Stench, Breeze, None, None, None]

Wumpus World: Properties

- **Fully Observable**
    - No - Only local perception
- **Deterministic**
    - Yes - outcomes exactly specified
- **Episodic**
    - No
- **Static**
    - Yes - Wumpus and Pits do not move
- **Discrete**
    - Yes
- **Single-Agent**
    - Yes

Logical Reasoning

- In each case where the agent draws a conclusion from the available information, that conclusion is guaranteed to be correct if the available information is correct
- The above is the fundamental of logical reasoning

Logics in General

- Logics are formal languages for representing information such that conclusions can be drawn
- **Syntax** defines the sentences in the language
- **Semantics** define the "meaning" of sentences;
    - i.e., define truth of a sentence in a world
- E.g., the language of arithmetic
    - $x+2 \geq y$ is a sentence; $x2+y > \{\}$ is not a sentence
    - $x+2 \geq y$ is true iff the number $x+2$ is no less than the number y
    - $x+2 \geq y$ is true in a world where $x = 7$, $y = 1$
    - $x+2 \geq y$ is false in a world where $x = 0$, $y = 6$

Propositional Logic

- Propositional Logic (PL) is the simplest logic.
- **Syntax of PL**: defines the allowable sentences or propositions.
- **Definition (Proposition)**: A proposition is a declarative statement that's either True or False.
- **Atomic Proposition**: Single proposition symbol. Each symbol is a proposition. Notation: upper case letters and may contain subscripts.
- **Compound proposition**: Constructed from atomic propositions using parentheses and logical connectives.

Atomic Proposition

- Example of atomic propositions:
  - $2 + 2 = 4$ is a true proposition.
  - $W_{1,3}$ is a proposition. It is true if there is a Wumpus in [1,3]
  - “If there is a stench in [1,2], then there is Wumpus in [1,3]” is a proposition
  - “How are you?” or “Hello” are not propositions. In general, statement that are questions, commands, or opinions are not propositions.

Compound Proposition

- Compound (Complex) proposition built using connectives.
- Let p, p_1 and p_2 be propositions
  - Negation: $\neg p$ is also a proposition. We call a literal either an atomic proposition or its negation.
  - Conjunction: $p_1 \wedge p_2$
  - Disjunction: $p_1 \vee p_2$
  - Implication: $p_1 \rightarrow p_2$
  - Iff (If and only if): $p_1 \leftrightarrow p_2$

Truth Tables

- The semantics define the rules to determine the truth of a sentence.
- Semantics can be specified by truth tables.
- Boolean values domain: T, F
- n-tuple: $(x_1, x_2, \ldots, x_n)$
- Operator on n-tuples: $g(x_1 = v_1, x_2 = v2, \ldots, x_n = = v_n)$
- Definition: A truth table defines an operator g on n-tuples by specifying a boolean for each tuple.
- Number of rows in a truth table? $R = 2^n$

Building Propositions

Negation:

p	**¬p**
T	F
F	T

Building Propositions

Conjunction:

p	**q**	**p ∧ q**
T	T	T
T	F	F
F	T	F
F	F	F

Building Propositions

Disjunction:

p	**q**	**p ∨ q**
T	T	T
T	F	T
F	T	T
F	F	F

Building Propositions

Exclusive or:

p	**q**	**$p \oplus q$**
T	T	F
T	F	T
F	T	T
F	F	F

Building Propositions

Implication:

p	**q**	**p → q**
T	T	T
T	F	F
F	T	T
F	F	T

Building Propositions

Biconditional or IFF(If and only if):

p	**q**	**$p \leftrightarrow q$**
T	T	T
T	F	F
F	T	F
F	F	T

Motivation for First-Order Logic (FOL)

- In propositional logic, we can only represent the facts, which are either true or false.

- Propositional logic has very limited expressive power (unlike natural language)

  - Example:
    - Cannot say “pits cause breezes in adjacent squares”except by writing one sentence for each square

First-OrderLogic (FOL)

- First-Order Logic is another way of knowledge representation in artificial intelligence. It is an extension to propositional logic.

- FOL is also known as **predicate logic**.

First-OrderLogic (FOL)

- **Propositional logic**
  - Assumes that the world contains facts
- **First-Order Logic (FOL)**
  - Assumes that the world contains
    - **Objects**
      - people, houses, numbers, theories, Donald Duck, colors, centuries, ….
    - **Relations**
      - red, round, prime, multistoried,….
      - brother of, bigger than, part of, has color, occurred after, owns,….
    - **Functions**
      - +, middle of, father of, one more than, beginning of, ..

