Artificial Intelligence

Complete architectures for intelligence?

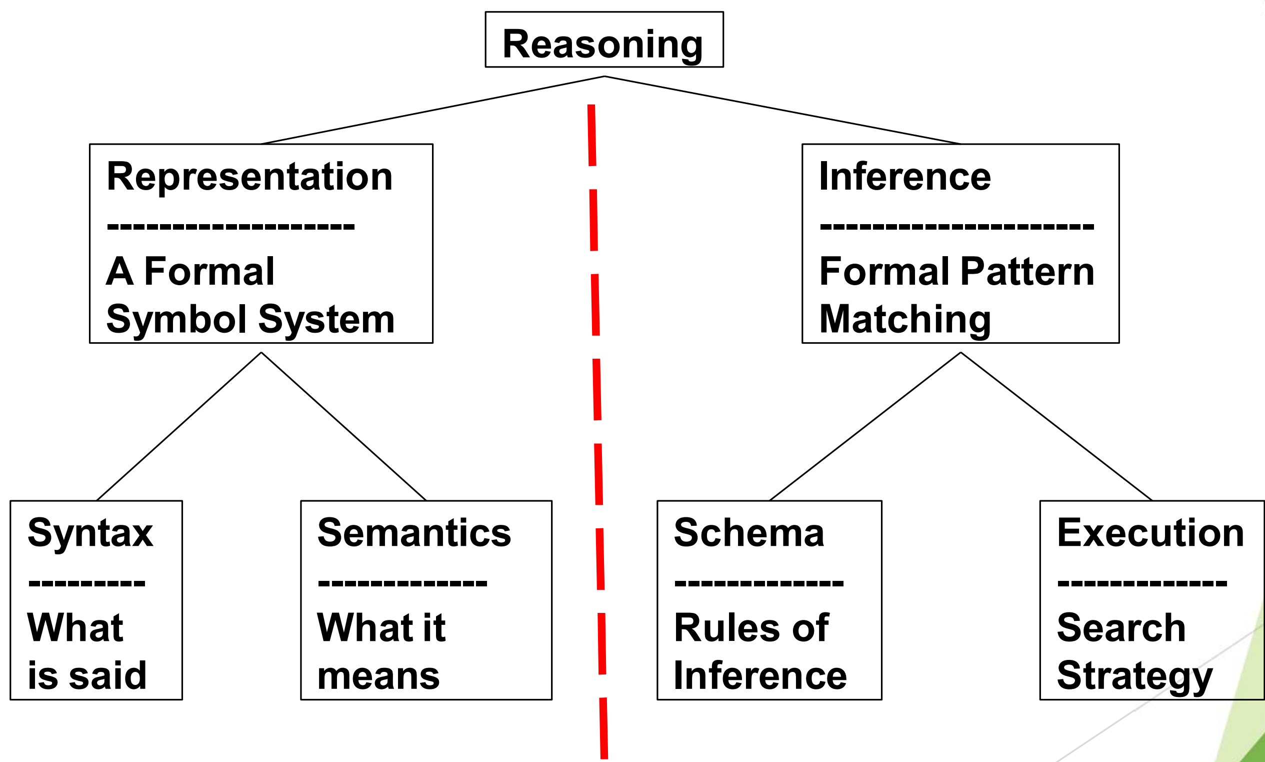
- <u>Search?</u>
  - Solve the problem of what to do.
- <u>Logic and inference?</u>
  - Reason about what to do.
  - Encoded knowledge/“expert” systems?
    - Know what to do.
- <u>Learning?</u>
  - Learn what to do.
- <u>Modern view: It’s complex & multi-faceted.</u>

Inference in Formal Symbol Systems: Ontology, Representation, Inference

- **Formal Symbol Systems**
  - **Symbols** correspond to **things/ideas** in the world
  - **Pattern matching & rewrite** corresponds to **inference**

- **Ontology:** What exists in the world?
  - → What must be represented?
- **Representation:** Syntax vs. Semantics
  - → What's Said vs. What's Meant
    - **Inference:** Schema vs. Mechanism
      - → Proof Steps vs. Search Strategy

Ontology:
What kind of things exist in the world?
What do we need to describe and reason about?



Why Do We Need Logic?

