

# Vorlesung

# Grundlagen der

# Künstlichen Intelligenz

Reinhard Lafrenz / Prof. A. Knoll

Robotics and Embedded Systems  
Department of Informatics – I6  
Technische Universität München

[www6.in.tum.de](http://www6.in.tum.de)

[lafrenz@in.tum.de](mailto:lafrenz@in.tum.de)

089-289-18136

Room 03.07.055



Wintersemester 2012/13

19.11.2012

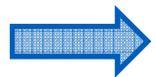


## Chapter 7 (3rd ed.)

# Logical Agents

## Questions:

- How was the knowledge represented so far?
- What does a (searching) agent know about the world?
- How is this knowledge applied?
- Where does this knowledge come from?
- Can this knowledge be updated?



Implicit vs. explicit representation



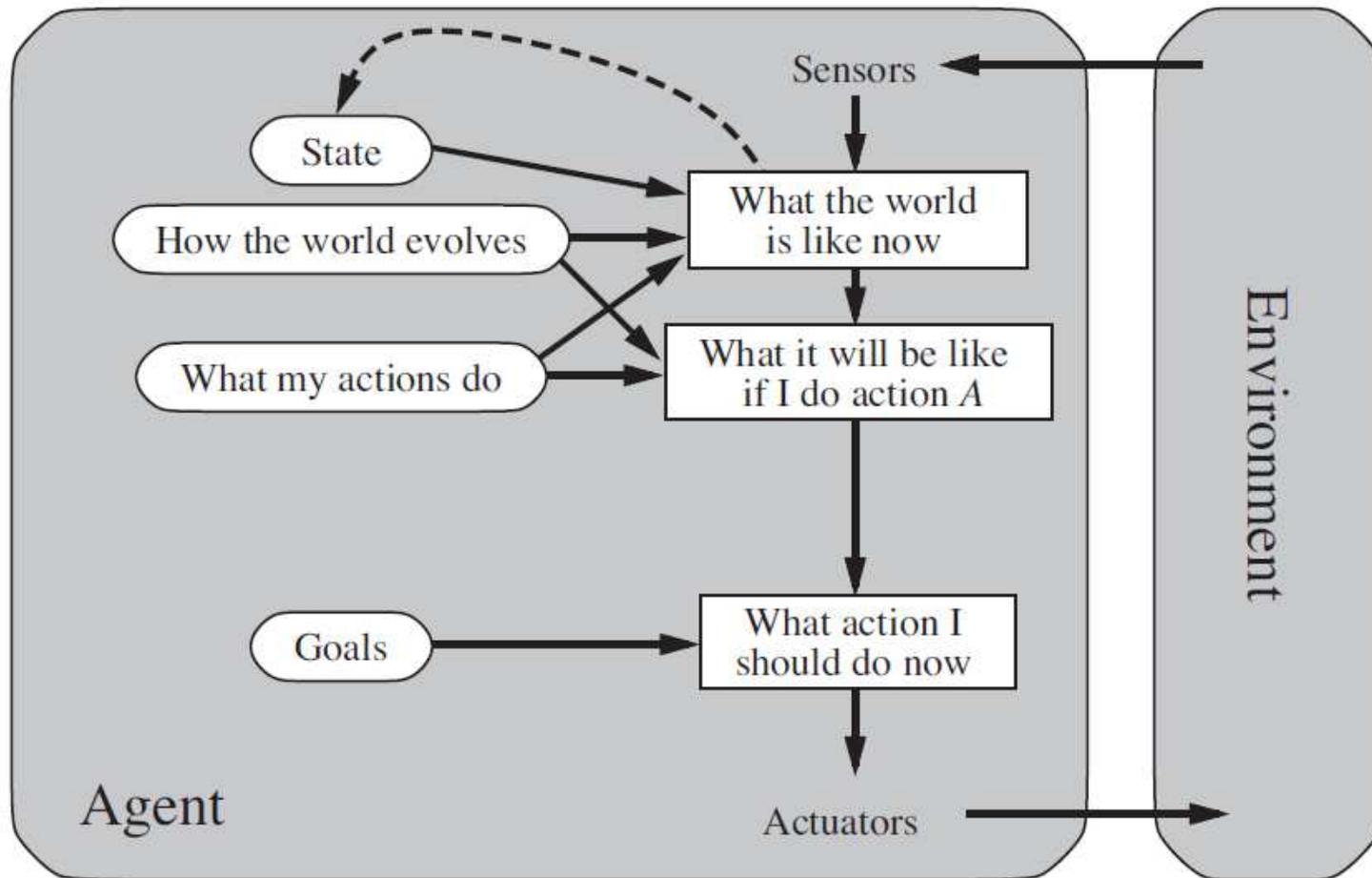
# Logical agents

- Knowledge-based agents with internal representation of knowledge
- Reasoning process to gain new knowledge and to draw conclusions
- Representation schemes (languages)
- A knowledge base is a collection of (formal) sentences
  
- Two operators on the knowledge base:
  - $\text{TELL}(KB, \textit{sentence})$
  - $\text{ASK}(KB, \textit{sentence})$



# Logical agents

- Reconsider a general agent scheme (see also chapter 2)



# Logical agents

- Two operators on the knowledge base:
  - $\text{TELL}(KB, \textit{sentence})$
  - $\text{ASK}(KB, \textit{sentence})$



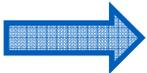
# Logical agent: operation principle

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

- The KB contains initially ( $t=0$ ) the **background knowledge**
- Choice of abstraction level is essential
  - E.g. for finding the way from A to B, no implementational details are needed



# Declarative vs. procedural knowlege

- There where intensive discussions about the better form of representation, explicitely or implicitely
- In real applications, both are useful  
 hybrid approaches
- Declarative knowledge can often be automatically compiled into procedural code



# Wumpus world

- Single agent in a 4x4 grid (cave)
- There is a beast called Wumpus
- ... and there are pits
- The agent has a bow and a single arrow
- It can turn left or right and move forward 1 field
  
- Goal: Search for gold in the cave and escape from the cave
  
- Simple “game“ used to illustrate main concepts of AI



# Wumpus world

## PEAS description (cont'd)

### Performance:

- +1000 for escaping with the gold from the cave
- -1000 for being „eaten“ by Wumpus or falling in a pit
- -1 for each motion
- -10 for shooting the arrow

### ■ Environment

- The grid with start position at [1,1]
- Gold and Wumpus are randomly distributed among the other fields
- Pits with probability 0.2 on each other field



# Wumpus world

## PEAS description

(Performance, Environment, Actuators, Sensors)

