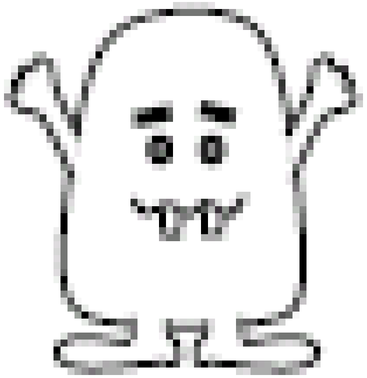
CSE 473

Lecture 9

Logic and Reasoning

Outline
(Chapter 7)

Knowledge-based agents

Wumpus world

Logic in general

Propositional logic

- Inference, validity, equivalence and satisfiability
- Reasoning
  - Resolution
  - Forward/backward chaining

Knowledge-Based Logical Agents

Chess program doesn't know that no piece can be on 2 different squares at the same time

Knowledge-based logical agents combine general **knowledge about the world** with current **percepts** to *infer hidden aspects* of their state

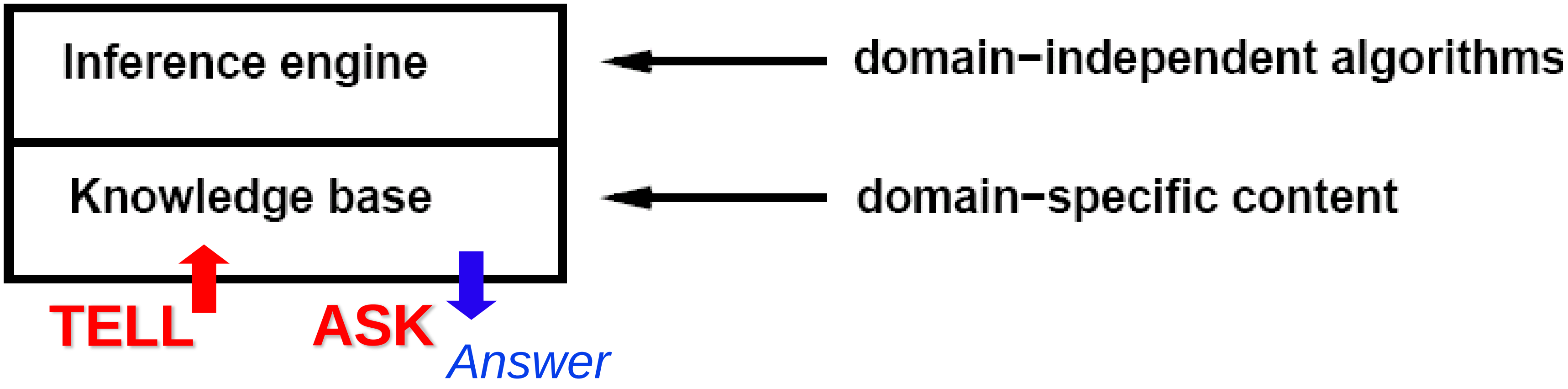
- Crucial in partially observable environments

Knowledge Base and Inference

Knowledge Base : set of sentences represented in a knowledge representation language

- stores assertions about the world

Inference: when you ASK the KB a question, answer should *follow* from what has been TELLed to the KB previously

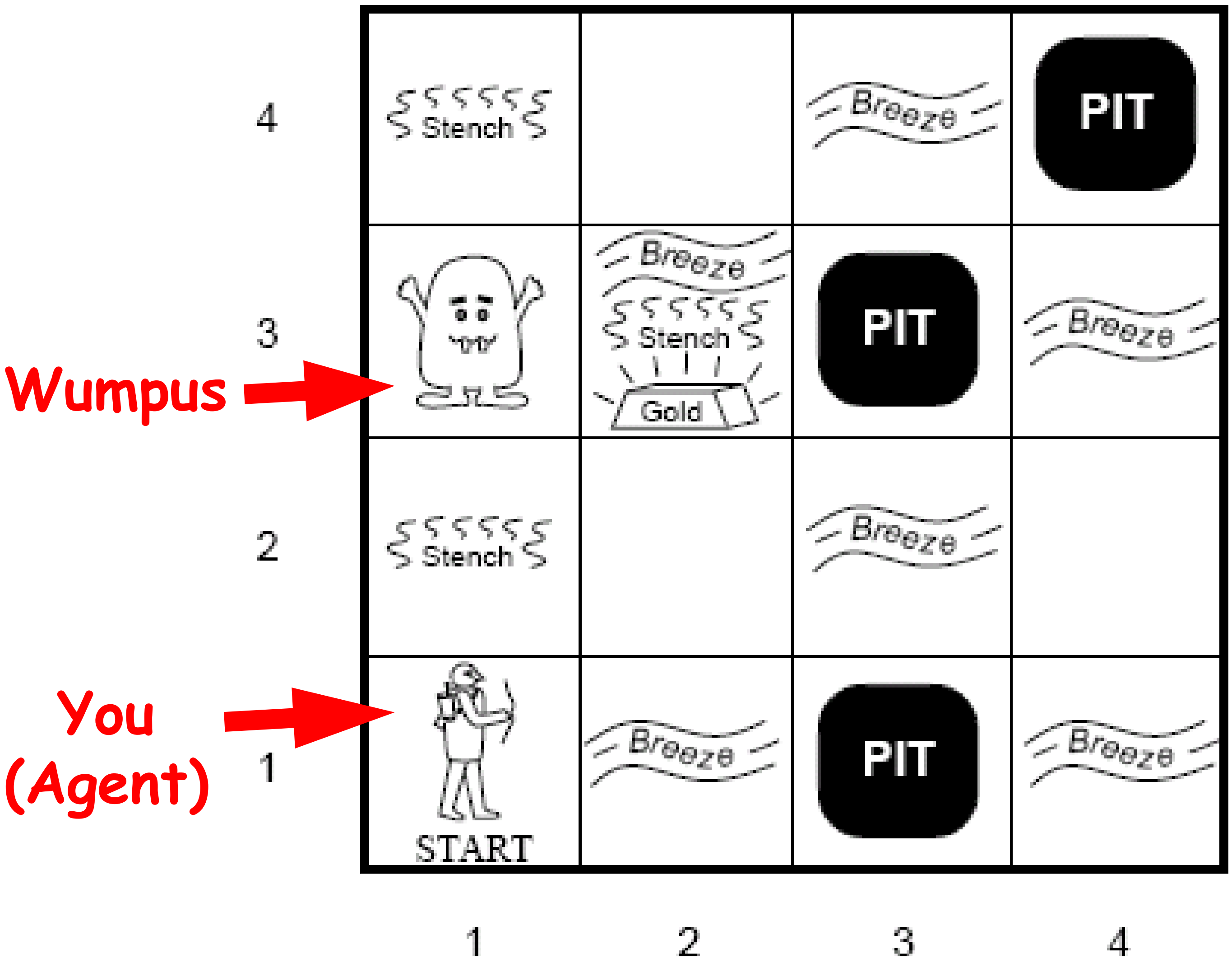


Abilities of a KB agent

Agent must be able to:

- Represent states and actions
- Incorporate new percepts
- Update internal representation of the world
- Deduce hidden properties of the world
- Deduce appropriate actions

A Typical Wumpus World



Wumpus World PEAS Description

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 same square

Climbing in [1,1] gets agent out of the cave

Sensors Stench, Breeze, Glitter, Bump, Scream

Actuators TurnLeft, TurnRight, Forward, Grab, Shoot, Climb

Wumpus World Characterization

Observable?

No, only local perception

Deterministic?

Yes, outcome exactly specified

Episodic?

No, sequential at the level of actions

Static?

Yes, Wumpus and pits do not move

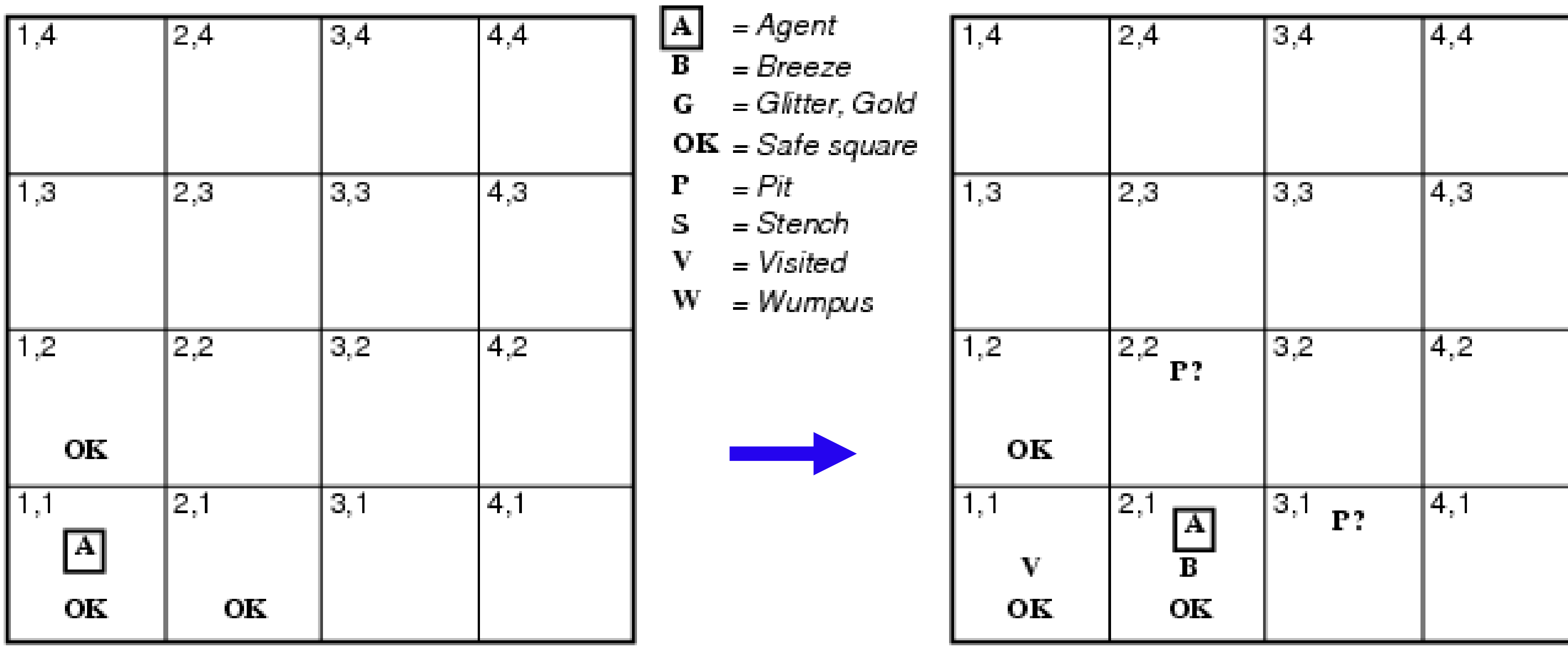
Discrete?

Yes

Single-agent?

Yes, Wumpus is essentially a "natural" feature of the environment

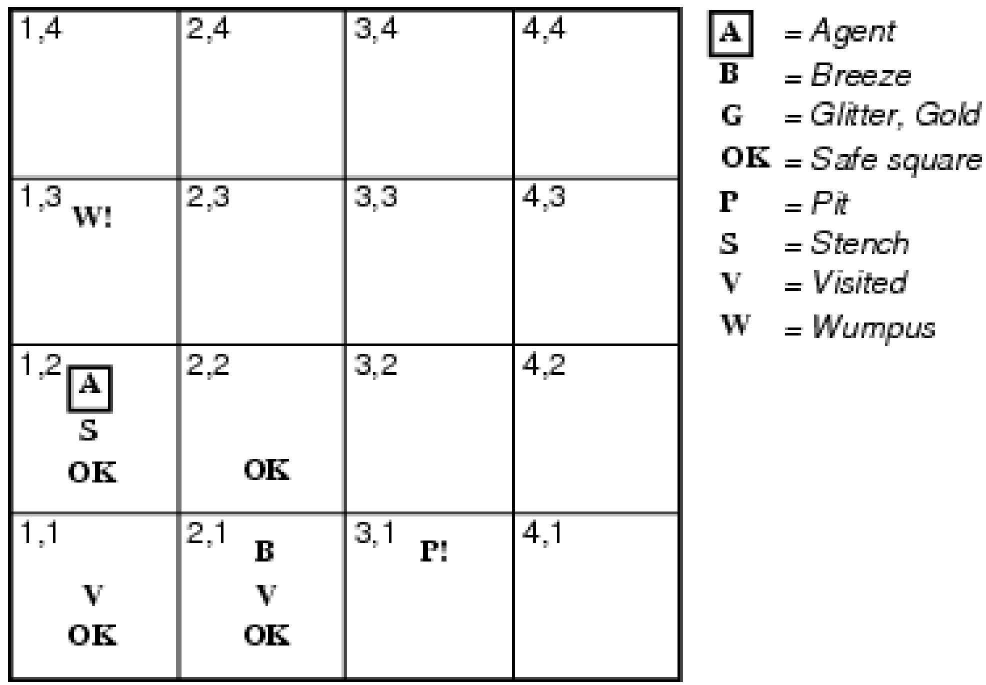
Exploring the Wumpus World



[1,1] KB initially contains the rules of the environment. First percept is [*none,none,none,none,none*], move to safe cell e.g. 2,1