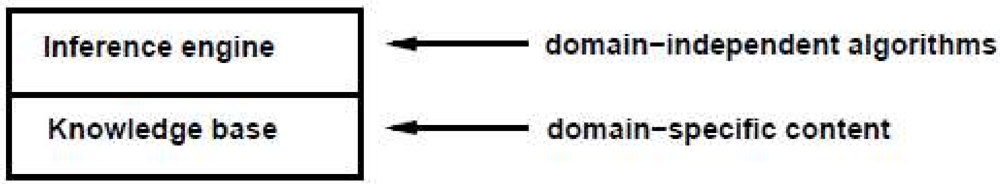
- Problem-solving agents were very inflexible: hard code every possible state.
- Search is almost always exponential in the number of states.
- Problem solving agents cannot infer unobserved information.
- We want an algorithm that reasons in a way that resembles reasoning in humans

Logical or Knowledge based Agents

- Basic Actions: Tell and Ask
- A Knowledge base keeps track of things
- We can tell an agent facts and ask for inference

- Example:
  - Tell: Father of John is Bob
  - Tell: Jane is John’s Sister
  - Tell: John’s father is the same as John’s sister father
  - Ask: Who is Jane’s father? (The answer requires inference on facts)

Knowledge bases



Knowledge base = set of sentences in a formal language

Declarative approach to building an agent (or other system):

  TELL it what it needs to know

Then it can ASK itself what to do—answers should follow from the KB

Agents can be viewed at the knowledge level

  i.e., what they know, regardless of how implemented

Or at the implementation level

  i.e., data structures in KB and algorithms that manipulate them

Components

- Knowledge base / KB (facts)
- Knowledge Representation Language (In what language would you tell agents the facts?)
- Inference
- Background Knowledge of the world

A simple knowledge-based agent

```
function KB-AGENT(percept) returns an action
    static: 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
```

The agent must be able to:

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

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

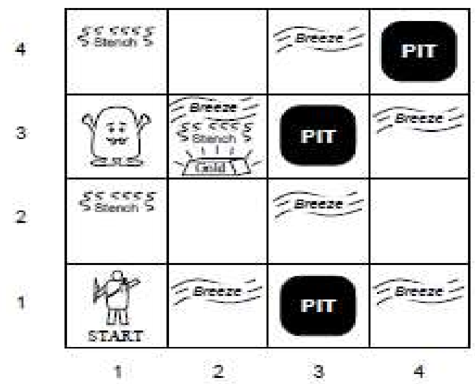
Glitter iff gold is in the same square

Shooting kills wumpus if you are facing it

Shooting uses up the only arrow

Grabbing picks up gold if in same square

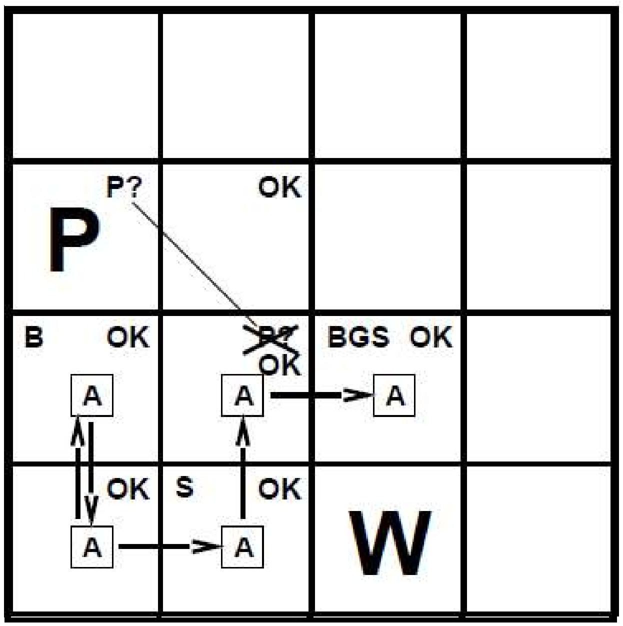
Releasing drops the gold in same square



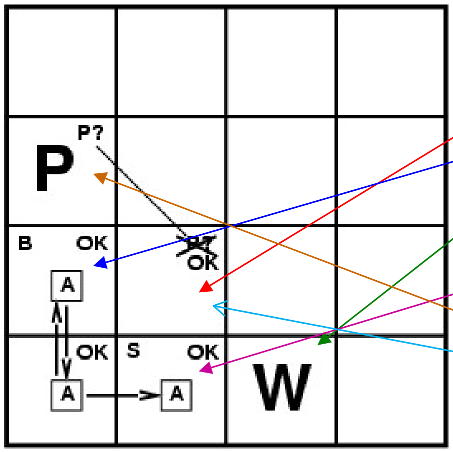
Actuators Left turn, Right turn,
Forward, Grab, Release, Shoot

