UNIVERSITÀ DEGLI STUDI DI NAPOLI
PARTHENOPE

Artificial Intelligence

Knowledge Representation and Inference

LESSON 10

prof. Antonino Staiano

M.Sc. In ''Machine Learning e Big Data'' - University Parthenope of Naples

Knowledge-based Agents

- Human intelligence is based on knowledge
  - People know information and facts about the world
  - Using that information, people draw conclusions
- Our interest is to focus on the ability to reason based on knowledge when applied to artificial intelligence
- We're going to develop what are known as knowledge-based agents
  - Agents that reason by operating on the internal representation of knowledge

Knowledge-Based Agents

- The core of a knowledge-based agent is the knowledge base
  - A set of sentences
- A sentence is expressed in a knowledge representation language
  - It represents an assertion about the world
- If a sentence is given without being derived from other sentences it is called an axiom
- Drawing conclusions, that is, deriving new sentences from the knowledge base is done through inference

Example from Harry Potter Saga

- Let's take one sentence that we know to be true
  - *If it didn't rain, Harry visited Hagrid today*
- Then take another fact
  - *Harry visited Hagrid or Dumbledore today, but not both*
- Finally, consider a third piece of information
  - *Harry visited Dumbledore today*

Knowledge base

- From this knowledge base, we can try to reason to draw some conclusion
  - *Harry did not visit Hagrid today*
  - *It rained today*
- It's this kind of reasoning, logical reasoning, where humans use logic based on the known information to enrich that information and draw conclusions

Human Reasoning vs Computer Reasoning

- Humans reason about logic generally in terms of human language
  - For instance, focusing on the English language, given a few sentences we try to reason through how it is that they relate to one another
- With computers, we need to be more formal
  - It is necessary to encode the notion of *logic*, and *truth* and *falsehood* inside a machine
- Introduction of a few more terms and a few symbols that will help us to realize the notion of logic inside of artificial intelligence

Logic

- A sentence in a natural language like English is just something that one is saying

- In the context of AI a sentence is
  - An assertion about the world in a knowledge representation language
    - some way of representing knowledge inside of computers

Knowledge-based Agents

- The knowledge base may be
  - extended with new sentences
    - An operation we can call TELL
  - Queried on what is known
    - An operation called ASK

```
function KB-AGENT(percept) returns an action
  persistent: KB, a knowledge base
              t, a counter, initially 0, indicating time

  TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
  action ← ASK(KB, MAKE-ACTION-QUERY(t))
  TELL(KB, MAKE-ACTION-SENTENCE(action, t))
  t ← t + 1
  return action
```

- Inference satisfies the requirement that when one ASKs a question to the knowledge base, the answer should follow from what has been told (or TELLed) previously to KB
- The agent takes a percept as input and returns an action
  - The agent maintains a knowledge base, KB
    - KB contains some background knowledge

Knowledge-based Agents

```
function KB-AGENT(percept) returns an action
  persistent: KB, a knowledge base
              t, a counter, initially 0, indicating time

  TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))
  action ← ASK(KB, MAKE-ACTION-QUERY(t))
  TELL(KB, MAKE-ACTION-SENTENCE(action, t))
  t ← t + 1
  return action
```

- When the agent program is called, it
  1. TELLs the knowledge base what it perceives
  2. ASKs the knowledge base what action it should perform
     - Answering the query, a reasoning process may occur involving the current state of the world, the outcomes of possible action sequences, …
  3. TELLs the knowledge base of the chosen action and returns the action to execute
- MAKE-PERCEPT-SENTENCE constructs a sentence asserting that the agent perceived the given percept at the given time
- MAKE-ACTION-QUERY constructs a sentence that asks what action should be done at the current time
- MAKE-ACTION-SENTENCE constructs a sentence asserting that the chosen action was executed