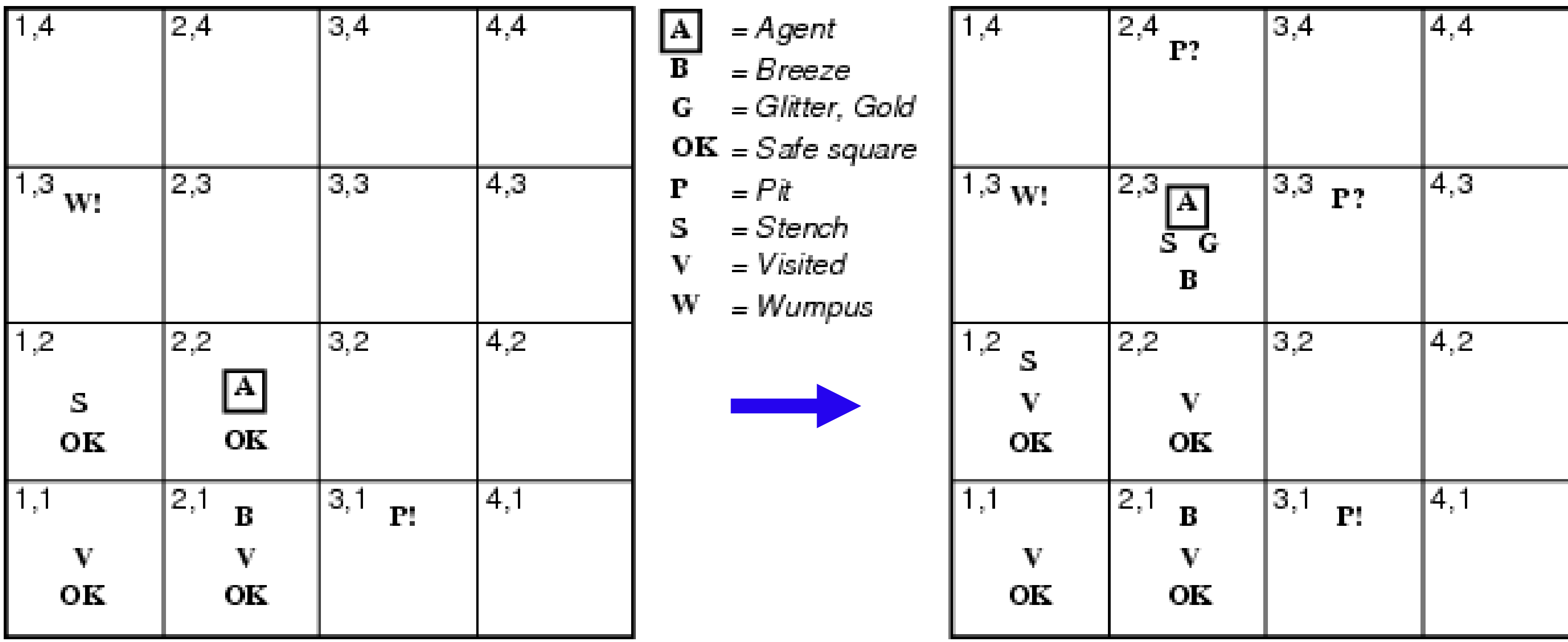
[2,1] Breeze which indicates that there is a pit in [2,2] or [3,1], return to [1,1] to try next safe cell

Exploring the Wumpus World



[1,2] Stench in cell which means that wumpus is in [1,3] or [2,2]
but not in [1,1]
YET … wumpus not in [2,2] or stench would have been
detected in [2,1]
THUS … wumpus must be in [1,3]
ALSO [2,2] is safe because of lack of breeze in [1,2]
THEREFORE pit must be in [3,1]
Move to next safe cell [2,2]

Exploring the Wumpus World



[2,2] Move to [2,3]

[2,3] Detect glitter, smell, breeze
Grab gold
Breeze implies pit in [3,3] or [2,4]

How do we represent rules of the world and percepts encountered?

Why not use logic?

What is a logic?

A formal language

- **Syntax** – what expressions are legal (well-formed)
- **Semantics** – what legal expressions mean
  - In logic the truth of each sentence evaluated with respect to *each possible world*

E.g. the language of arithmetic

- Syntax: x+2 >= y is a legal sentence
  x2y+= is not
- Semantics:
  - x+2 >= y is true in a world where x=7 and y=1
  - x+2 >= y is false in a world where x=0 and y=6

How do we draw conclusions and deduce new facts about the world using logic?

Entailment

Knowledge Base KB

Sentence α

KB $\models \alpha$ **(KB "entails" sentence α)**
if and only if α **is true in all worlds (models) where KB is true.**

E.g. x>4 entails x>0

(because x>0 is true for all values of x for which x>4 is true)

But not vice versa! (x>0 does not entail x>4)

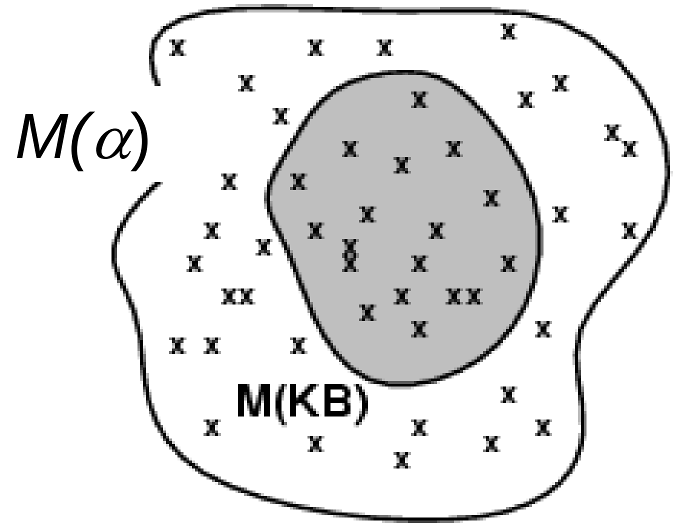
Models and Entailment

m is a model of a sentence α if α is true in *m*

e.g. $m = \{x=5\}$ is a model for α = "x>4"

