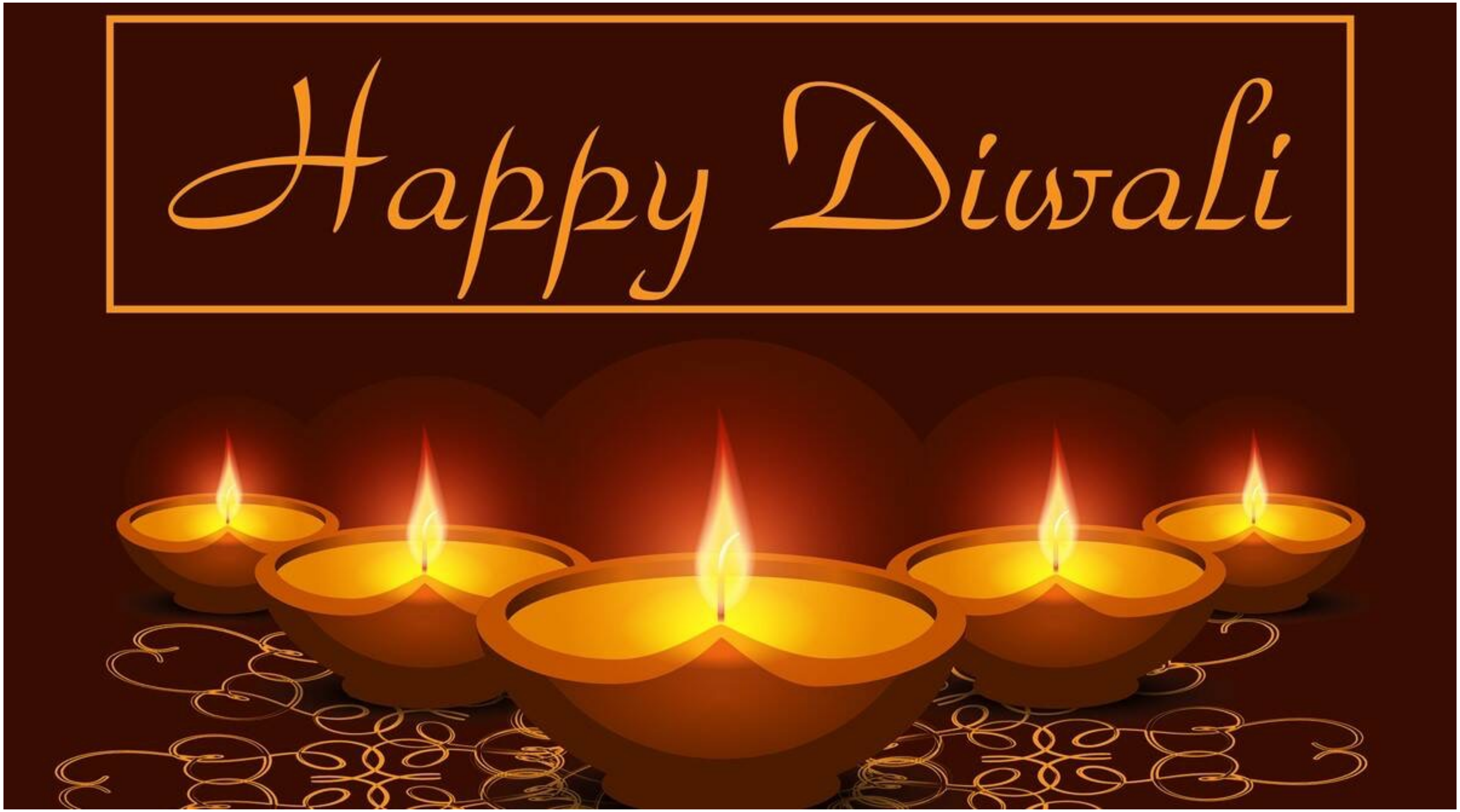
COSC76/276 Artificial Intelligence
Fall 2022
Logical agents

Soroush Vosoughi
Computer Science
Dartmouth College
Soroush@Dartmouth.edu


Happy Diwali

Reminders

- PA3 due tonight at midnight!
- PA4 (due Nov 4th) and SA5 (Due Oct 28th) have been already out

Chess Tournament

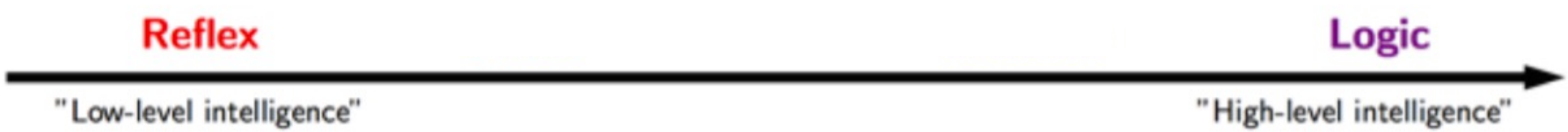
Recap:

- Agents reason using states
- States represent set of possible worlds
- Many possible worlds -> large belief space

Today’s learning objectives

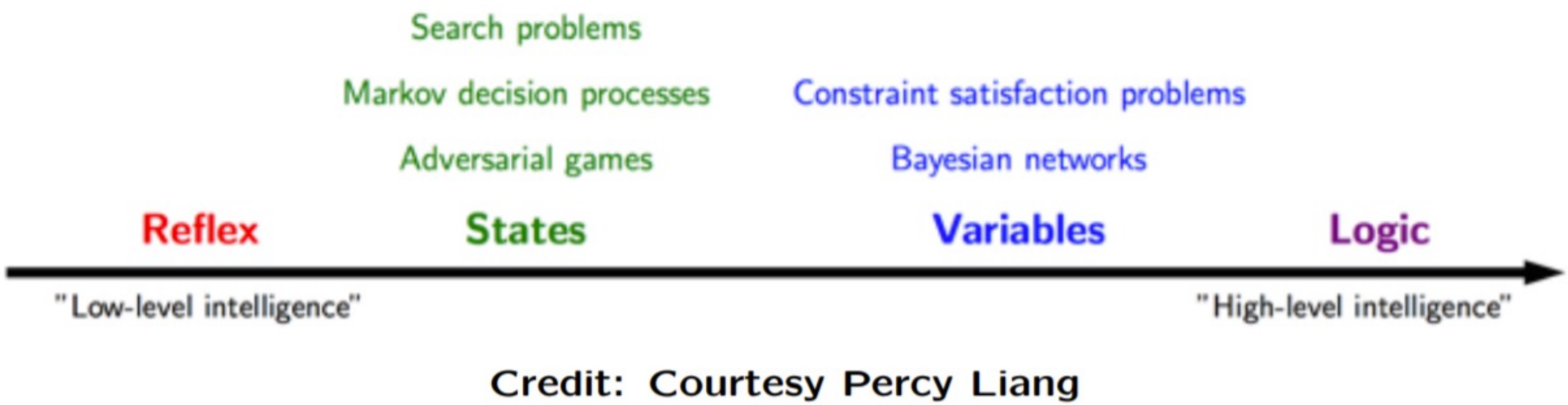
- Model compact representation of knowledge allowing agents to reason and draw conclusions.
- Basic representation of knowledge → Will restrict the space of possible worlds.