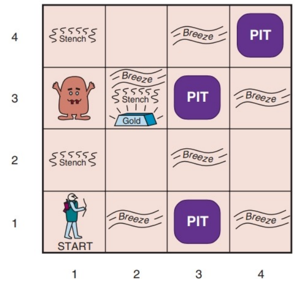
Some other examples

- Automatic theorem proving
  - Write a computer program capable to prove or to refute the following statement
    - Goldbach's conjecture (1742)
      - *For any even number $p>=4$, there exists at least one pair of prime numbers q and r (identical or not) such that*

$$q + r = p$$

Some other examples

- Game playing
  - Write a computer program capable of playing the Wumpus game, a text-based computer game used in a modified version as an AI's toy-problem
  - Basic version
    - The Wumpus world: a cave made up of connected rooms, bottomless pits, a heap of gold, and the Wumpus, a beast that eats anyone who enter its room
    - Goal: starting from room (1,1), find the gold and go back to (1,1), without falling into a pit or hitting the Wumpus
    - Main rules of the game:
      - The content of any room is known only after entering it
      - In rooms neighboring the Wumpus and pits a stench and a breeze are perceived, respectively

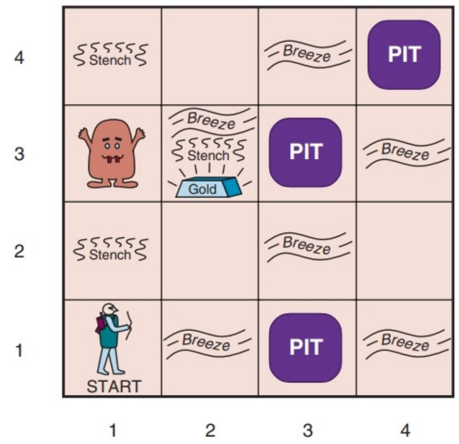


Knowledge-based Systems

- Humans usually solve problems like the ones above by combining high-level abstract knowledge representation and reasoning
- Knowledge-based systems aim at automating the human capabilities
  - Representing knowledge about the world
  - Reasoning to derive new knowledge and to guide action

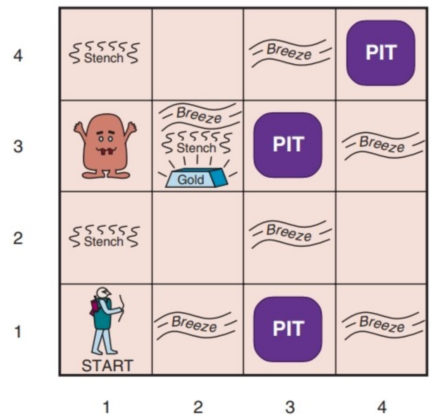
Wumpus World Description (PEAS)

- Performance measure
  - gold +1000, death -1000
  - -1 per step, -10 for using the arrow
- Environment
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills Wumpus if you are facing it
  - Shooting uses up the only arrow
  - Grabbing picks up gold if in the same square
  - Releasing drops the gold in the same square
- Actuators
  - Left turn, Right turn, Forward, Grab, Release, Shoot
- Sensors
  - Breeze, Glitter, Smell



Playing the Wumpus Game

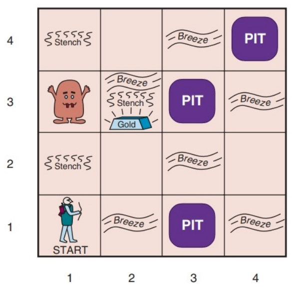
- Consider the following initial configuration of the Wumpus game, and remember that the content of any room is known only after entering it