### ■ Actuators:

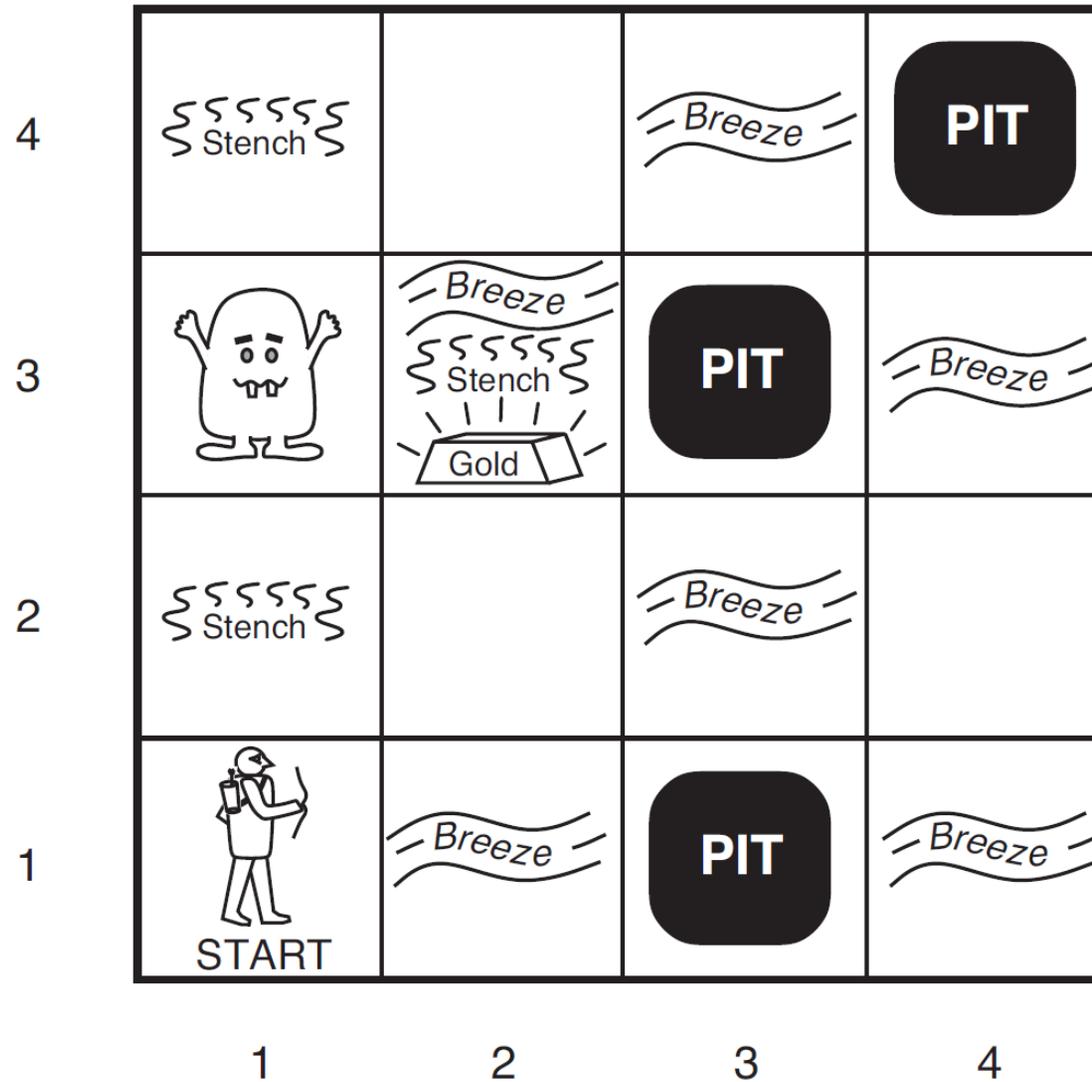
- Forward, turns left, right (90 deg.)
- Grab (to get the gold)
- Shoot (shoot arrow in view direction)
- Climb (leave cave, only possible on field [1,1])

### ■ Sensors

- At location of the Wumpus + the four grid-neighbors the agent senses a *stench*
- At the four grid-neighbors of a pit the agent senses a *breeze*
- At the field where the gold is located, the agent senses a *glitter*
- If the agent runs into a wall, he senses a bump
- If the agent kills the Wumpus, a scream can be heard in the cave



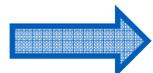
# Wumpus world



# Wumpus world: The environment

## Classification of the environment

- Discrete, static, single agent
- Partially observable (only sensing on the current field)
- Sequential (rewards several time steps after an action)
  
- Agent not able to “win“ in all situations (~ 21% unfair)
  - Gold may be in a room with a pit
  - No way to gold without pits

 Decision wheter to risk the life



# Wumpus world: Simple graphical representation

Sensor percepts described as a vector  
 [Stench, Breeze, Glitter, Bump, Scream]

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK			
1,1 A OK	2,1 OK	3,1	4,1

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

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2 P?	3,2	4,2
OK			
1,1 V OK	2,1 A B OK	3,1 P?	4,1

(a)

[None, None, None, None, None]

(b)

[None, Breeze, None, None, None]



# Wumpus world: Simple graphical representation

## Later steps

1,4	2,4	3,4	4,4
1,3 W!	2,3	3,3	4,3
1,2 A S OK	2,2 OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

(a)

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

1,4	2,4 P?	3,4	4,4
1,3 W!	2,3 A S G B	3,3 P?	4,3
1,2 S V OK	2,2 V OK	3,2	4,2
1,1 V OK	2,1 B V OK	3,1 P!	4,1

(b)

After 3rd move with percept  
[Stench, None, None, None, None]

After 5th move with percept  
[Stench, Breeze, Glitter, None, None]



# Wumpus world: Reasoning

- Background Knowledge
  - Agent is at [1,1]
  - [1,1] is safe “OK“
  - All perception and action rules
- Gained Knowledge
  - States of fields visited and ist neighbours
- Reasoning based on logical **entailments**

Conclusions drawn from available knowledge (background + percepts) are correct!