Sensors Breeze, Glitter, Smell

Exploring a wumpus world



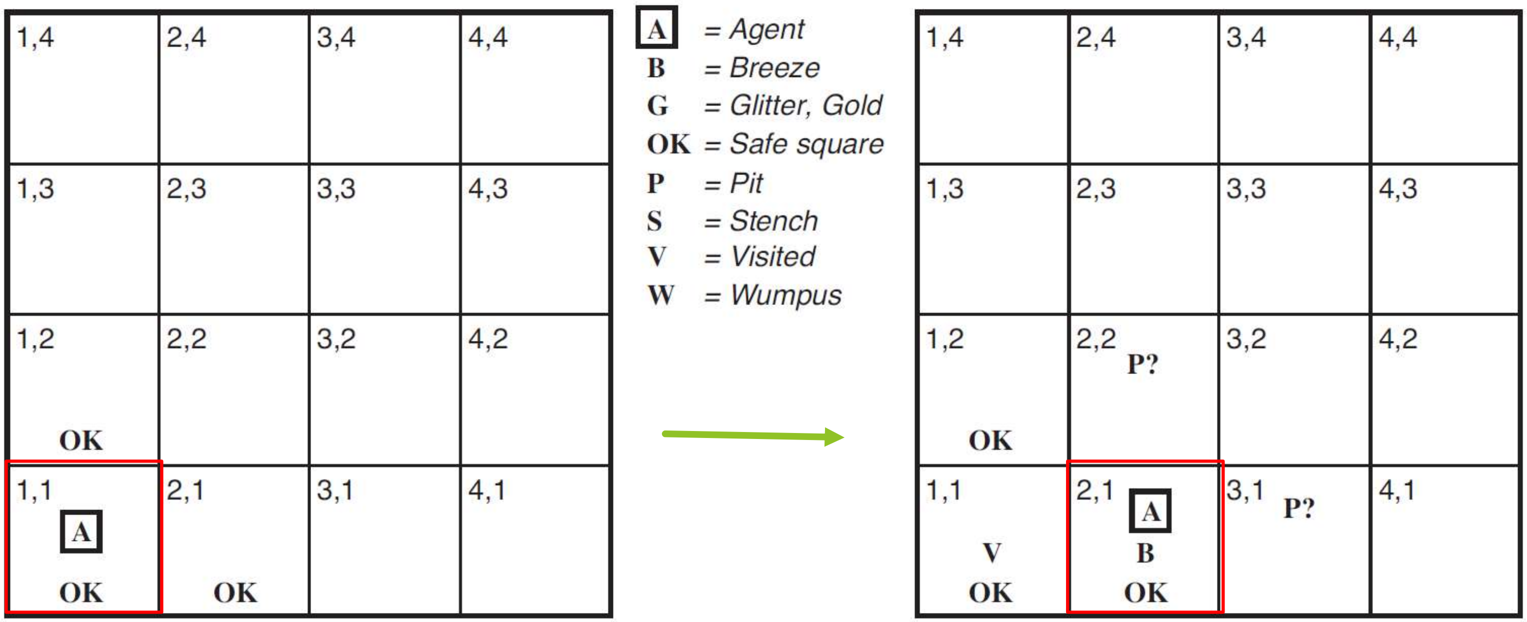
Exploring a Wumpus world



If the Wumpus were here, stench should be here. Therefore it is here.
Since, there is no breeze here, the pit must be there, and it must be OK here

We need rather sophisticated reasoning here!