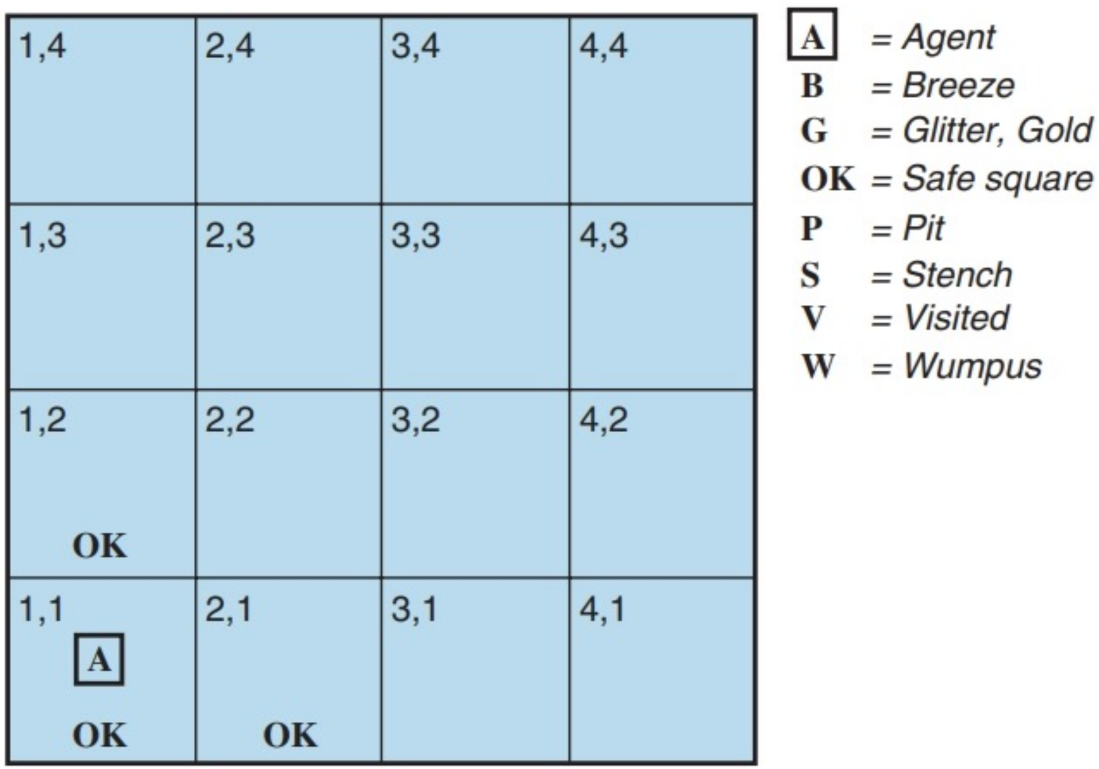


- If you are the player, how would you reason to decide the next move to do at each game step?

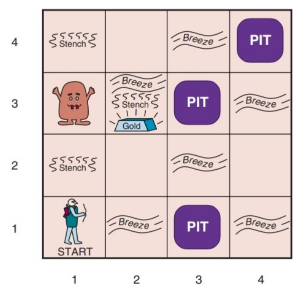
Exploring a Wumpus World

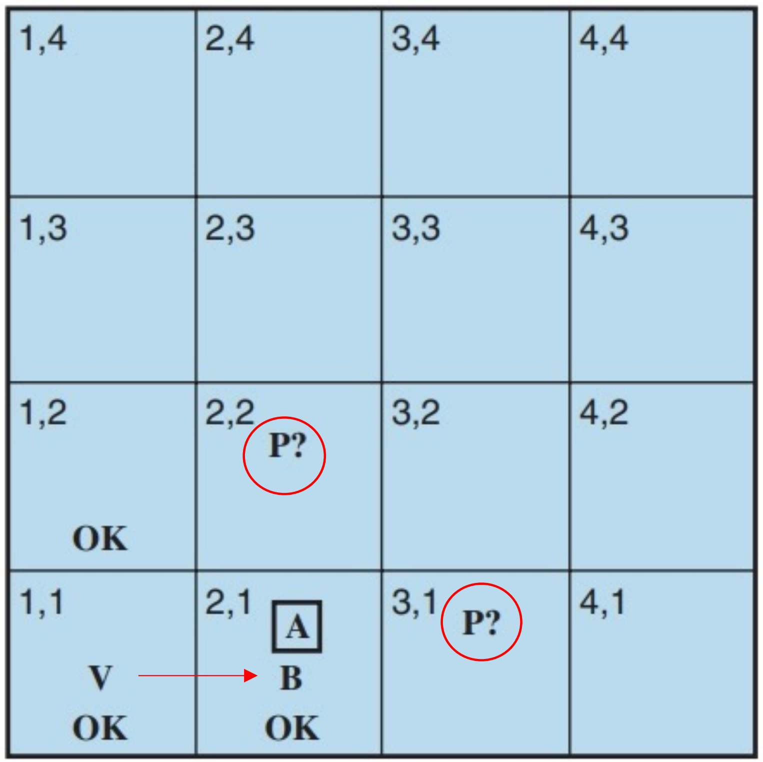
- Sketch of a possible reasoning process for deciding the next move