$m = \{x=2\}$ is a model for α = "x>0" but not for the sentence "x>4"

$M(\alpha)$ is the set of all models of α

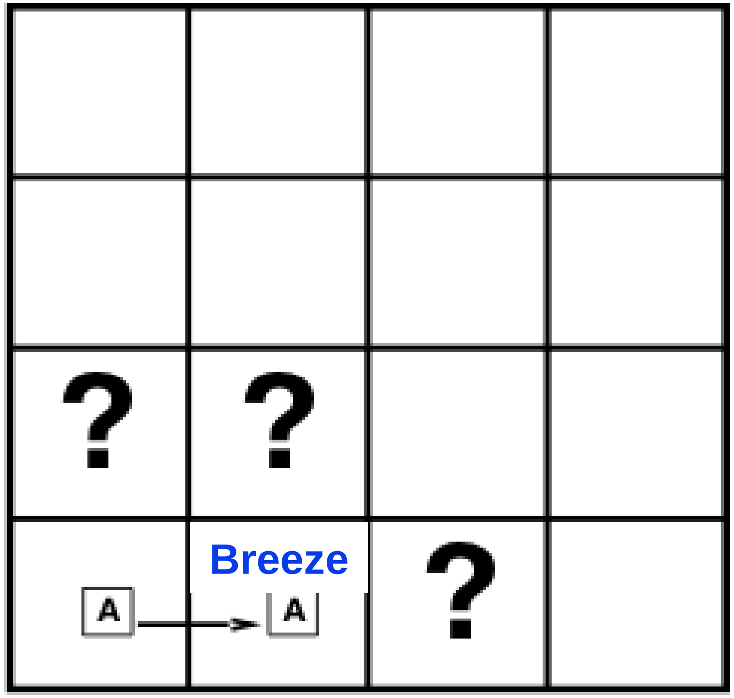


Then KB $\models \alpha$ iff $M(KB) \subseteq M(\alpha)$

E.g. KB = x>4
α = x>0

Wumpus world model

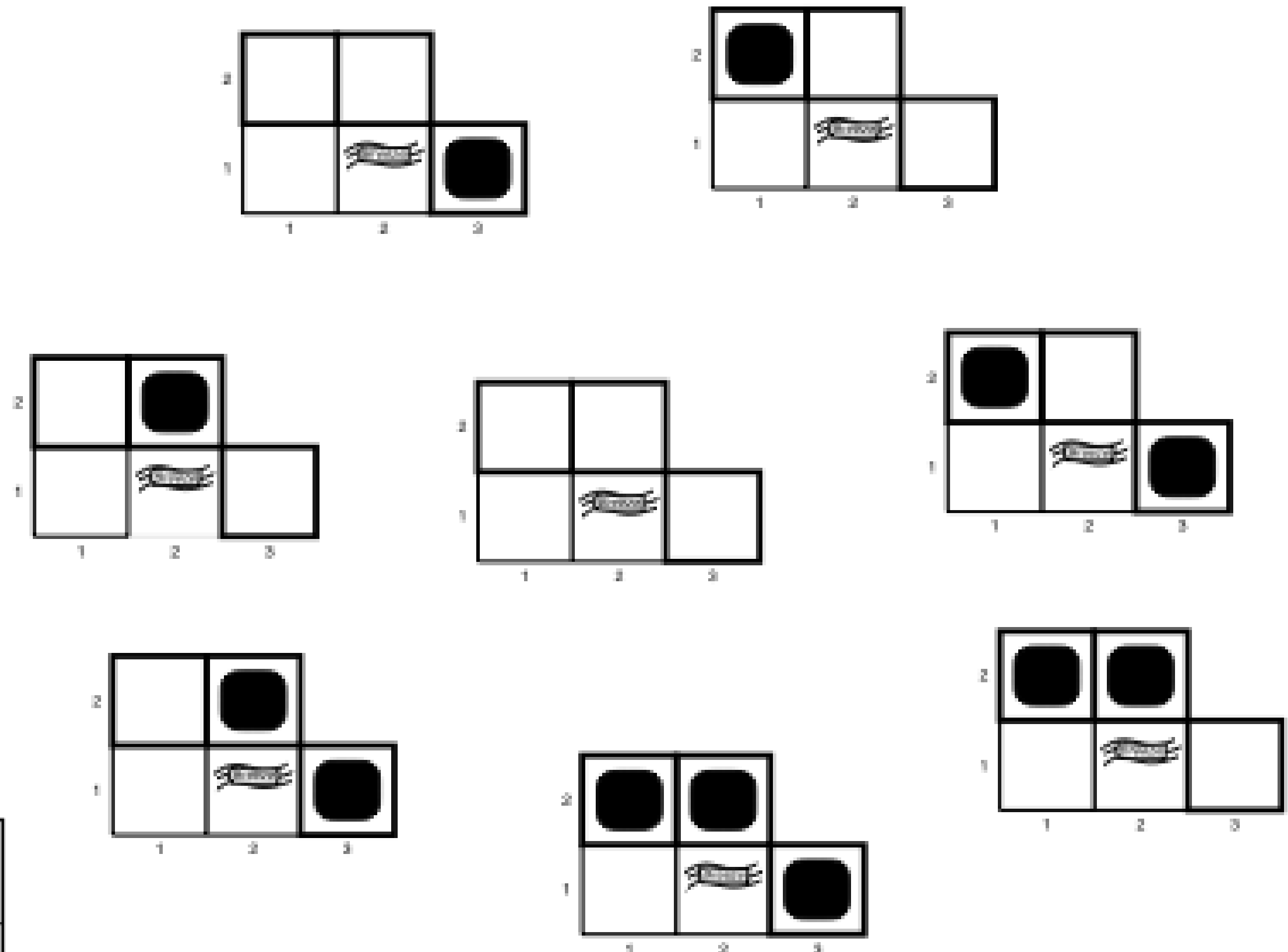
Situation after detecting nothing in [1,1], moving right, breeze in [2,1]