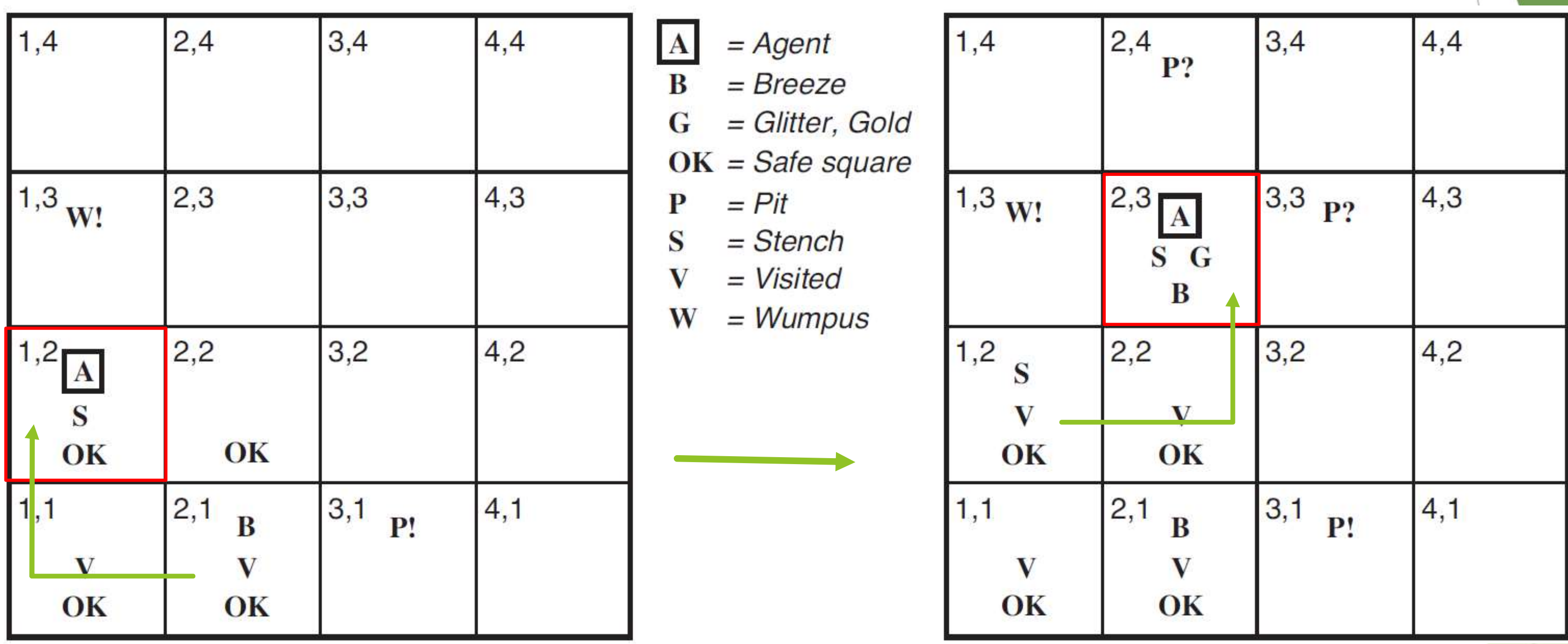
Wumpus World



[None, None, None, None, None]

[None, Breeze, None, None, None]

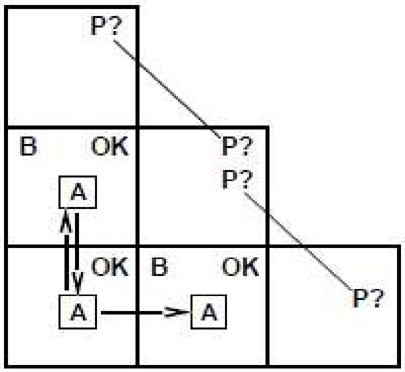
Wumpus World



[Stench, None, None, None, None]

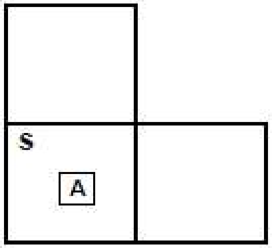
[Stench, Breeze, Glitter, None, None]

Other tight spots



Breeze in (1,2) and (2,1)
$\Rightarrow$ no safe actions

Assuming pits uniformly distributed,
(2,2) has pit w/ prob 0.86, vs. 0.31



Smell in (1,1)
$\Rightarrow$ cannot move
Can use a strategy of coercion:
shoot straight ahead
wumpus was there $\Rightarrow$ dead $\Rightarrow$ safe
wumpus wasn't there $\Rightarrow$ safe

Logic

- We used logical reasoning to find the gold.
- Logics are formal languages for representing information such that conclusions can be drawn from formal inference patterns
- Syntax defines the well-formed sentences in the language
- Semantics define the "meaning" or interpretation of sentences:
  - connect symbols to real events in the world
  - i.e., define truth of a sentence in a world