Knowledge-based agents



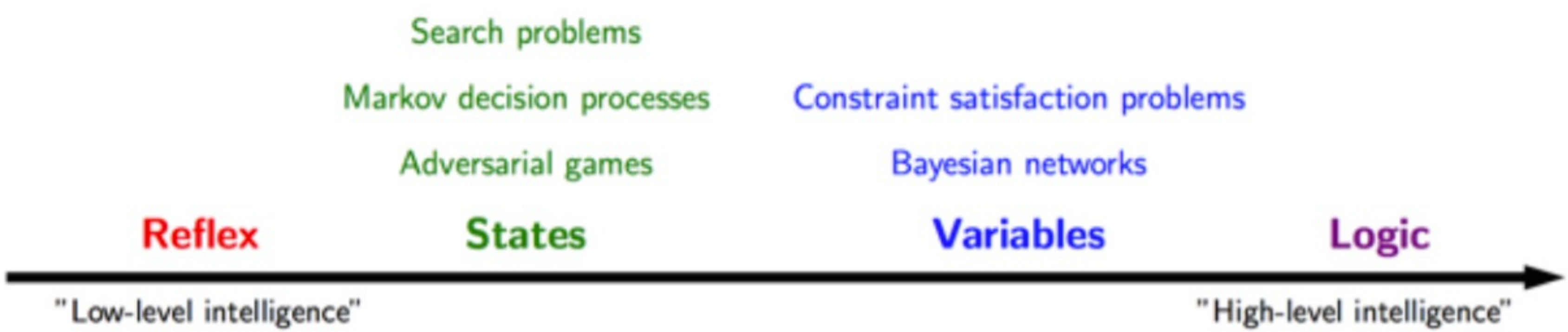
Credit: Courtesy Percy Liang

Knowledge-based agents



Knowledge-based agents

Where would Machine Learning go?



Credit: Courtesy Percy Liang

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

Why Logic?

- Richer representation of knowledge than belief space.
- Automated theorem proving.
- A good basis for probabilistic reasoning.

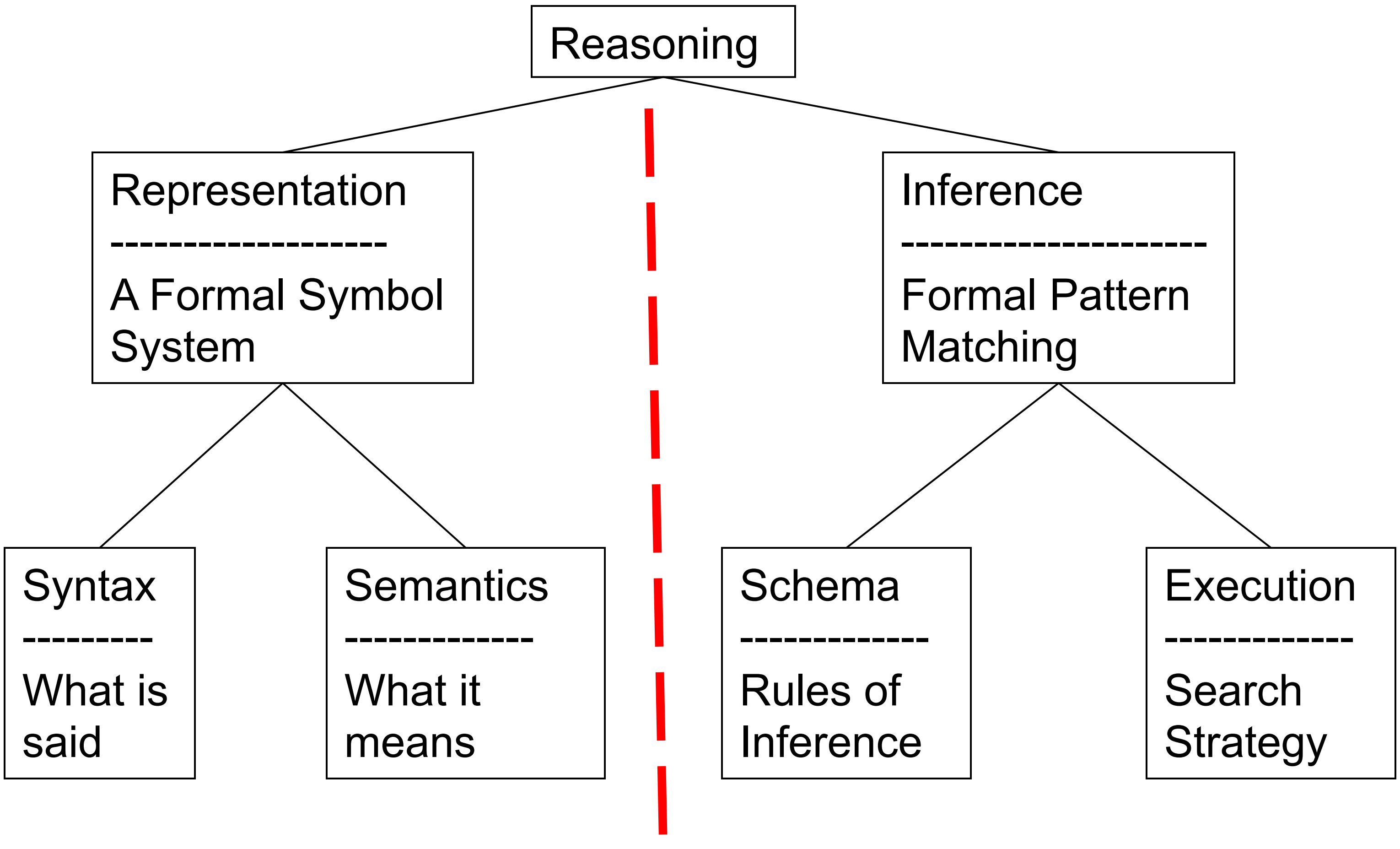
- We want an algorithm that reasons in a way that resembles reasoning in humans

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?