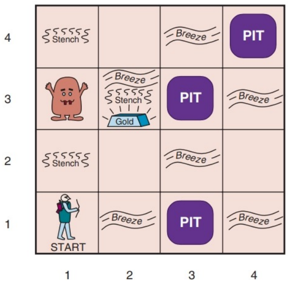
Exploring a Wumpus World

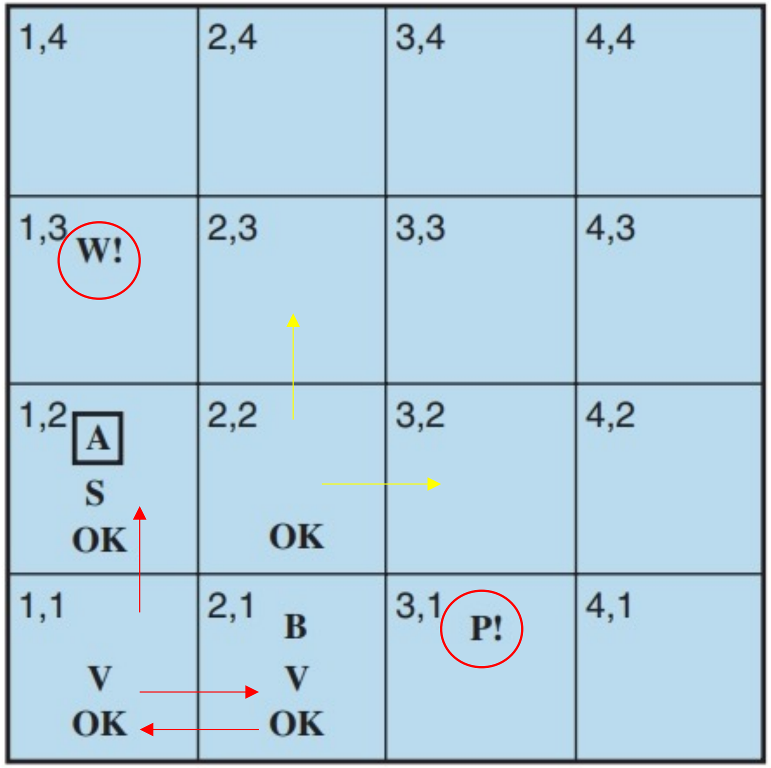




A = *Agent*
B = *Breeze*
G = *Glitter, Gold*
OK = *Safe square*
P = *Pit*
S = *Stench*
V = *Visited*
W = *Wumpus*