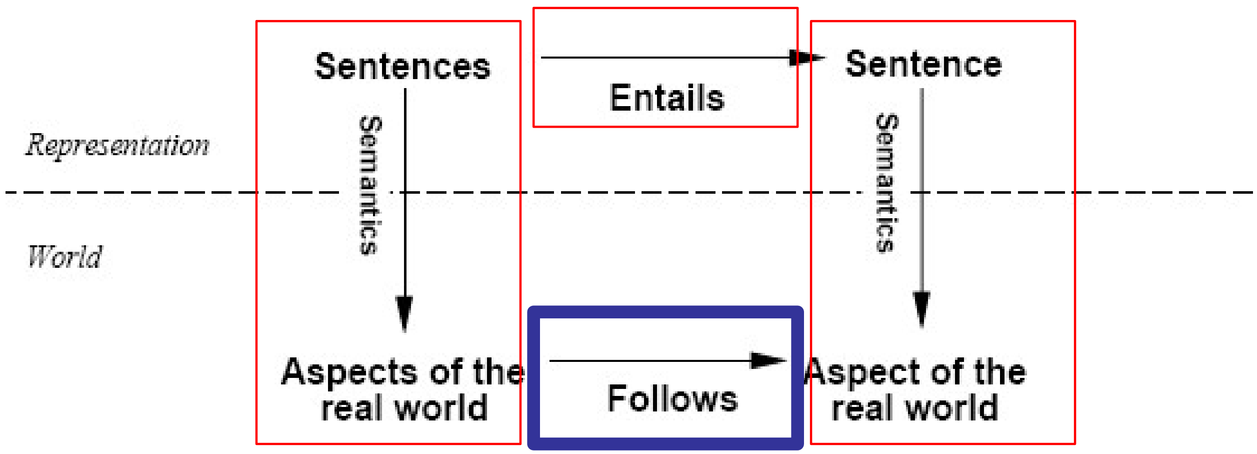
- E.g., the language of arithmetic:
  - $x+2 \geq y$ is a sentence
  - $-x2+y > \{\}$ is not a sentence

  → syntax

- $x+2 \geq y$ is true in a world where $x = 7$, $y = 1$
- $x+2 \geq y$ is false in a world where $x = 0$, $y = 6$

→ semantics

Schematic perspective



If KB is true in the real world,
then any sentence α entailed by KB
is also true in the real world.

For example: If I tell you (1) Sue is Mary's sister, and (2) Sue is Amy's mother, then it **necessarily follows in the world** that Mary is Amy's aunt, even though I told you nothing at all about aunts. This sort of reasoning pattern is what we hope to capture.

Semantics

- Knowledge bases consist of sentences.
- E.g., “x + y = 4” is a well-formed sentence, whereas “x4y+ =” is not.
- A logic must also define the semantics or meaning of sentences.
- The semantics defines the **truth** of each sentence with respect to each possible world or a model. For example, the semantics for arithmetic specifies that the sentence “x + y =4” is true in a world where x is 2 and y is 2, but false in a world where x is 1 and y is 1.
- In standard logics, every sentence must be either true or false in each possible world—there is no in between.

Examples

- EXAMPLES. The following are propositions:
  - – the reactor is on;
  - – the wing-flaps are up;
  - – John Major is prime minister.

whereas the following are not:

– are you going out somewhere?

– 2+3

Exercises

- If Edith eats her vegetables, then she can have a cookie. Edith ate her vegetables. Therefore Edith gets a cookie.

$$\begin{array}{rc} & P \to Q \\ & P \\ \hline \therefore & Q \end{array}$$

- "If it's your birthday or there will be cake, then there will be cake."

Create a truth table

- P:P: it's your birthday; Q:Q: there will be cake. $(P \vee Q) \to Q$

Models

- For $x^2 + y^2 = 5$, one possible model that satisfies the equation/sentence is m1 = (x,y) = (2,1), another model that does so is m2 = (1,2).
- If a sentence α is true in model m, we say that m satisfies α or sometimes m is a model of α.
- We use the notation M(α) to mean the set of all models of α i.e., M(α) ={m1(α) ,m2(α) ,….} where all models m1,m2,…. satisfy α
- m1(α) $\longrightarrow$ refers to a single solution / model
- M(α) $\longrightarrow$ refers to the set of all possible solutions / models

Logical Entailment

- Logical Reasoning involves the relation of logical entailment between sentences—the idea that a sentence *follows logically* from another sentence

$$\alpha \models \beta \quad \text{(alpha entails beta)}$$

- $\alpha \models \beta$ if and only if, in every model in which alpha is true, beta is also true