Knowledge base

- **knowledge base**: set of sentences that describe things agent knows

tell → knowledge base → ask

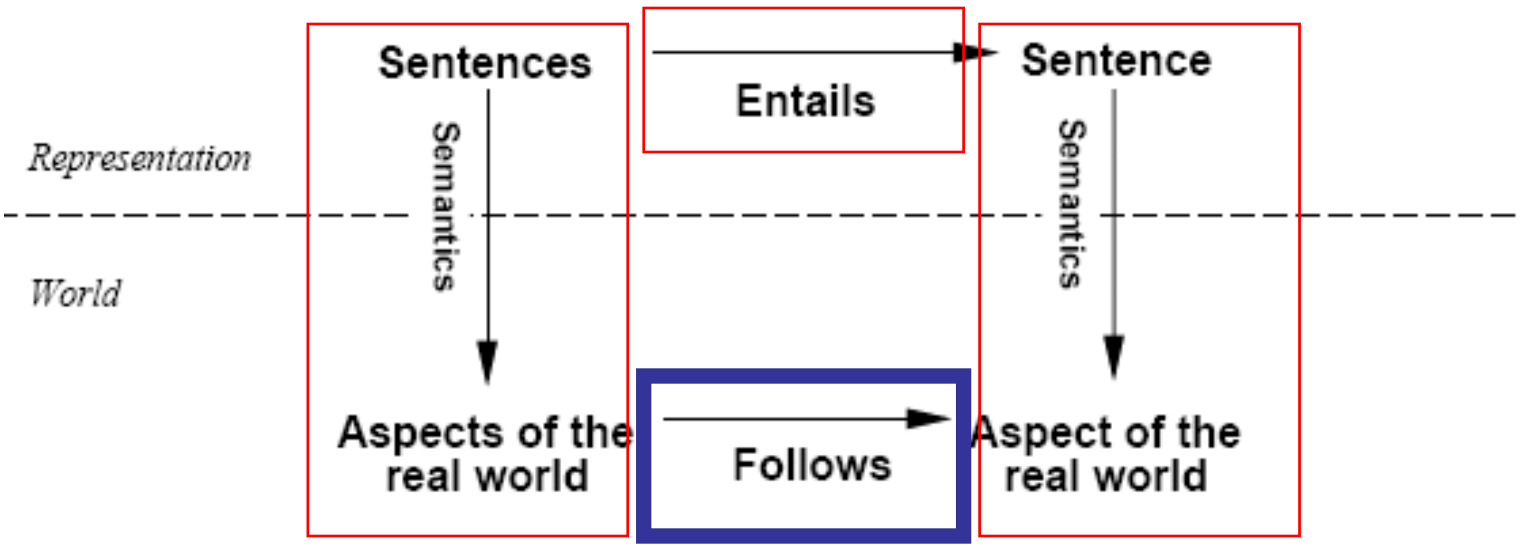
- **tell**: add sentence to knowledge base
- **ask**: For every possible world for which all the sentences in the KB are true, is some other sentence true?

Knowledge base

Viewed as constraints:

- tell: adds a constraint on the belief space.
- ask: if all constraints are satisfied

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.

Inference

- Both tell and ask may involve **inference**: combining old sentences to form new.
- Example
  - TELL: Father of John is Bob
  - TELL: Jane is John’ sister
  - TELL: John’s father is the same as John’ sister’s father
  - ASK: Who’s Jane father
- We need precise rules

Knowledge-Based Agents

- **KB = knowledge base**
  - A set of sentences or facts
  - e.g., a set of statements in a logic language

- **Inference**
  - Deriving new sentences from old
  - e.g., using a set of logical statements to infer new ones

- **A simple model for reasoning**
  - Agent is told or perceives new evidence
    - E.g., agent is told or perceives that A is true
  - Agent then infers new facts to add to the KB
    - E.g., KB = { (A -> (B OR C)); (not C) }

      then given A and not C the agent can infer that B is true
    - B is now added to the KB even though it was not explicitly asserted, i.e., the agent inferred B

Types of Logics

- **Propositional logic:** concrete statements that are either true or false
  - E.g., John is married to Sue.
- **Predicate logic (also called first order logic, first order predicate calculus):** allows statements to contain variables, functions, and quantifiers
  - For all X, Y: If X is married to Y then Y is married to X.
- **Probability:** statements that are possibly true; the chance I win the lottery?
- **Fuzzy logic:** vague statements; paint is <u>slightly grey</u>; sky is <u>very cloudy</u>.
- **Modal logic** is a class of various logics that introduce modalities:
  - **Temporal logic:** statements about time; John was a student at UCI for <u>four years</u>, and <u>before that</u> he spent <u>six years</u> in the US Marine Corps.
  - **Belief and knowledge:** Mary <u>knows</u> that John is married to Sue; a poker player <u>believes</u> that another player will fold upon a large bluff.
  - **Possibility and Necessity:** What <u>might</u> happen (possibility) and <u>must</u> happen (necessity); I <u>might</u> go to the movies; I <u>must</u> die and pay taxes.
  - **Obligation and Permission:** It is <u>obligatory</u> that students study for their tests; it is <u>permissible</u> that I go fishing when I am on vacation.

Types of Logics

- **Propositional logic:** concrete statements that are either true or false
  - E.g., John is married to Sue.
- **Predicate logic (also called first order logic, first order predicate calculus):** allows statements to contain variables, functions, and quantifiers
  - For all X, Y: If X is married to Y then Y is married to X.
- **Probability:** statements that are possibly true; the chance I win the lottery?
- **Fuzzy logic:** vague statements; paint is slightly grey; sky is very cloudy.
- **Modal logic** is a class of various logics that introduce modalities:
  - **Temporal logic:** statements about time; John was a student at UCI for four years, and before that he spent six years in the US Marine Corps.
  - **Belief and knowledge:** Mary knows that John is married to Sue; a poker player believes that another player will fold upon a large bluff.
  - **Possibility and Necessity:** What might happen (possibility) and must happen (necessity); I might go to the movies; I must die and pay taxes.
  - **Obligation and Permission:** It is obligatory that students study for their tests; it is permissible that I go fishing when I am on vacation.