# Logic

- Knowledge base
  - Consists of **sentences** in the **syntax** of the **representation language**
- Syntax
  - **Well-formed** sentences/formulas
  - E.g. “ $x+y = 4$ ” vs. “ $x4y+=$ ”
- Semantics
  - **Meaning** of a sentence
  - **Truth** in view of each **possible world**
  - In classical logic, each sentence is either true or false in a world
- **Model** is a more precise term for “possible world”



# Model

- A model can be seen as a variable assignment
  - E.g. in  $x+y=4$  each assignment of  $[x,y]$  is a model
  - How many models are possible?
- If a sentence  $\alpha$  is true in a model  $m$  then we say  $m$  satisfies  $\alpha$  or  $\alpha$  is a model of  $m$
- $M(\alpha)$  is the set of all models of  $\alpha$ 
  - i.e., all variable assignments where  $\alpha$  is true



# Entailment (logical consequence)

A sentence  $\beta$  is a logical consequence of sentence  $\alpha$

$$\alpha \models \beta$$

This means that in each model where  $\alpha$  is true,  $\beta$  is also true

In terms of sets

$$\alpha \models \beta \quad \text{iff} \quad M(\alpha) \subseteq M(\beta) \quad (\text{iff: if and only if})$$

Or, in other words, the statement  $\alpha$  is more strict than the statement  $\beta$