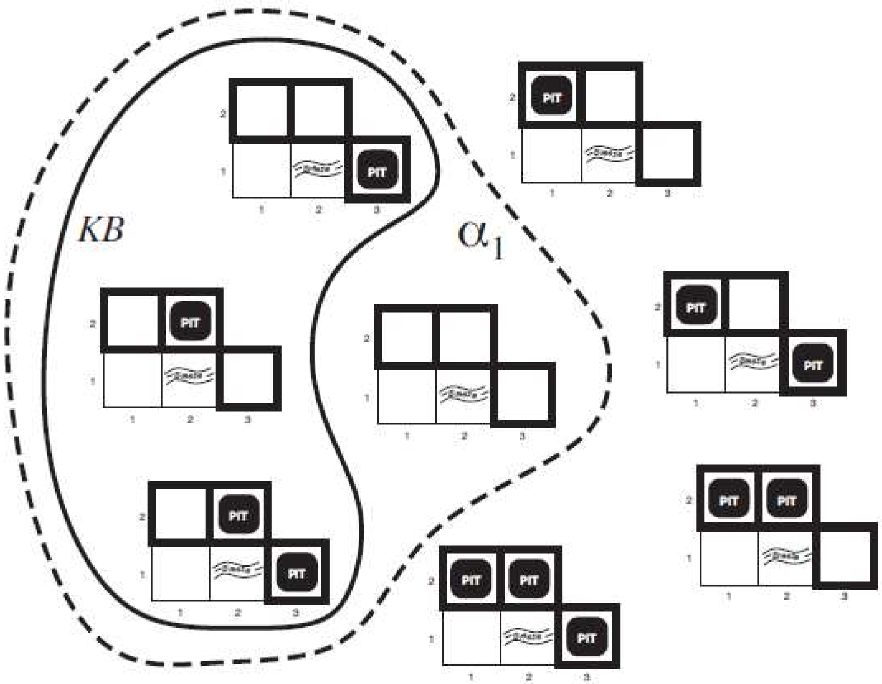
$$\alpha \models \beta \text{ if and only if } M(\alpha) \subseteq M(\beta)$$

Logical Entailment from Math

- Alpha (sentence 1) -> x=0
- Beta (sentence 2) -> xy=0
- There's only one possible solution of x=0 and that is if x is 0 (x=0), therefore M(alpha) = {x=0}
- But there are many possible solutions of sentence 2, i.e., M(beta)={(x,y)=(0,0), (x,y)=(0,1), (x,y)=(1,0), …..} one of which is x=0
- Since all possible solutions of alpha an be found in the solution set of beta, we say that alpha entails beta

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

Wumpus World: Entailment



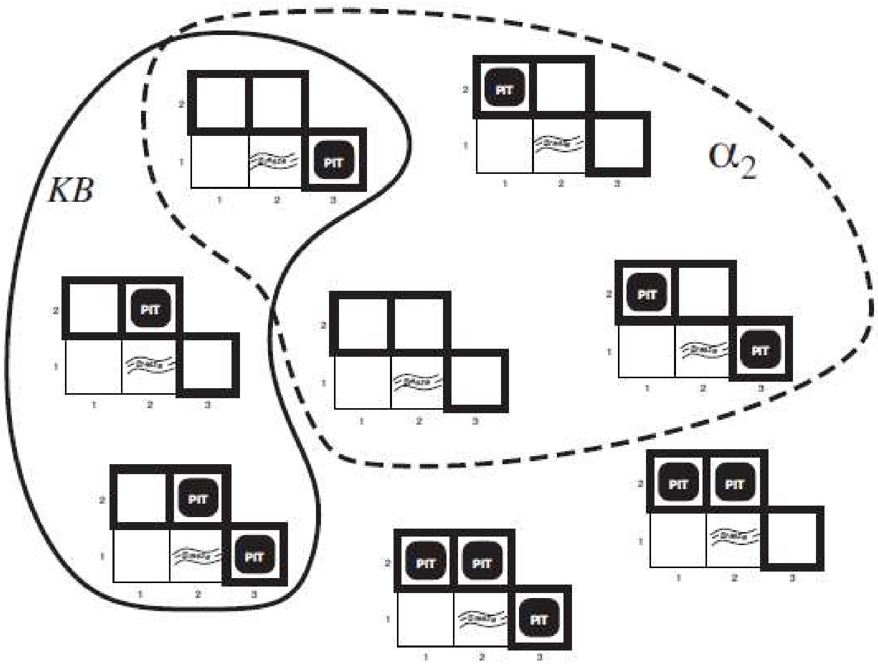
Alpha 1 (sentence) -> no pits in [1, 2]