Other Reasoning Systems

- How to produce new facts from old facts?
- **Induction**
  - Reason from facts to the general law
  - Scientific reasoning
- **Abduction**
  - Reason from facts to the best explanation
  - Medical diagnosis, hardware debugging
- **Analogy (and metaphor, simile)**
  - Reason that a new situation is like an old one

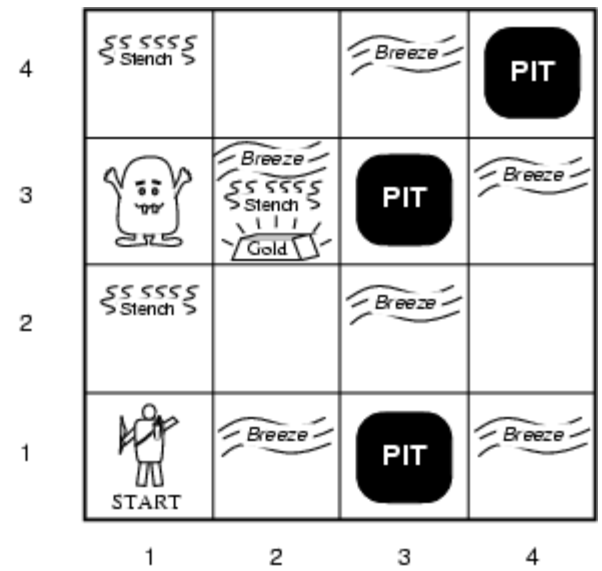
Other Reasoning Systems

Where would ML fit?

- How to produce new facts from old facts?
- **Induction**
  - Reason from facts to the general law
  - Scientific reasoning
- **Abduction**
  - Reason from facts to the best explanation
  - Medical diagnosis, hardware debugging
- **Analogy (and metaphor, simile)**
  - Reason that a new situation is like an old one

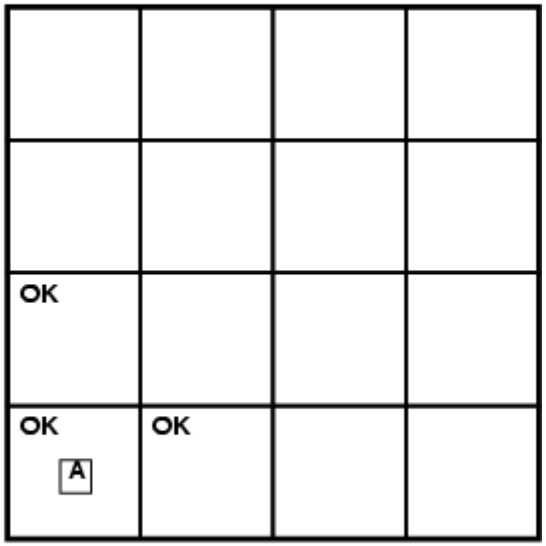
Wumpus World PEAS description

- 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



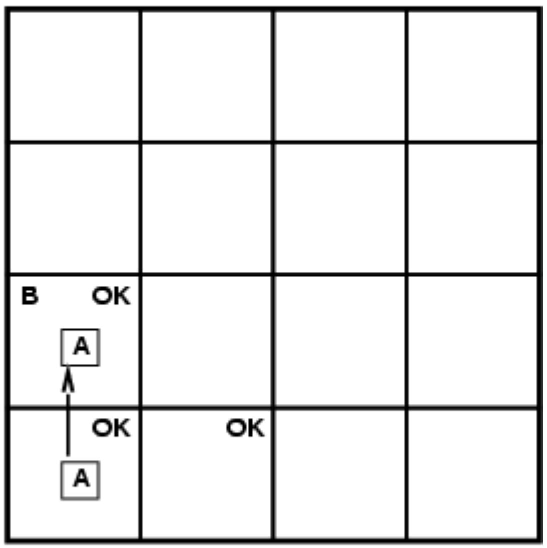
- Sensors: Stench, Breeze, Glitter, Bump, Scream
- Actuators: Left turn, Right turn, Forward, Grab, Release, Shoot

Exploring a wumpus world



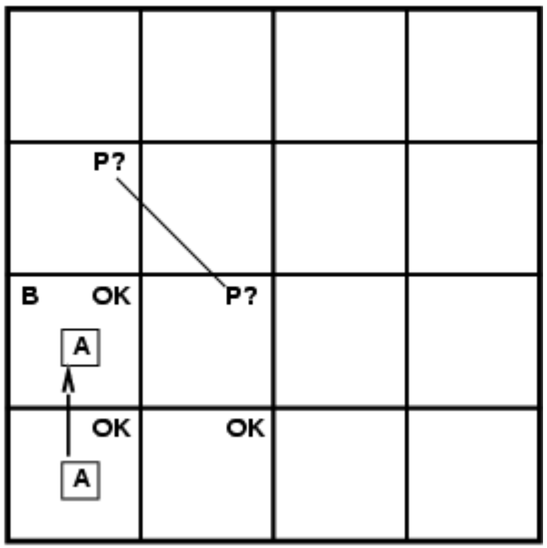
- Performance measure
  - gold: +1000, death: -1000
  - -1 per step, -10 for using the arrow
- Environment
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the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

Exploring a wumpus world



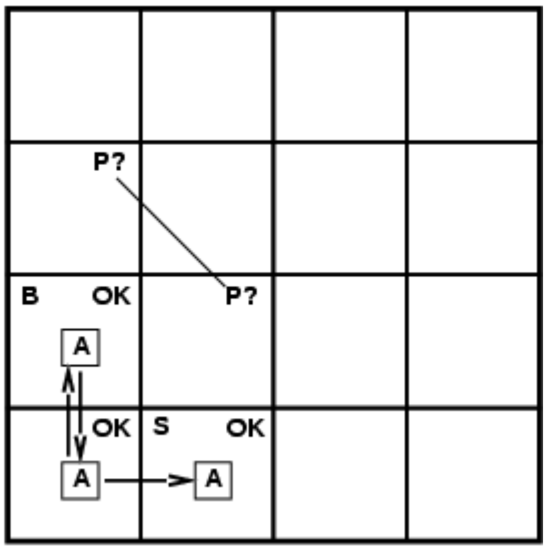
- Performance measure
  - gold: +1000, death: -1000
  - -1 per step, -10 for using the arrow
- Environment
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the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