Consider possible models for ?s assuming only pits

3 Boolean choices $\Rightarrow$ 8 possible models

8 possible models for pits

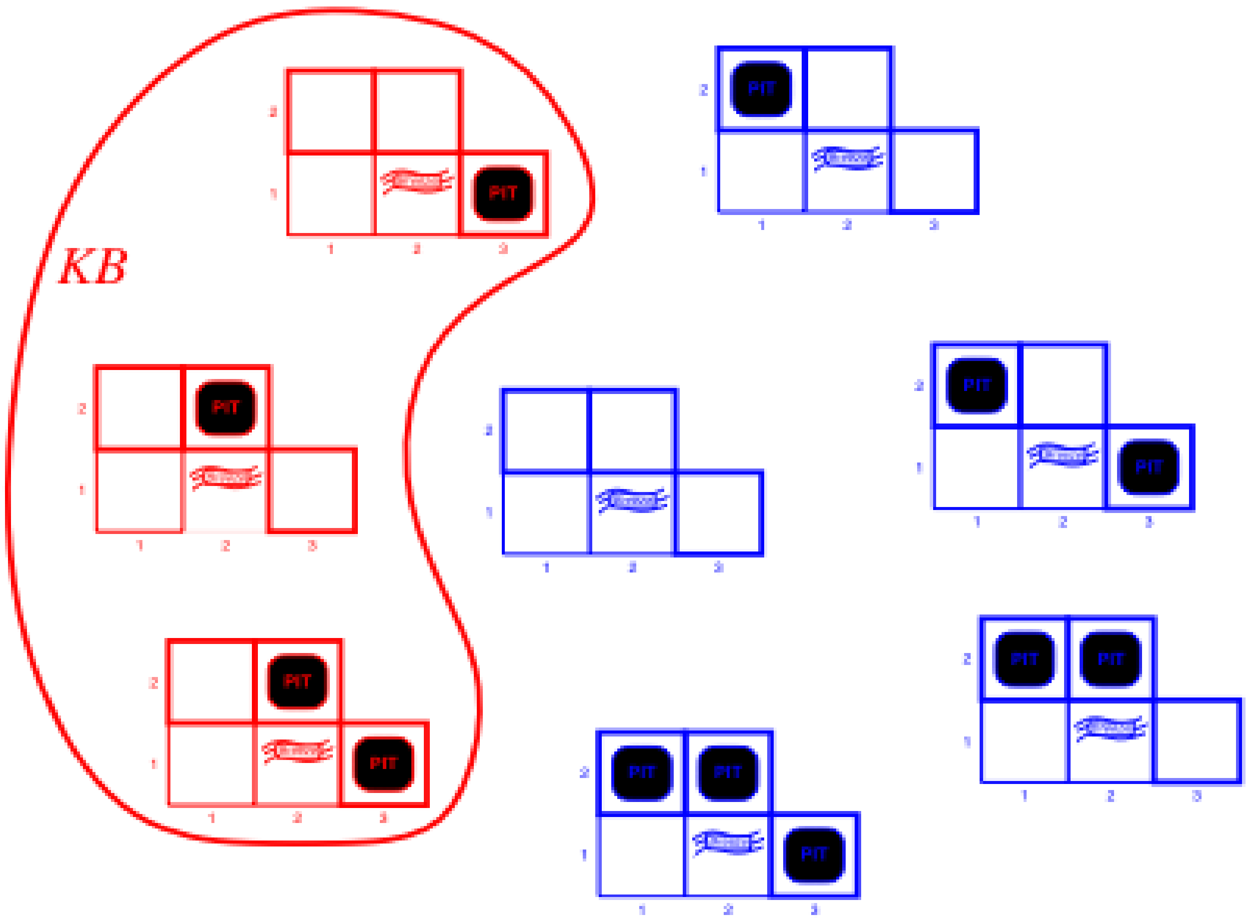


1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2 OK	2,2 P?	3,2	4,2
1,1 V OK	2,1 $\boxed{A}$ B OK	3,1 P?	4,1

Observations thus far

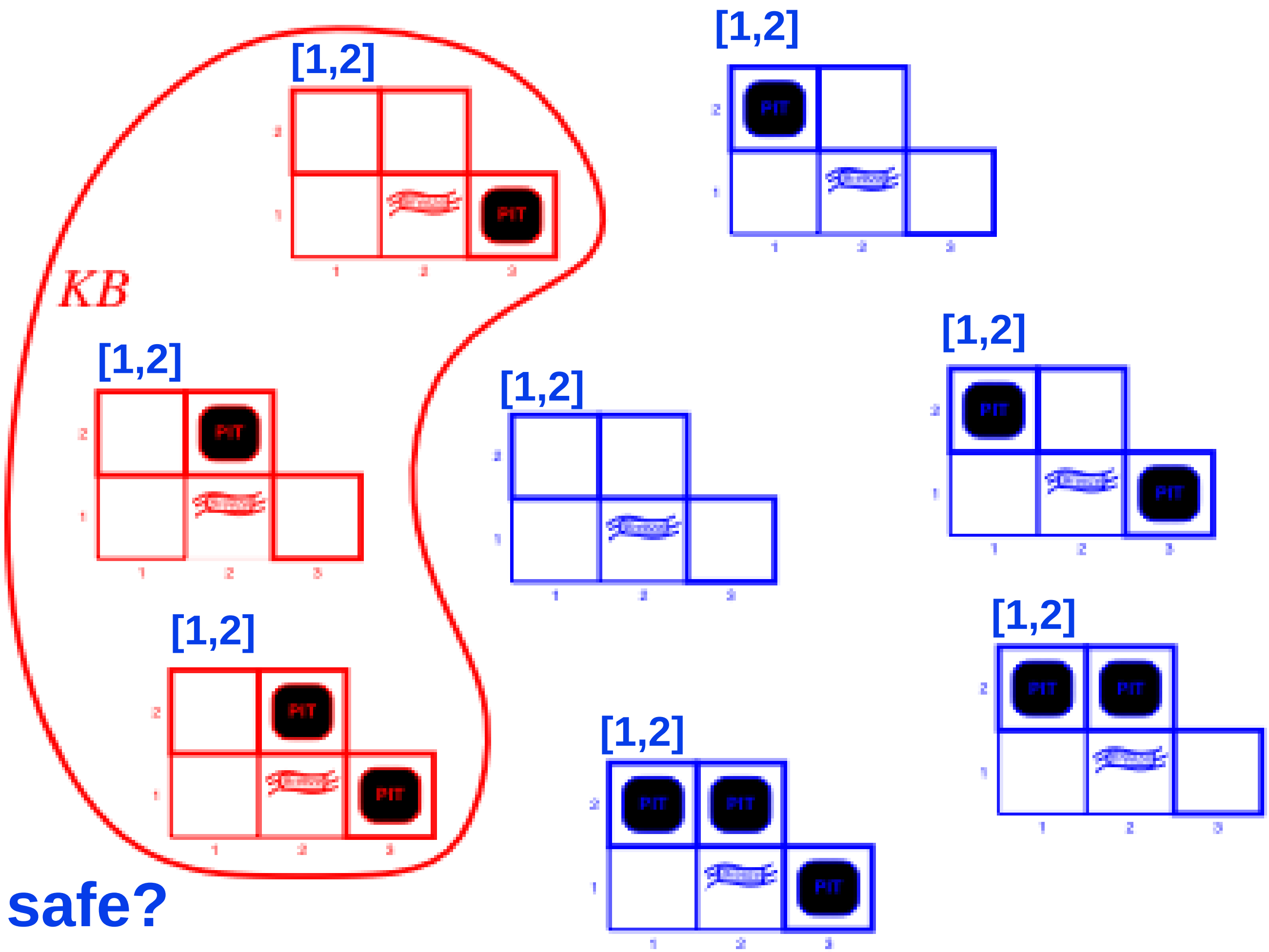
Models consistent with rules + observations?