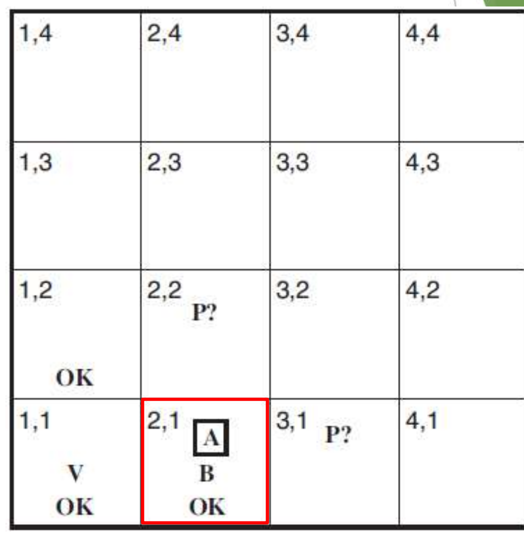
KB |= α_1

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

[None, Breeze, None, None, None]

Wumpus World: Entailment





[None, Breeze, None, None, None]

Alpha 2 (sentence) -> no pits in [2, 2]

$KB \not\models \alpha_2$

Propositional Logic: Syntax

- The atomic sentences consists of a single proposition symbol.
- P, Q, R, $W_{1,3}$ are proposition symbols that evaluate to True or False. $W_{1,3}$ is True if there's a wumpus in (1,3).
- Complex sentences are constructed from simpler sentences using parenthesis and logical connectivities.

$$
\begin{aligned}
AtomicSentence &\rightarrow True \mid False \mid P \mid Q \mid R \mid \ldots \\
ComplexSentence &\rightarrow (\, Sentence \,) \mid [\, Sentence \,]
\end{aligned}
$$

Logical Connectivities

- ¬ (not). A sentence such as $\neg W_{1,3}$ is called the negation of $W_{1,3}$. A literal is either an atomic sentence (a positive literal) or a negated atomic sentence (a negative literal).
- ∧ (and). A sentence whose main connective is ∧, such as $W_{1,3} \wedge P_{3,1}$, is called a conjunction; its parts are the conjuncts. (The ∧ looks like an "A" for "And.")
- ∨ (or). A sentence using ∨, such as $(W_{1,3} \wedge P_{3,1}) \vee W_{2,2}$, is a disjunction of the disjuncts $(W_{1,3} \wedge P_{3,1})$ and $W_{2,2}$. (Historically, the ∨ comes from the Latin "vel," which means "or." For most people, it is easier to remember ∨ as an upside-down ∧.)

Logical Connectivities

- ⇒ (implies). A sentence such as $(W_{1,3} \wedge P_{3,1}) \Rightarrow \neg W_{2,2}$ is called an implication (or conditional). Its premise or antecedent is $(W_{1,3} \wedge P_{3,1})$, and its **conclusion** or **consequent** is $\neg W_{2,2}$. Implications are also known as rules or if–then statements. The implication symbol is sometimes written in other books as $\supset$ or $\rightarrow$.
- ⇔ (if and only if). The sentence $W_{1,3} \Leftrightarrow \neg W_{2,2}$ is a biconditional. Some other books write this as $\equiv$.

Implication vs Biconditional

P ⇒ Q (if P then Q)	P ⇔ Q (P if and only if Q) or Q ⇔ P (Q if and only if P)
If rainy then take umbrella (doesn't mean that if you have taken Umbrella, then it must rain)	If pit in a square then breeze in atleast one of the neighboring square, also if breeze in a square then pit must be in atleast one of the neighboring square
If a bullet hits my head, I'll die (doesn't mean that if I die, the bullet must have hit me)	If you fail a mandatory course, you'll have to retake it, if you're retaking a course then it must be that you have failed it in the past

Operator Precedence

$$
\begin{array}{rcl}
Sentence & \rightarrow & AtomicSentence \mid ComplexSentence \\
AtomicSentence & \rightarrow & True \mid False \mid P \mid Q \mid R \mid \ldots \\
ComplexSentence & \rightarrow & (\ Sentence\ ) \mid [\ Sentence\ ] \\
& \mid & \neg\ Sentence \\
& \mid & Sentence \land Sentence \\
& \mid & Sentence \lor Sentence \\
& \mid & Sentence \ \Rightarrow\ Sentence \\
& \mid & Sentence \ \Leftrightarrow\ Sentence \\
\\
\text{OPERATOR PRECEDENCE} & : & \neg, \land, \lor, \Rightarrow, \Leftrightarrow
\end{array}
$$