Example from maths:  $x=0 \models xy=0$



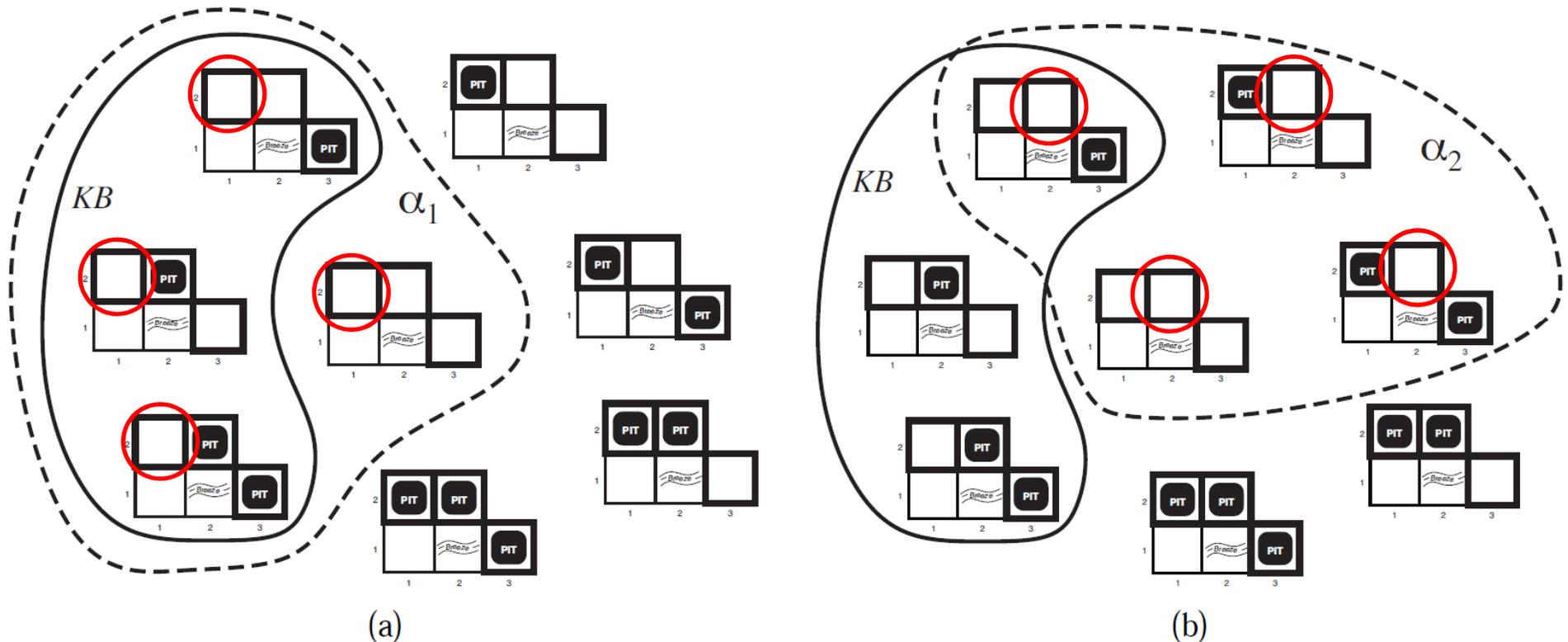
## Models in the Wumpus world

- Assumption: Only pits are of interest at the moment
- Agent perceives nothing in [1,1], moves to [2,1] and perceives there a breeze.
- Question: are pits in [1,2], [2,2], [3,1]  
 $2^3 = 8$  possible worlds (models)
- Note, that there is no statement about truth when identifying all possible worlds, it's just the collection of all variable assignments.



# Models in the Wumpus world

Solid line: models consistent with first 2 percepts, i.e. the KB is true, or each sentence in the KB is true



(a)  $\alpha_1$  : No pit at [1,2]

(b)  $\alpha_2$  : no pit at [2,2]



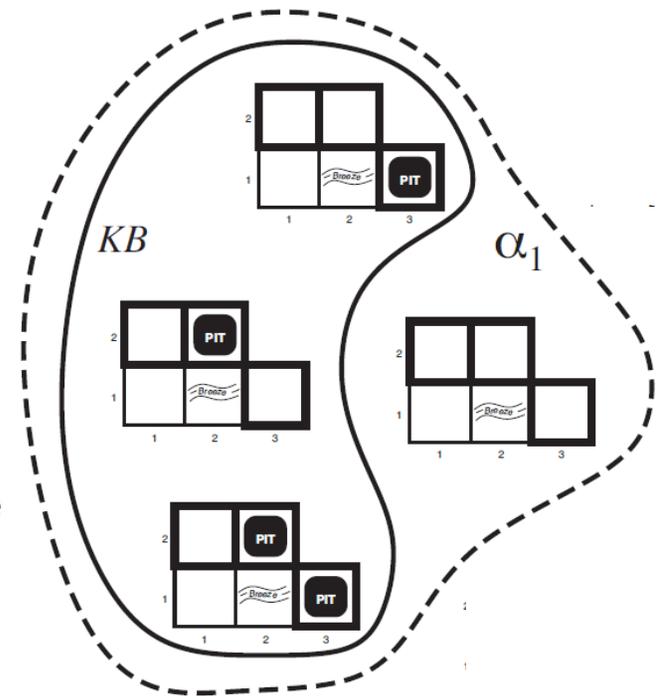
# Models in the Wumpus world

In each model, where KB is true,  $\alpha_1$  is also true

$KB \models \alpha_1$  (no pit at [1,2])