First-OrderLogic (FOL)

- Like natural languages, predicate logic has two main parts:
  - **Syntax**:
    - Collection of symbols and rules
  - **Semantic**:
    - Meaning of the expressions

Syntax of FOL: Basic Elements

Constant

1, 2, A, John, Mumbai, cat,....

Variables

x, y, z, a, b,....

Predicates (represents a property or a relation)

Brother, Father, >,....

Function

sqrt, LeftLegOf, ...

Connectives

$\wedge$, $\vee$, $\neg$, $\Rightarrow$, $\Leftrightarrow$

Equality

==

Quantifier

$\forall$, $\exists$

Syntax of FOL: Atomic Sentences

Atomic sentence

predicate ($term_1, \ldots, term_n$)

or

$term_1 = term_2$

Term

function ($term_1, \ldots, term_n$)

or

constant

or

variable

FOL: Atomic Sentences - Example

$$\underbrace{\underbrace{Brother}_{\text{predicate}}\,(\,\underbrace{\underbrace{KingJohn}_{\text{constant}}}_{\text{term}},\ \underbrace{\underbrace{RichardTheLionheart}_{\text{constant}}}_{\text{term}}\,)}_{\text{atomic sentence}}$$

$$\underbrace{\underbrace{>}_{\text{predicate}}\quad(\,\underbrace{\underbrace{Length}_{\text{function}}(\underbrace{LeftLegOf}_{\text{function}}(\underbrace{Richard}_{\text{constant}}))}_{\text{term}},\ \underbrace{\underbrace{Length}_{\text{function}}(\underbrace{LeftLegOf}_{\text{function}}(\underbrace{KingJohn}_{\text{constant}}))}_{\text{term}}\,)}_{\text{atomic sentence}}$$

Syntax of FOL: Complex Sentences

Built from atomic sentences using connectives

$$\neg S \qquad S_1 \wedge S_2 \qquad S_1 \vee S_2 \qquad S_1 \Rightarrow S_2 \qquad S_1 \Leftrightarrow S_2$$

(as in propositional logic)

Example

$$\underbrace{\underbrace{\underbrace{Sibling}_{\text{predicate}}(\underbrace{KingJohn}_{\text{term}},\underbrace{Richard}_{\text{term}})}_{\text{atomic sentence}} \Rightarrow \underbrace{\underbrace{Sibling}_{\text{predicate}}(\underbrace{Richard}_{\text{term}},\underbrace{KingJohn}_{\text{term}})}_{\text{atomic sentence}}}_{\text{complex sentence}}$$

Semantics in FOL

Models of first-order logic

Sentences are true or false with respect to models, which consist of

- a **domain** (also called universe)
- an **interpretation**

Domain

A non-empty (finite or infinite) set of arbitrary elements

Interpretation

Assigns to each

- constant symbol: a domain element
- predicate symbol: a relation on the domain (of appropriate arity)
- function symbol: a function on the domain (of appropriate arity)

Universal Quantification: Syntax

- Syntax: $\forall$
- Example: $\forall n \in$ Integers: (n is even $\rightarrow n^2$ is even)

Universal Quantification: Semantics

Semantics

$\forall x P$ is true in a model

iff

for all domain elements d in the model:
P is true in the model when x is interpreted by d

Intuition

$\forall x P$ is roughly equivalent to the conjunction of all instances of P

Existential Quantification: Syntax

- Syntax: $\exists$

- Example: $\exists x \in$ Real Numbers: $x^2=4$

Existential Quantification: Semantics

Semantics

$\exists x P$ **is true in a model**

iff

there is a domain element d **in the model such that:**
P **is true in the model when** x **is interpreted by** d

Intuition

$\exists x P$ **is roughly equivalent to the disjunction of all instances of** P

Equality

Semantics

$term_1 = term_2$ **is true under a given interpretation**

if and only if

$term_1$ **and** $term_2$ **have the same interpretation**

Properties of FOL

1.Validity: A formula is valid if it holds true for all possible interpretations. For instance, the formula **$P \lor \neg P$** (where P is a proposition) is always true, regardless of the truth value of **P**. It is known as the principle of the excluded middle.

2.Satisfiability: A formula is satisfiable if it can be made true by some interpretation. For example, the formula **$P \land Q$** is satisfiable when both **P** and **Q** are true.

3.Unsatisfiability: A formula is unsatisfiable if it cannot be made true by any interpretation. For instance, the formula **$P \land \neg P$** (where **P** is a proposition) is always false, regardless of the truth value of **P** . It represents a logical contradiction.

4.Entailment: Entailment occurs when one formula logically implies another. For example, if we have the premises **$\forall x\ (P(x) \rightarrow Q(x))$** and **$\forall x\ P(x)$** , we can logically infer **$\forall x\ Q(x)$.** This means that the truth of the premises implies the truth of the conclusion.