Models consistent with rules + observations



KB = wumpus-world rules + observations

Example of Entailment



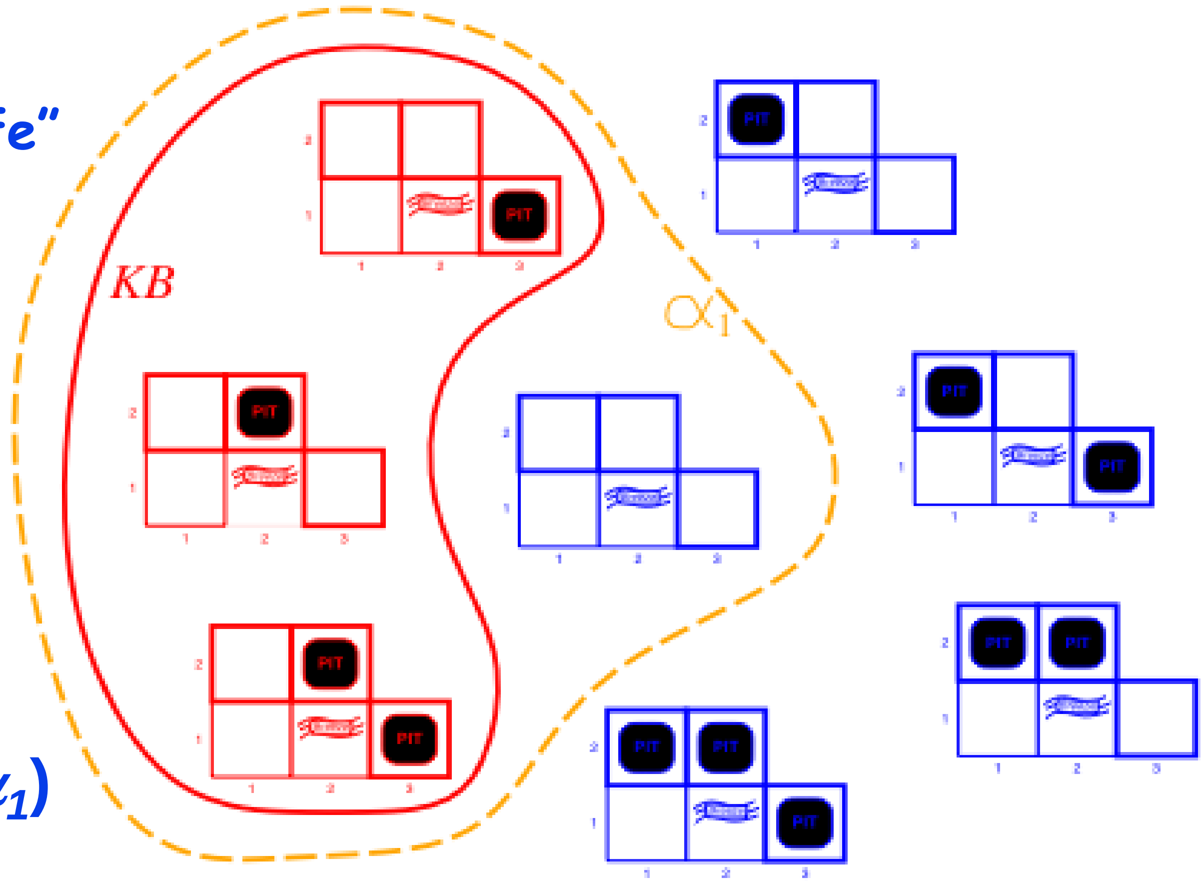
ASK: Is [1,2] safe?

α_1 = "[1,2] is safe"; Does *KB* $\models \alpha_1$?

Is $M(KB) \subseteq M(\alpha_1)$?

Entailment by Model Checking

α_1 = "[1,2] is safe"