Exploring a wumpus world



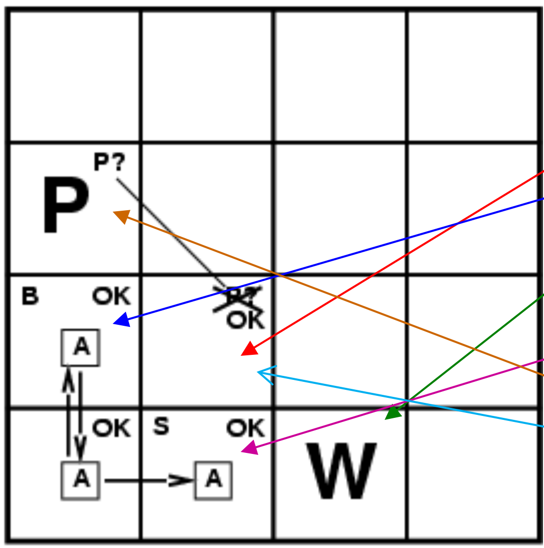
- Performance measure
  - gold: +1000, death: -1000
  - -1 per step, -10 for using the arrow
- Environment
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the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

Exploring a wumpus world



- 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
- Sensors: Stench, Breeze, Glitter, Bump, Scream
- Actuators: Left turn, Right turn, Forward, Grab, Release, Shoot

Exploring a Wumpus world

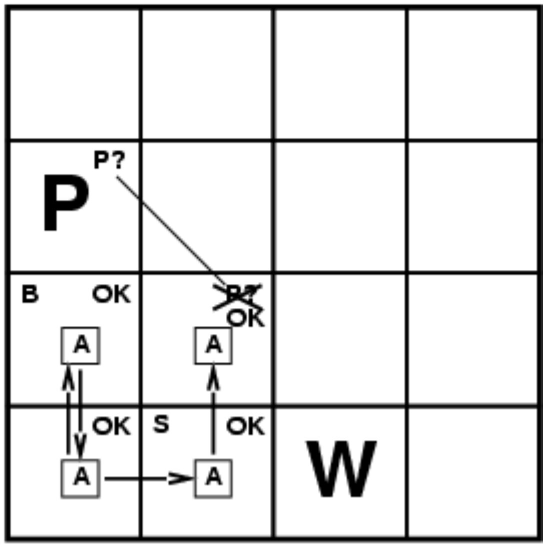


If the Wumpus were here, stench should be here. Therefore it is here.
Since, there is no breeze here, the pit must be there, and it must be OK here

- 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
- Sensors: Stench, Breeze, Glitter, Bump, Scream
- Actuators: Left turn, Right turn, Forward, Grab, Release, Shoot

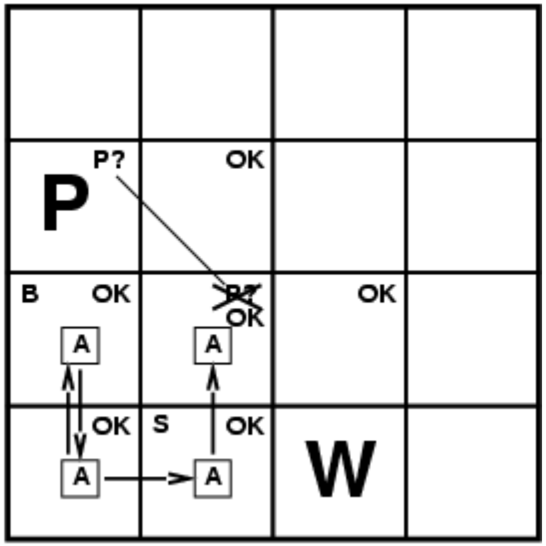
We need rather sophisticated reasoning here!

Exploring a wumpus world



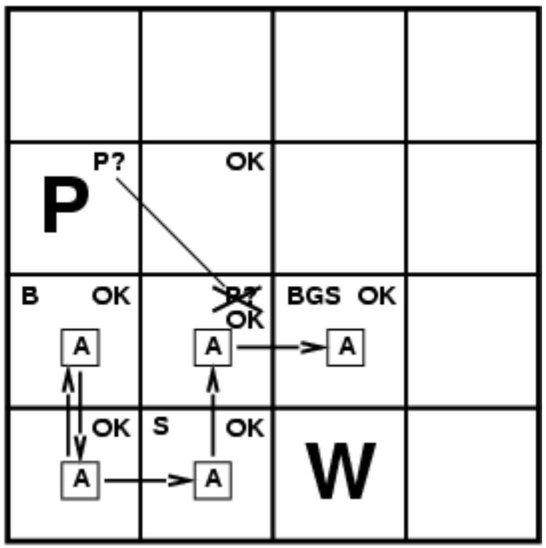
- Performance measure
  - gold: +1000, death: -1000
  - -1 per step, -10 for using the arrow
- Environment
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the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

Exploring a wumpus world



- 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
- Sensors: Stench, Breeze, Glitter, Bump, Scream
- Actuators: Left turn, Right turn, Forward, Grab, Release, Shoot