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

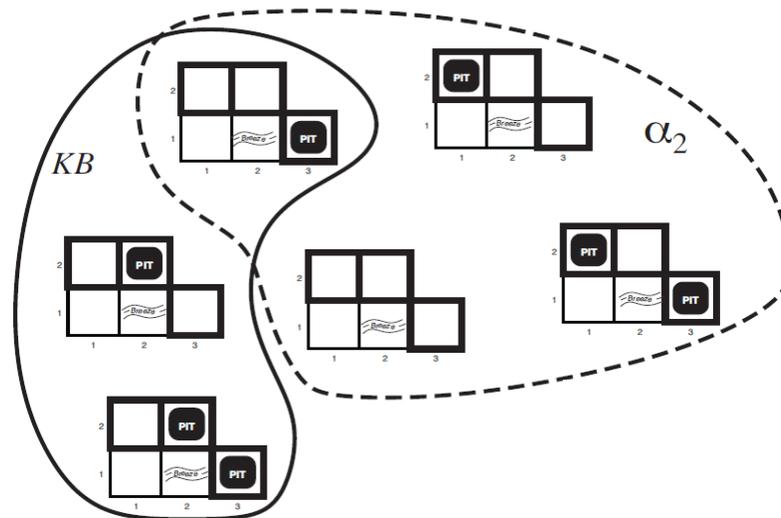
**Model checking**, test whether  $\alpha$  is true for all models where KB is true



# Models in the Wumpus world

On the other hand, in some of the models where KB is true,  $\alpha_2$  is false, i.e.  $KB \not\models \alpha_2$

The agent is not able to conclude, that there is *no* pit at [2,2], but also not able to conclude that there *is* a pit at [2,2]



# Formal inference

If an inference algorithm  $i$  is able to derive  $\alpha$  from KB,  
we write  $\text{KB} \vdash_i \alpha$

## Soundness / Correctness

- Property of maintaining truth
- Only derivable sentences are derived

## Completeness

- The algorithm is able to derive *all* derivable sentences
- Compare with completeness of search algorithms,  
e.g. depth first search



# Formal inference

If an inference algorithm  $i$  is able to derive  $\alpha$  from KB,  
we write  $\text{KB} \vdash_i \alpha$

## Soundness / Correctness

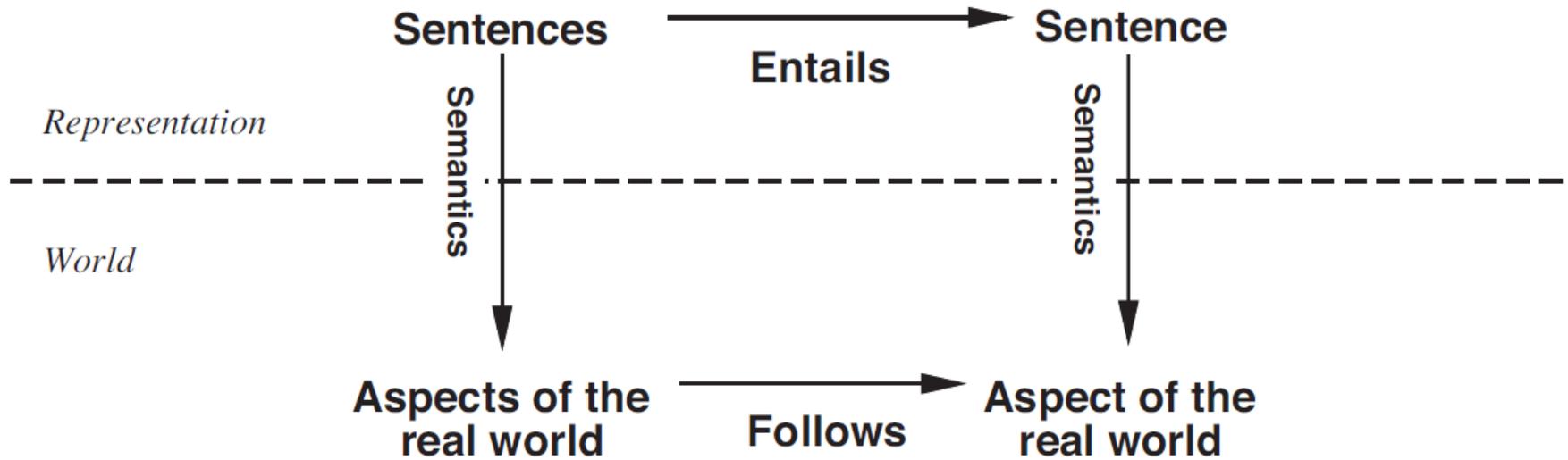
- Property of maintaining truth
- Only derivable sentences are derived

## Completeness

- The algorithm is able to derive *all* derivable sentences
- Compare with completeness of search algorithms,  
e.g. depth first search



# Syntax and semantics



If a KB is true in the real world (and the inference is correct), then the derived sentences are also true in the real world



# Symbol grounding

How do we know that a KB is true in the real world?