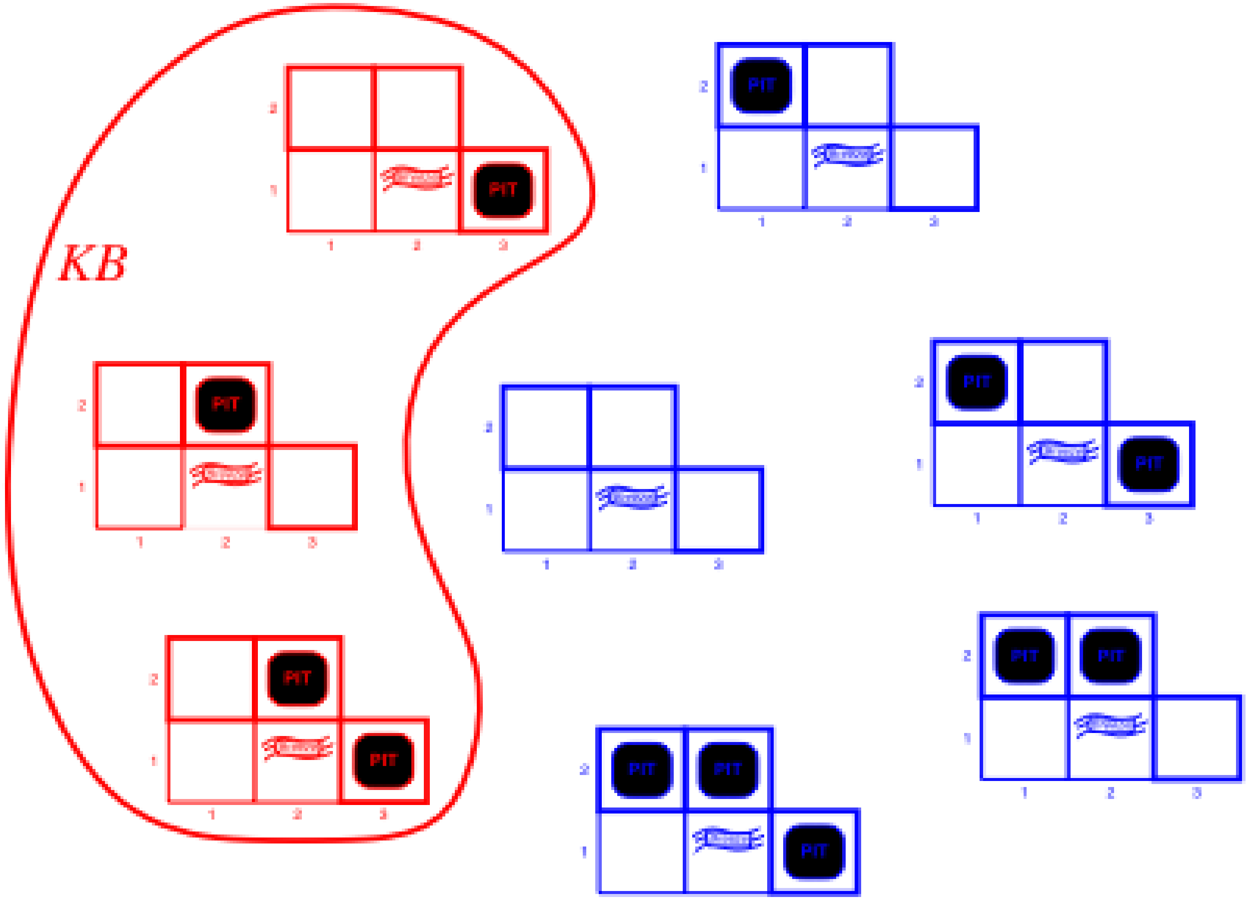
M(KB) ⊆ M(α₁)

KB = wumpus-world rules + observations

α_1 = "[1,2] is safe", $KB \models \alpha_1$, proved by model checking

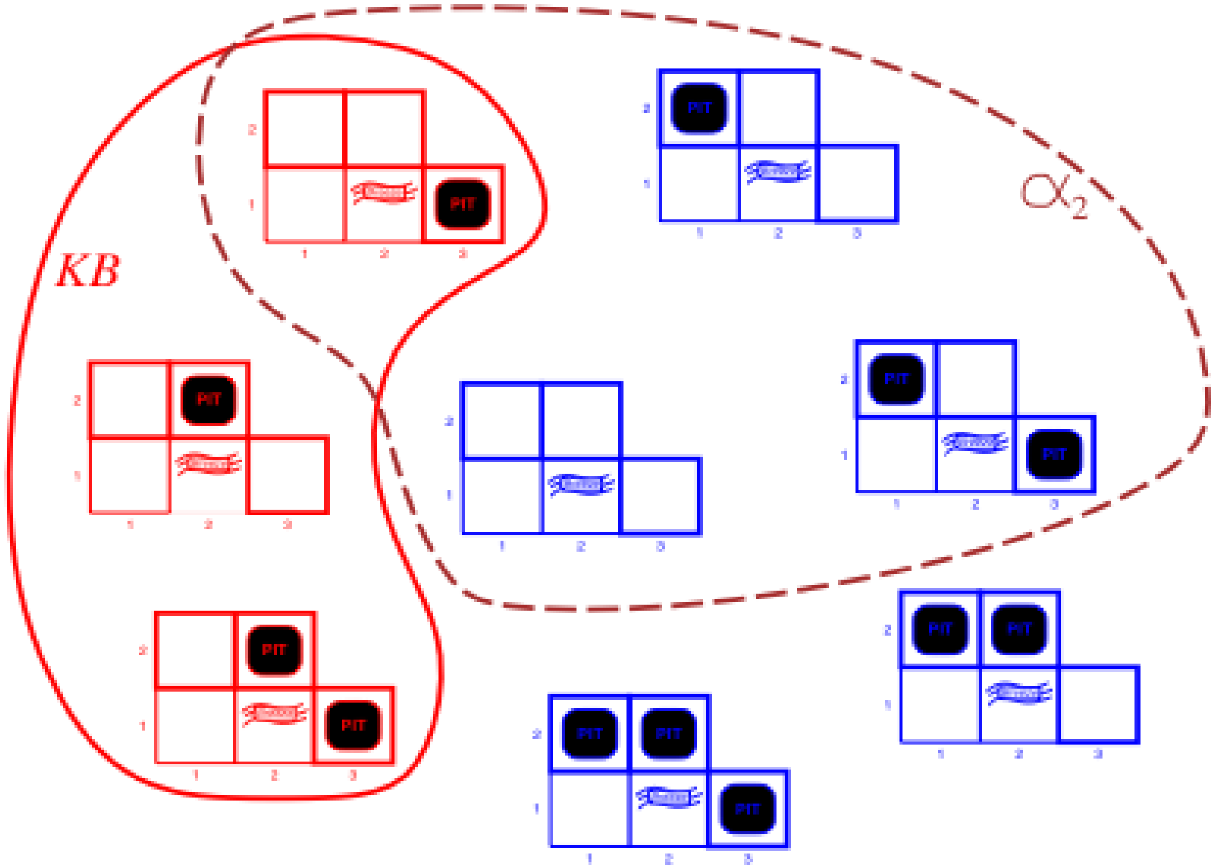
Another Example

Is [2,2] safe?