Exploring a wumpus world

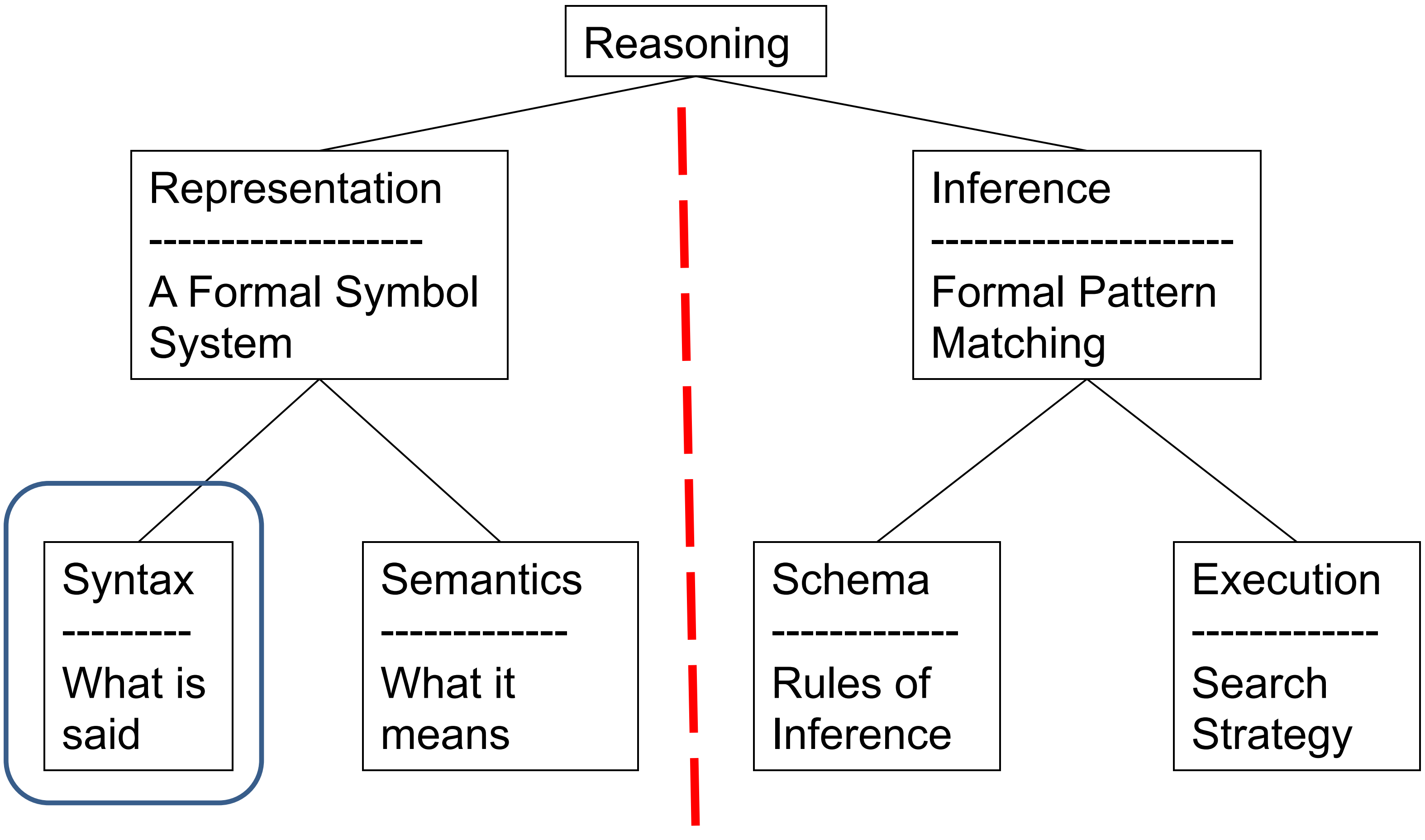


- Performance measure
  - gold: +1000, death: -1000
  - -1 per step, -10 for using the arrow
- Environment
  - Squares adjacent to wumpus are smelly
  - Squares adjacent to pit are breezy
  - Glitter iff gold is in the same square
  - Shooting kills wumpus if you are facing it
  - Shooting uses up the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

Logic in general

- 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

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



Syntax

- **Syntax** is a set of rules defining well-formed sentences
- Syntax gives the *domain* of possible sentences: the set from which sentences may be drawn

Syntax example

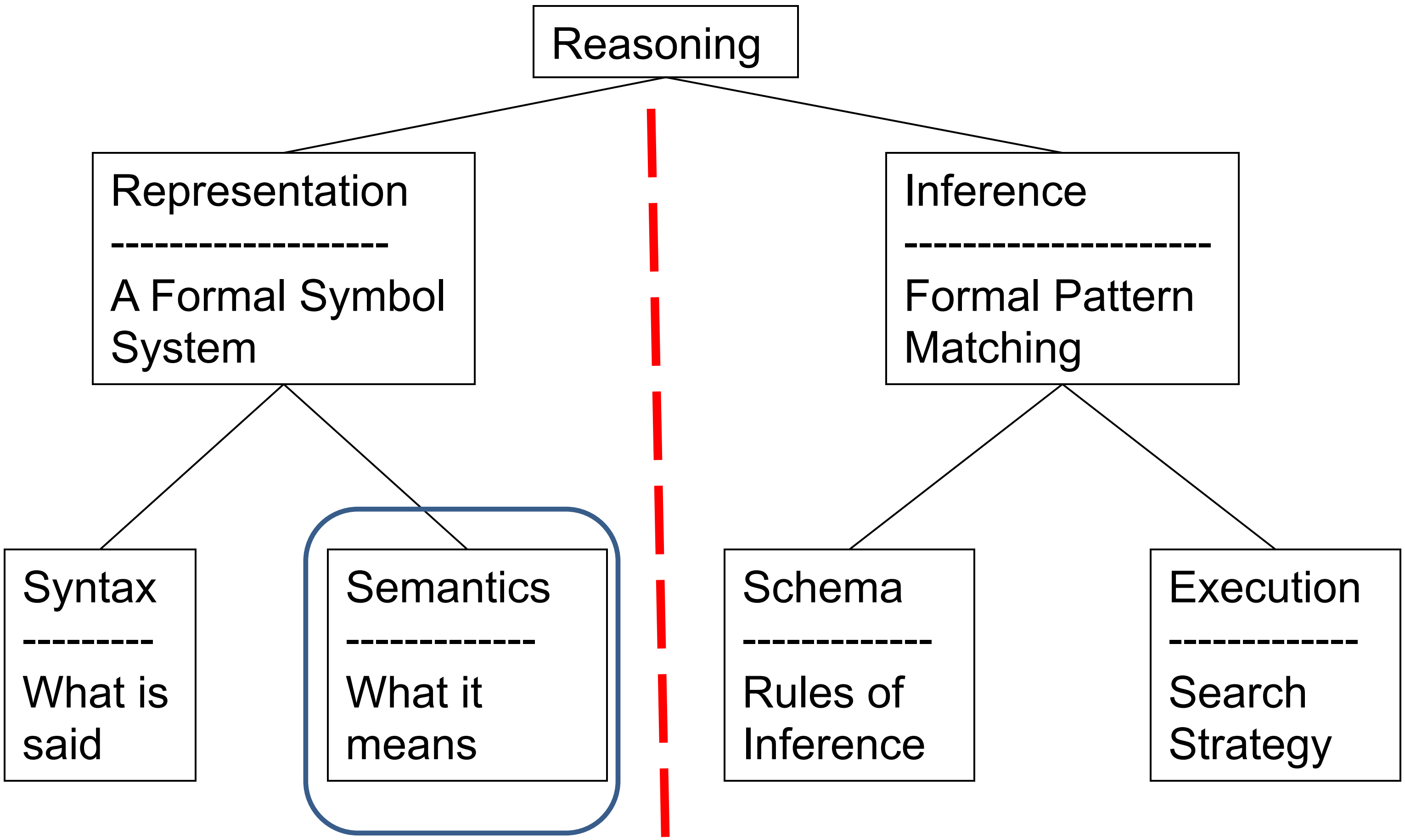
- Programming language
  - print("hello world")
  - ++++++++++[>+++++++>++++++++++>+++<<<-]>++.>+.+++++++..+++.>++.<<+++++++++++++++.>.+++.------.--------.>+.