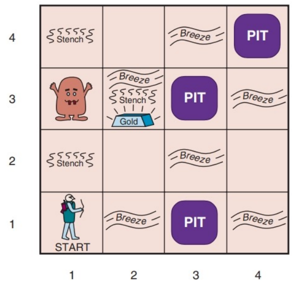
Exploring a Wumpus World

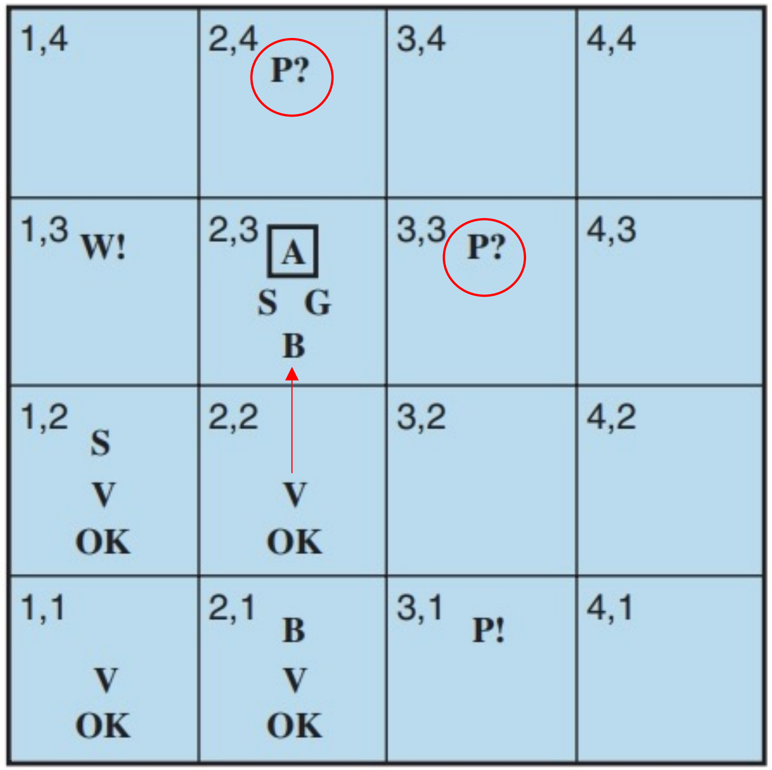




A = *Agent*
B = *Breeze*
G = *Glitter, Gold*
OK = *Safe square*
P = *Pit*
S = *Stench*
V = *Visited*
W = *Wumpus*

Exploring a Wumpus World



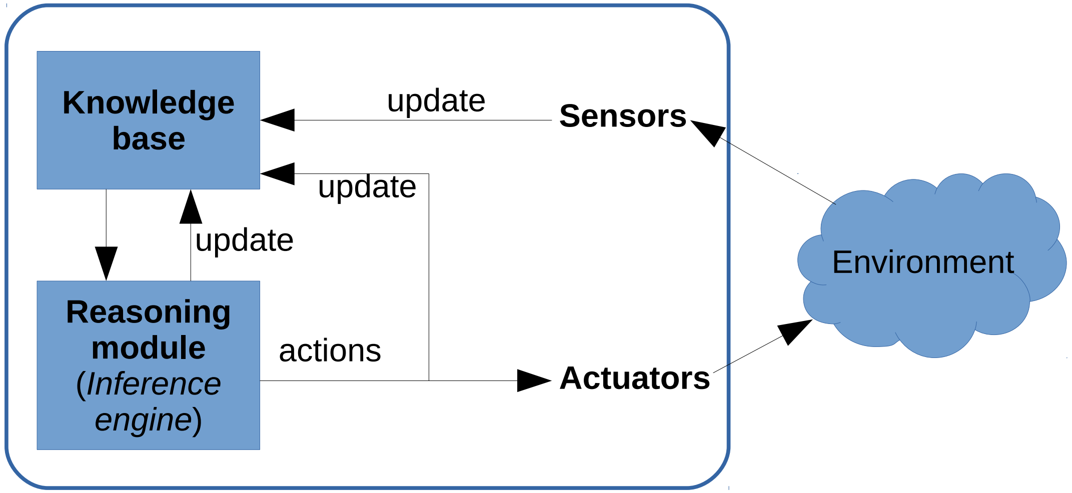


A = *Agent*
B = *Breeze*
G = *Glitter, Gold*
OK = *Safe square*
P = *Pit*
S = *Stench*
V = *Visited*
W = *Wumpus*