KB = wumpus-world rules + observations

Another Example



M(KB) $\nsubseteq$ ***M(***α_2***)***

KB = wumpus-world rules + observations

α_2 = "[2,2] is safe", $\boxed{KB \not\models \alpha_2}$

Inference Algorithms: Soundness and Completeness

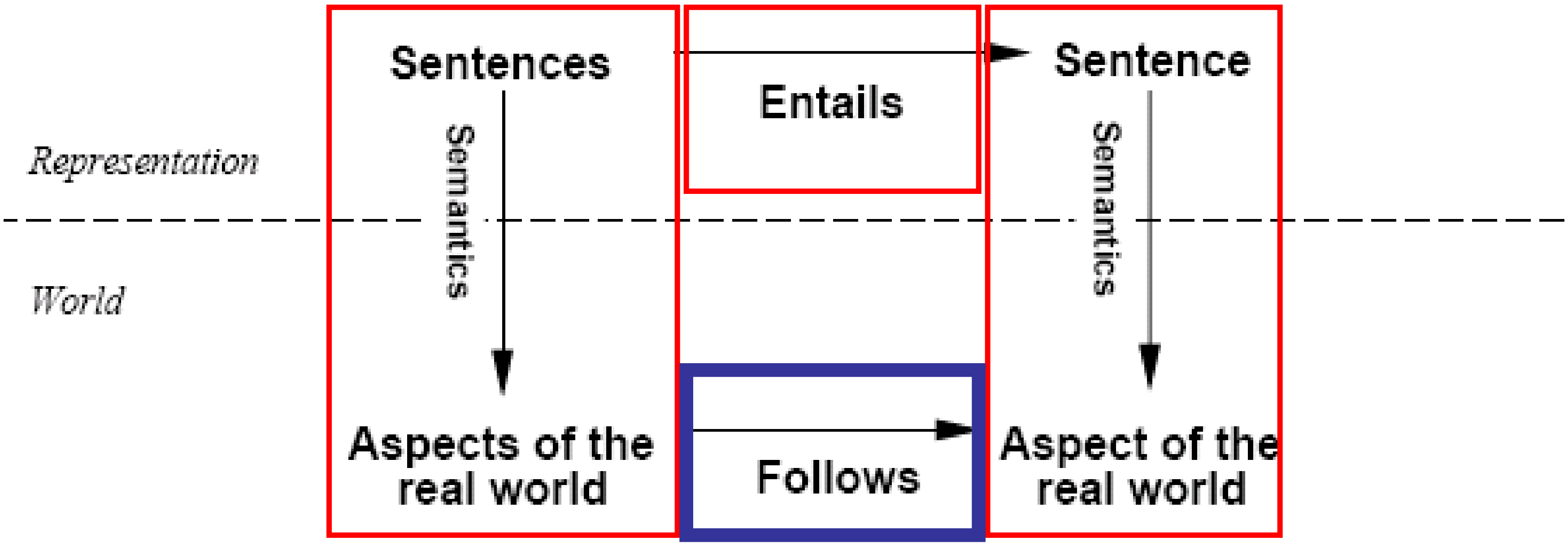
If an inference algorithm only derives entailed sentences, it is called *sound* (or *truth preserving*).

- Otherwise it just makes things up
- *Algorithm i is sound if whenever KB* $\vdash_i \alpha$ *(i.e.* α *is derived by i from KB) it is also true that KB* $\models \alpha$

Completeness: An algorithm is complete if it can derive any sentence that is entailed.

i is complete if whenever KB $\models \alpha$ ***it is also true that KB*** $\vdash_i \alpha$

Relating to the Real World



If a KB is true in the real world, then any sentence α derived from the KB by a sound inference procedure is also true in the real world

Propositional Logic: Syntax

Propositional logic is the simplest logic – illustrates basic ideas

Atomic sentences = proposition symbols = A, B, $P_{1,2}$, $P_{2,2}$ etc. used to denote properties of the world

- *Can be either True or False*

E.g. $P_{1,2}$ = "There's a pit in location [1,2]" is either true or false in the wumpus world

Propositional Logic: Syntax

Complex sentences constructed from simpler ones recursively using logical operators

If S is a sentence, $\neg S$ is a sentence (negation)

If S_1 and S_2 are sentences, $S_1 \wedge S_2$ is a sentence (conjunction)

If S_1 and S_2 are sentences, $S_1 \vee S_2$ is a sentence (disjunction)

If S_1 and S_2 are sentences, $S_1 \Rightarrow S_2$ is a sentence (implication)

If S_1 and S_2 are sentences, $S_1 \Leftrightarrow S_2$ is a sentence (biconditional)

Propositional Logic: Semantics

A model specifies true/false for each proposition symbol

E.g.	$P_{1,2}$	$P_{2,2}$	$P_{3,1}$
	false	true	false

Rules for evaluating truth w.r.t. a model *m*:

$\neg S$ is true iff S is false
$S_1 \wedge S_2$ is true iff S_1 is true and S_2 is true
$S_1 \vee S_2$ is true iff S_1 is true or S_2 is true
$S_1 \Rightarrow S_2$ is true iff S_1 is false or S_2 is true
$S_1 \Leftrightarrow S_2$ is true iff both $S_1 \Rightarrow S_2$ and $S_2 \Rightarrow S_1$ are true

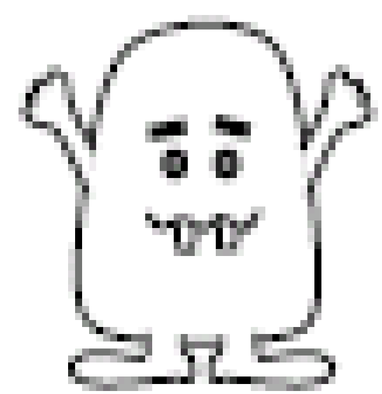
Truth Tables for Connectives

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
$false$	$false$	$true$	$false$	$false$	$true$	$true$
$false$	$true$	$true$	$false$	$true$	$true$	$false$
$true$	$false$	$false$	$false$	$true$	$false$	$false$
$true$	$true$	$false$	$true$	$true$	$true$	$true$

Pssst…here's a useful tip:

$P \Rightarrow Q$ is equivalent to $\neg P \vee Q$

(prove it!)



Propositional Logic: Semantics

Simple recursive process can be used to evaluate an arbitrary sentence

E.g., Model:

$P_{1,2}$	$P_{2,2}$	$P_{3,1}$
false	true	false

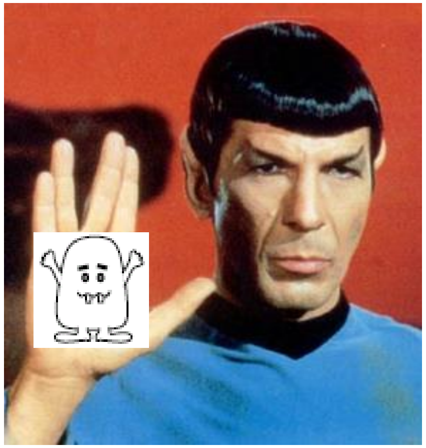
$\neg P_{1,2} \wedge (P_{2,2} \vee P_{3,1})$

$= \textit{true} \wedge (\textit{true} \vee \textit{false})$

$= \textit{true} \wedge \textit{true}$

$= \textit{true}$

Next Time: Inference using Propositional Logic



To Do:

Project #2 (Multi-Agent PacMan) assigned today!

Read Chapter 7 in textbook