From <https://en.wikipedia.org/wiki/Esoteric_programming_language>

Syntax example

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

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



Semantics

- **Semantics** define the truth of the sentence w.r.t. each possible world, called a **model**.

- Typically, sentences involve some variables, and variables have domains. A **model** is an assignment of values to variables.

Example. Is the sentence x+y=4 true? (It satisfies the syntax for arithmetic expressions.)

- We could imagine 1+6=4; just symbols.
- But we expect there to be some values of x and y (models) for which the sentence is true, and some other values for which it is not.

Semantics

- We use semantics to **define** the set of worlds for which x+y=4 is true:

x	y	x+y=4
0	0	False
1	0	False
3	1	True
1	3	True
2	2	True

Semantics

- The table is incomplete. We need to know if sentence is true or false for each possible model, and there are infinitely many models.

$$\begin{array}{l} \text{sentence} \rightarrow \\ \text{model} \rightarrow \end{array} \boxed{\text{semantics}} \rightarrow \text{true/false}$$

Entailment – formalism

- Let α and β be sentences.
- We say that $\alpha \vDash \beta$ iff for every model in which α is true, β is true.
- We let $M(\alpha)$ be the set of models for which a sentence α is true. Then $\alpha \vDash \beta$ means $M(\alpha) \subset M(\beta)$

Entailment for the logic agent

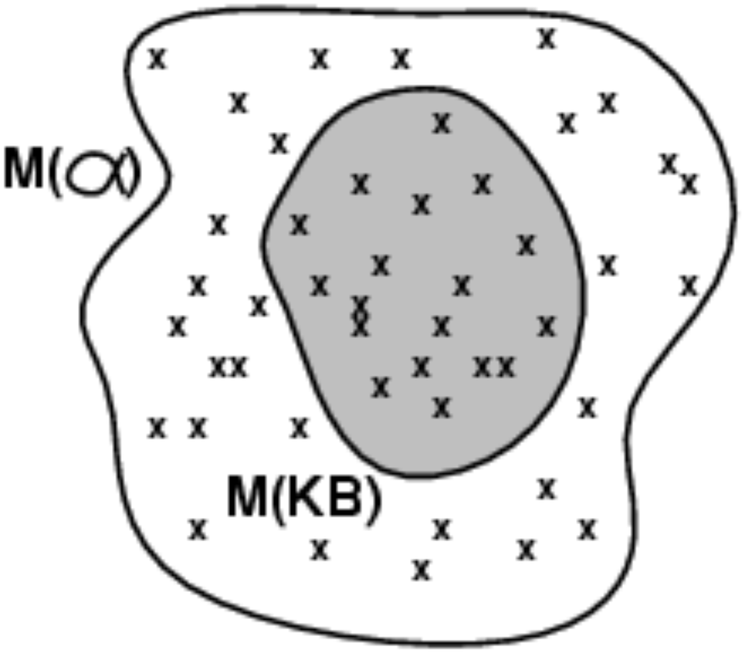
- Entailment means that one thing follows from another set of things:

$$KB \models \alpha$$

- Knowledge base *KB* entails sentence α if and only if α is true in all worlds wherein *KB* is true
- The entailed α MUST BE TRUE in ANY world in which KB IS TRUE.
- Any new sentence that is entailed is less constraining. But maybe in a more useful format.

Models

- 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
- Then KB $\models$ α iff *M(KB)* $\subseteq$ *M*(α)
- Think of KB and α as collections of constraints and of models m as possible states. M(KB) are the solutions to KB and M(α) the solutions to α. Then, KB $\models$ α when all solutions to KB are also solutions to α.

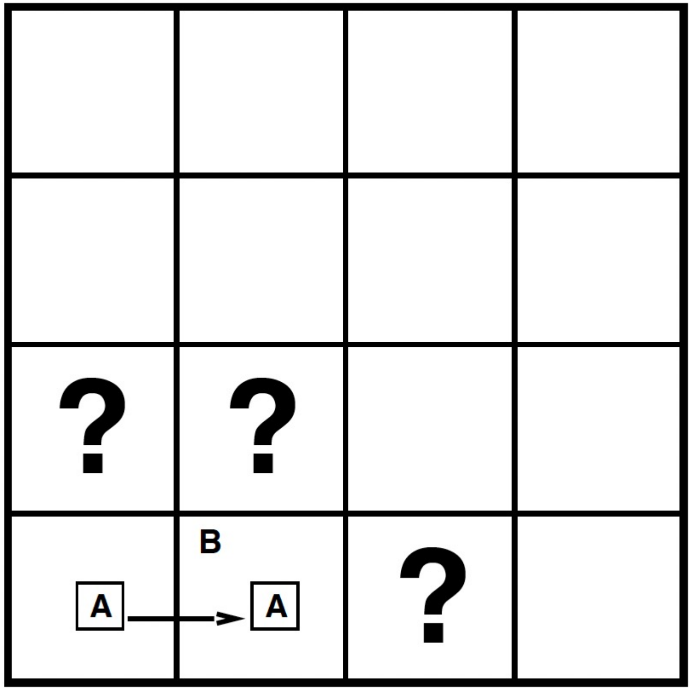


Entailment examples

- E.g., the KB = “the Giants won and the Reds won” entails α = “The Giants won”.
- Example. In arithmetic, we say that $x=0 \models xy=0$. If you choose a model (say x=0, y=6) such that x=0 is true, then the sentence xy=0 is also true.
- E.g., KB = “x+y = 4” entails α = “4 = x+y”
- E.g., KB = “Mary is Sue’s sister and Amy is Sue’s daughter” entails α = “Mary is Amy’s aunt.”