- Perceptions tell us the truth
- We assume that the a priori knowledge is also true

Complex problem in real applications



# Propositional logic (PL)

Syntax:

- Atomic sentences: symbols TRUE, FALSE, variables
- Composite sentences with logical operators
- Grammar in Backus-Naur-Form (BNF)

Sentence  $\rightarrow$  Atomic Sentence | Complex Sentence

Atomic Sentence  $\rightarrow$  TRUE | FALSE | P | Q | R | ...

Complex Sentence  $\rightarrow$  (Sentence) | [Sentence] |

$\neg$  Sentence |

(Sentence  $\wedge$  Sentence |

(Sentence  $\vee$  Sentence) |

(Sentence  $\Rightarrow$  Sentence) |

(Sentence  $\Leftrightarrow$  Sentence)

Operator sequence:  $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$



# Propositional logic (PL)

Semantics:

- Truth values for each symbol  $\sim$   $\rightarrow$  model
- $\neg P$  is true **iff**  $P$  is false in  $m$
- $P \wedge Q$  is true **iff**  $P$  and  $Q$  are true in  $m$
- ...

How many functions for 2 variables are possible?

2 variables  $\sim$   $\rightarrow$  4 states

4 states individually evaluating to true, false  $\sim$   $\rightarrow$   $2^4=16$  possible functions



# Propositional logic (PL)

Truth tables for some functions

<b>P</b>	<b>Q</b>	<b><math>\neg P</math></b>	<b><math>P \wedge Q</math></b>	<b><math>P \vee Q</math></b>	<b><math>P \Rightarrow Q</math></b>	<b><math>P \Leftrightarrow Q</math></b>
<i>false</i>	<i>false</i>	<i>true</i>	<i>false</i>	<i>false</i>	<i>true</i>	<i>true</i>
<i>false</i>	<i>true</i>	<i>true</i>	<i>false</i>	<i>true</i>	<i>true</i>	<i>false</i>
<i>true</i>	<i>false</i>	<i>false</i>	<i>false</i>	<i>true</i>	<i>false</i>	<i>false</i>
<i>true</i>	<i>true</i>	<i>false</i>	<i>true</i>	<i>true</i>	<i>true</i>	<i>true</i>



# A simple KB for the Wumpus world

## Rules

- $P_{x,y}$  is true if there is a pit at  $[x,y]$
- $W_{x,y}$  is true if Wumpus is at  $[x,y]$
- $B_{x,y}$  is true if the agent feels a breeze at  $[x,y]$
- $S_{x,y}$  is true if the agent perceives a stench at  $[x,y]$

## Facts:

- $R_1: \neg P_{1,1}$  There is no pit at  $[1,1]$
- $R_2: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$
- $R_3: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,1} \vee P_{3,1})$

These facts are true in all worlds



## KB after the first 2 perceptions

- $R_4: \neg B_{1,1}$  There is no breeze at [1,1]
- $R_5: B_{2,1}$
- How to answer to specific questions, such as  $KB \models P_{2,2}$  ?
- Simple inference mechanism



# Truth-table based reasoning

KB  $\models$   $P_{2,2}$  ?

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	$R_1$	$R_2$	$R_3$	$R_4$	$R_5$	KB
false	true	true	true	true	false	false						
false	false	false	false	false	false	true	true	true	true	true	false	false
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
false	true	false	false	false	false	false	true	true	false	true	true	false
false	true	false	false	false	false	true	true	true	true	true	true	<u>true</u>
false	true	false	false	false	true	false	true	true	true	true	true	<u>true</u>
false	true	false	false	false	true	true	true	true	true	true	true	<u>true</u>
false	true	false	false	true	false	false	true	false	false	true	true	false
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
true	false	true	true	false	true	false						



# Summary

- Logics as formal representation for knowledge
- Syntax and semantics
- Models (possible worlds)