Main Approaches to AI system Design

- Procedural
  - The desired behavior (actions) are encoded directly as program code
    - No explicit knowledge representation and reasoning
- Declarative
  - Explicit representation, in a knowledge base, of
    - Background knowledge (e.g., the rules of the Wumpus game)
    - Knowledge about one specific problem instance (e.g., what the agent knows about a specific Wumpus cave it is exploring)
    - The agent's goal
  - Actions are then derived from reasoning

Architecture of Knowledge-based Systems



- A peculiar characteristic is a **separation** between knowledge representation and reasoning
  - The knowledge base contains all the agent's knowledge about its environment, in a **declarative** form
  - The inference engine implements a reasoning process to derive new knowledge and to make decisions

Knowledge Representation and Reasoning

- Logic is one of the main tools used in AI for
  - Knowledge representation
    - Logical languages
      - Propositional logic
      - Predicate (first-order) logic
  - Reasoning
    - Inference rules and algorithms
- Inherited contributions
  - Aristotle (4^{th} century BC): the "laws of thought"
  - G. Boole (1815-64): Boolean algebra (propositional logic)
  - G. Frege (1848-1925): predicate logic
  - K. Gödel (1906-78): incompleteness theorem

Main Applications

- Automatic theorem provers
- Logic programming languages (e.g., Prolog)
- Expert Systems

UNIVERSITÀ DEGLI STUDI DI NAPOLI
PARTHENOPE

Logic

Logic

- The study of conditions under which an argumentation (reasoning) is correct
- The concepts involved in the above definition
  - Argumentation
    - A set of statements consisting of some premises and one conclusion:
      - *All men are mortal*; *Socrates is a man*; *then, Socrates is mortal*
  - Correctness
    - When the conclusion cannot be *false* when all the premises are true
  - Proof
    - A procedure to assess the correctness

Propositions

- Natural language is very complex and vague ⟹ hard to formalize
- Logic considers argumentations made up of only a subset of statements
  - Propositions (declarative statements)
    - *A proposition is a statement expressing a concept that can be either true or false*
- Example
  - *Socrates is a man*
  - *Two and two makes four*
  - *If the Earth had been flat, then Columbus would have not reached America*

Simple and Complex Propositions

- A proposition is
  - Simple, if it does not contain simpler propositions
  - Complex, if it is made up of simpler propositions connected by logical connectives
- Example
  - Simple propositions
    - *Socrates is a man*
    - *Two and two makes four*
  - Complex propositions
    - *A basketball match can be won or lost*
    - *If the Earth had been flat, then Columbus would have not reached America*

Argumentations

- When can a proposition be considered true or false?
  - This is a **philosophical** question
  - Logic does not address this question: it only analyzes the structure of argumentation
- Example
  - *All men are mortal; Socrates is a man; then, Socrates is mortal*
  - Informally, the structure of this argumentation is:
    - All P are Q; *x* is P; then *x* is Q
- Is it correct, **whatever** P, Q, and *x* are, that is, **regardless** of whether the corresponding propositions "all P are Q", "*x* is P" and "*x* is Q" are true or false?

Formal Languages

- Logic provides formal languages for representing propositions in the form of sentences
- A formal language is defined by syntax and semantics
  - Syntax (grammar)
    - Rules that define what sentences are well-formed
  - Semantics
    - Rules that define the meaning of well-formed sentences
      - The truth of sentences with respect to each possible world
- Examples (of formal languages)
  - Arithmetic: propositions about numbers
  - Programming languages: instructions to be executed by a computer
- Logicians typically think in terms of models, which are formally structured worlds with respect to which truth can be evaluated
  - We say m is a model of a sentence α if α is *true* in m
    - M (α) is the set of all models of α

Natural vs Logical Languages

- Natural languages
  - Syntax is not rigorously defined
  - Semantics defines the content of a statement, i.e., what it refers to in the real world
- Example (syntax)
  - *The book is on the table*: syntactically correct statement, with a clear semantics
  - *Book the on is table the*: syntactically incorrect statement, no meaning can be attributed to it
  - *Colorless green ideas sleep furiously*: syntactically correct, but what does it mean?