Wumpus world sentences

Let $P_{i,j}$ be true if there is a pit in $[i, j]$.
Let $B_{i,j}$ be true if there is a breeze in $[i, j]$.

$$\neg P_{1,1}$$
$$\neg B_{1,1}$$
$$B_{2,1}$$

"Pits cause breezes in adjacent squares"

$$B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$$
$$B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$$

"A square is breezy if and only if there is an adjacent pit"

Semantics

- In propositional logic, a model simply fixes the truth value—true or false—for every proposition symbol e.g., m1 = {$P_{1,2}$ =false, $P_{2,2}$ =false, $P_{3,1}$ =true} says there's no pit in 2,2 and 1,2 but there's one in 3,1.

- $\neg P$ is true iff P is false in m.
- $P \wedge Q$ is true iff both P and Q are true in m.
- $P \vee Q$ is true iff either P or Q is true in m.
- $P \Rightarrow Q$ is true unless P is true and Q is false in m.
- $P \Leftrightarrow Q$ is true iff P and Q are both true or both false in m.

Semantics

		Negation	Conjunction	Disjunction	Implication	Biconditional
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*

Four possible models shown above

Implication

	P	Q	$P \Rightarrow Q$
m1	*false*	*false*	*true*
m2	*false*	*true*	*true*
m3	*true*	*false*	*false*
m4	*true*	*true*	*true*

- $\Rightarrow$ says if P is True; then Q must be True (m4).
- There is no way that P is True and Q being False (m3), given P $\Rightarrow$ Q is True.
- If P is False, then we are making no claim, Q can be either True or False (m1 and m2)

Entailment in Propositional Logic

- (x=0) |= (xy=0)
- (p=True) |= (p ∨ q)
- p=True is True in 2 models both of which are subset of 3 models in which (p ∨ q) is True

p	q	p ∨ q
True	True	True
True	False	True
False	True	True
False	False	False

Entailment in Propositional Logic

- $(p \wedge q) \models (p \vee q)$
- $p \wedge q$ is True in 1 model which is a subset of 3 models in which $(p \vee q)$ is True

p	q	p ∨ q	p ∧ q
True	True	True	True
True	False	True	False
False	True	True	False
False	False	False	False

Entailment in Propositional Logic

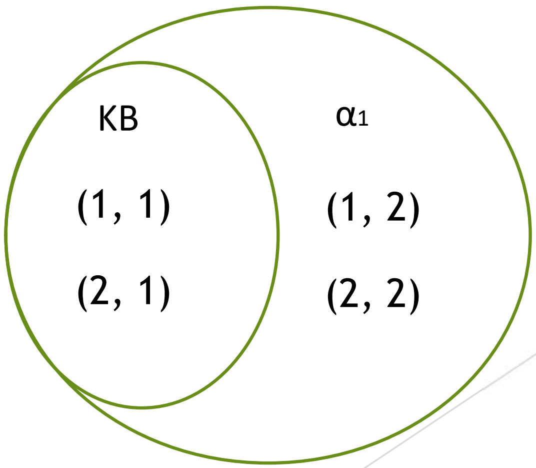
- $((p \Leftrightarrow q) \wedge r) \models (q \Rightarrow p)$
- $((p \Leftrightarrow q) \wedge r)$ is True in 2 models both of which are subset of 6 models in which $(q \Rightarrow p)$ is True

p	q	r	$(p \Leftrightarrow q) \wedge r$	$q \Rightarrow p$
False	False	False	False	True
False	False	True	True	True
False	True	False	False	False
False	True	True	False	False
True	False	False	False	True
True	False	True	False	True
True	True	False	False	True
True	True	True	True	True

Using entailment for answers

- Imagine two variables, cleanliness and dependability (1 for very clean/dependabale; 3 for not at all)
- Knowledge base (contains info about John): I know about John through my friends that he is not messy (1 or 2) and always dependable (1)
- Alpha 1: Good roomates always score 1 or 2 in either cleanliness or dependability.
- Question: Is John a good roommate?

$KB \models \alpha_1$?



Using entailment for answers

- Imagine two variables, cleanliness and dependability (1 for very clean/dependabale; 3 for not at all)
- Knowledge base (contains info about John): I know about John through my friends that he is not messy (1 or 2) and he is not always dependable (2 or 3)
- Alpha 1: Good roomates always score 1 or 2 in either cleanliness or dependability.
- Question: Is John a good roommate?

KB $|= \alpha_1$?