Natural vs Logical Languages

- Logical languages
  - Syntax
    - formally defined
  - Semantics
    - Rules that define the truth value of each well-formed sentence w.r.t. each possible model (a possible world represented by that sentence)
- Example
  - Syntax
    - **x + y = 4** is a well-formed sentence, **x4y + =** is not
  - Model
    - The symbol 4 represents the natural number four, x and y any pair of natural numbers, + the sum operator, etc.
  - Semantics
    - x + y = 4 is true for x = 1 and y = 3, x = 3 and y = 2, etc.

Logical Entailment

- Logical reasoning is based on the relation of logical entailment between sentences
  - Defines when a sentence logically follows from one another
- Definition
  - The sentence α entails the sentence β if and only if in every model in which α is true, also β is true

$$\alpha \vDash \beta$$

- Example (arithmetic)
  - $x + y = 4 \vDash x = 4 - y$
    - In every model (i.e., for any assignment of numbers to x and y) in which $x + y = 4$ is *true*, also $x = 4 - y$ is *true*

Logical Inference

- Logical inference
  - *The process of deriving conclusions from premises*
- Inference algorithm
  - *A procedure that derives sentences (conclusions) from other sentences (premises) in a formal language*
- Formally, the fact that an inference algorithm A derives a sentence α from a set of sentences (knowledge base) KB is written as

$$KB \vdash_A \alpha$$

Properties of Inference Algorithms

- Definition
  - Soundness (truth preservation)
    - If an inference algorithm derives only sentences entailed by the premises, i.e.:

$$\text{if } KB \vdash_A \alpha, \text{ then } KB \vDash \alpha$$

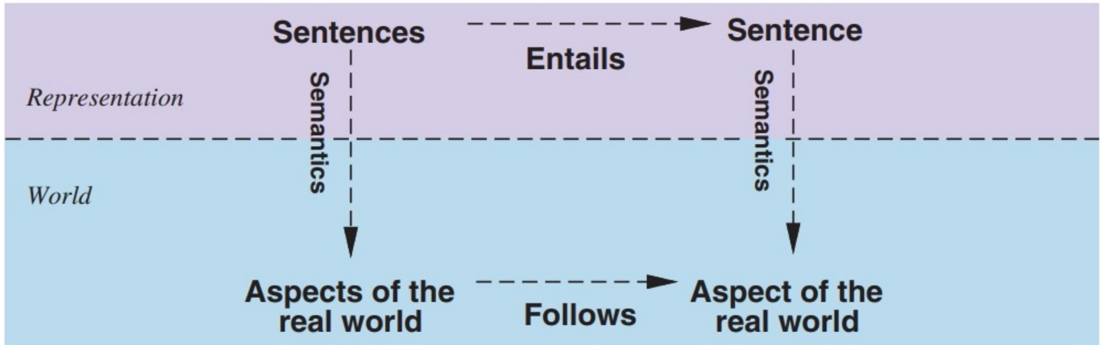
  - Completeness
    - If an inference algorithm derives all the sentences entailed by the premises, i.e.:

$$\text{If } KB \vDash \alpha, \text{ then } KB \vdash_A \alpha$$

- A sound algorithm derives conclusions that are guaranteed to be true in any world in which the premises are true

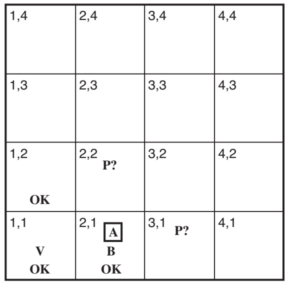
Properties of Inference Algorithms

- Inference algorithms operate only at the syntactic level:
  - sentences are physical configurations of an agent (e.g., bits in registers)
  - inference algorithms construct new physical configurations from previous ones
  - logical reasoning should ensure that new configurations represent aspects of the world that actually follow from the ones represented by previous configurations



Applications of Inference Algorithms

- In AI inference is used to answer two main kinds of questions
  - Does a given conclusion α logically follow from the agent's knowledge KB? That is, $KB \models \alpha$
  - What are all the conclusions that logically follow from the agent's knowledge? That is, find all α such that $KB \models \alpha$
- Example
  - Does a breeze in room (2,1) entail the presence of a pit in room (2,2)?
  - What conclusions can be derived about the presence of pits and of the Wumpus in each room, from the current knowledge?



Inference Algorithms: Model Checking

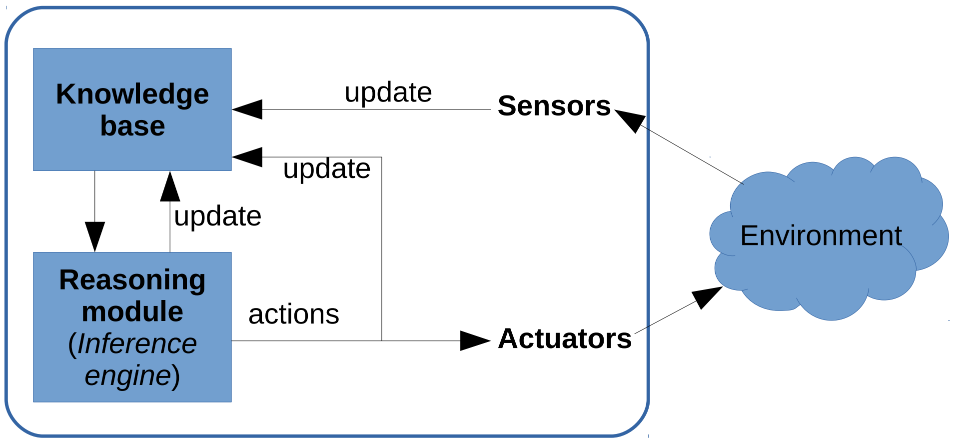
- The definition of entailment can be directly applied to construct a simple inference algorithm, named Model checking
  - Given a set of premises, KB, and a sentence α, enumerate all possible models and check whether α is *true* in every model in which KB is *true*
- Example (arithmetic)
  - $KB : \{x + y = 4\}$
  - $\alpha : y = 4 - x$
  - Is the inference $\{x + y = 4\} \vdash y = 4 - x$ correct?
    - Model checking: enumerate all possible pairs of numbers x, y, and check whether $y = 4 - x$ is true whenever $x + y = 4$ is

The Issue of Grounding

- A knowledge base *KB* is just "syntax" (a physical configuration of the agent):
  - what is the connection between a *KB* and the real world?
  - how does one know that *KB* is true in the real world?
- This is a **philosophical** question. For humans:
  - a set of beliefs (set of statements considered true) is a physical configuration of our brain
  - how do we know that our beliefs are true in the real world?
- A simple answer can be given for agents (e.g., computer programs or robots): the connection is created by
  - sensors, e.g.: perceiving a breeze in the wumpus world
  - learning, e.g., when a breeze is perceived, there is a pit in some adjacent room
  - Of course, both perception and learning are **fallible**

Architecture of Knowledge-based Systems Revisited

- If logical language is used
  - Knowledge base
    - A set of sentences in a given logical language
  - Inference engine
    - An inference algorithm for the same logical language



Logical Languages

- Propositional logic
  - The simplest logic language
  - An extension of Boolean algebra
- Predicate (or first-order) logic
  - A more expressive and concise extension of propositional logic
  - Seminal work by Frege