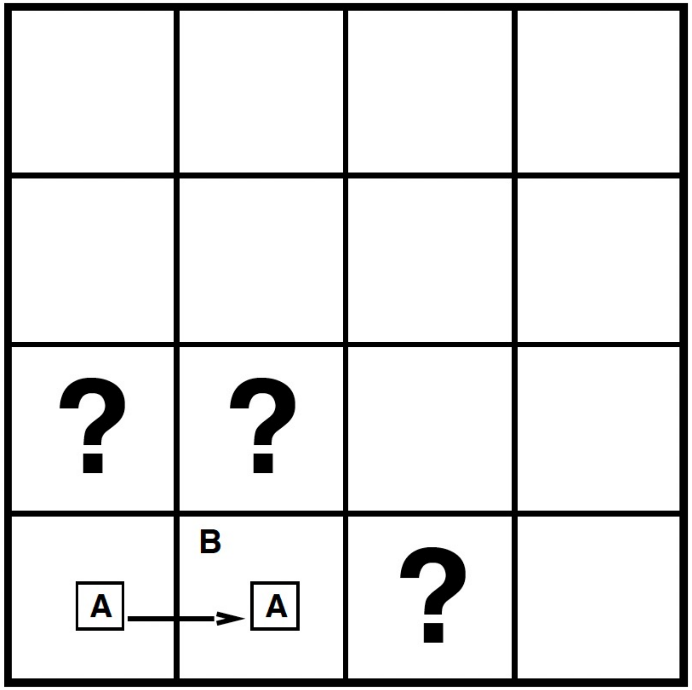
Entailment example in Wumpus world

- Situation after detecting nothing in [1,1],
- moving right, breeze in [2,1]
- Consider possible models for ?s assuming only pits

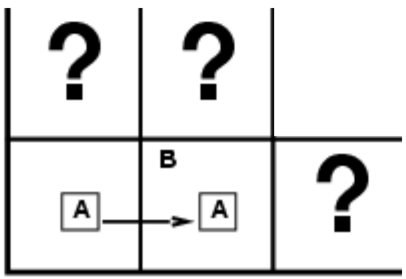


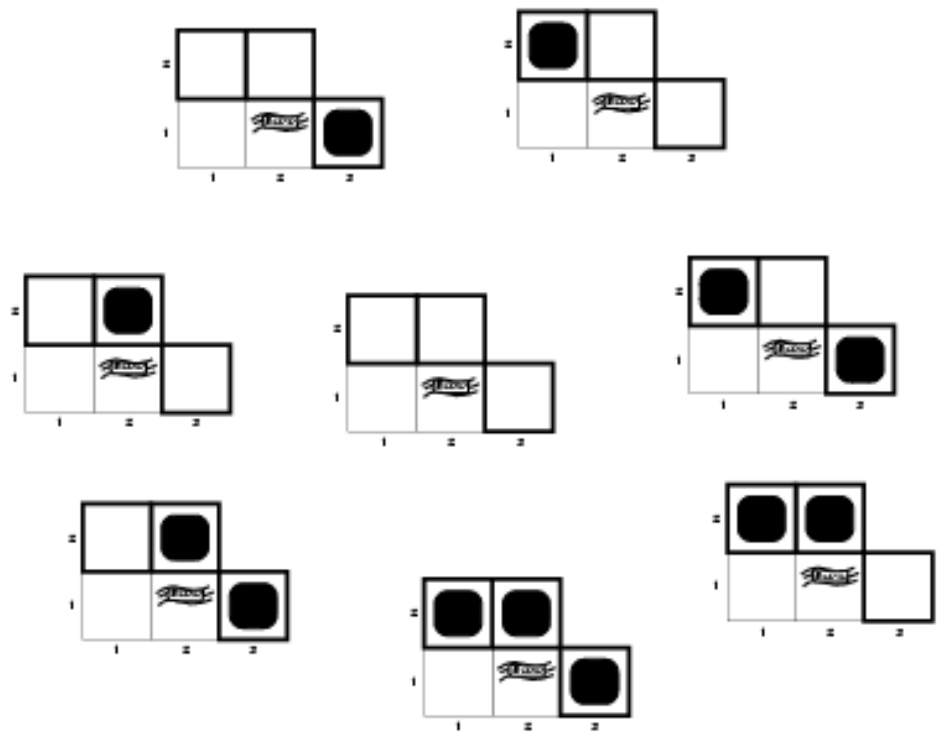
Entailment example in Wumpus world

- Situation after detecting nothing in [1,1],
- moving right, breeze in [2,1]
- Consider possible models for ?s assuming only pits
- 3 Boolean choices -> 8 possible models



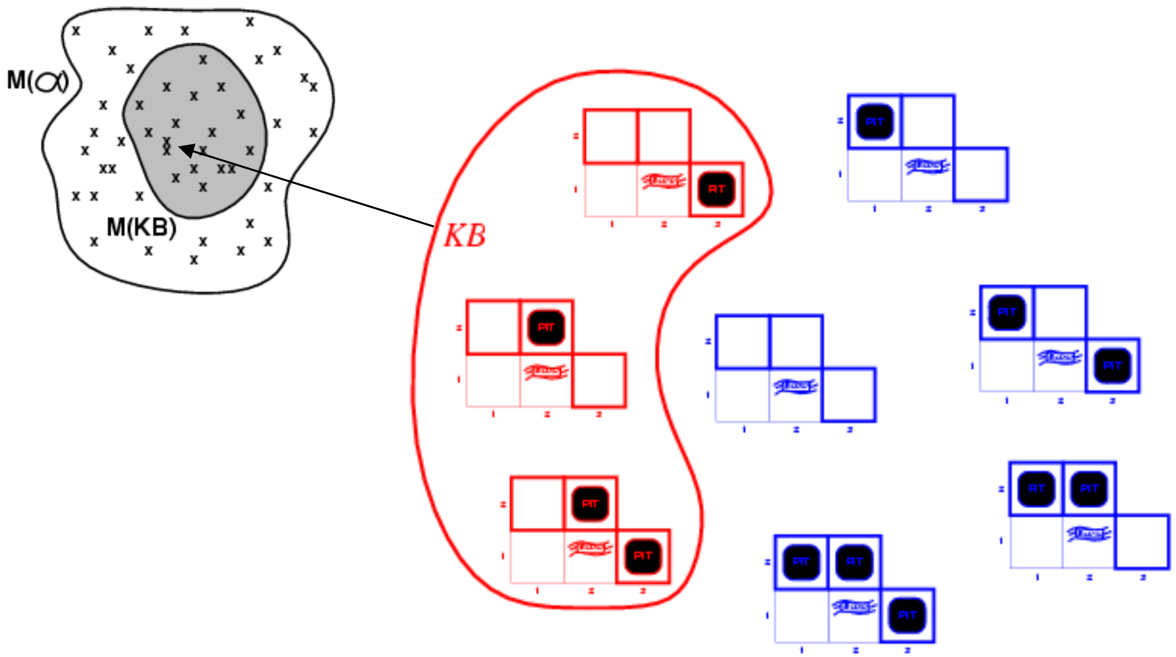
Wumpus models





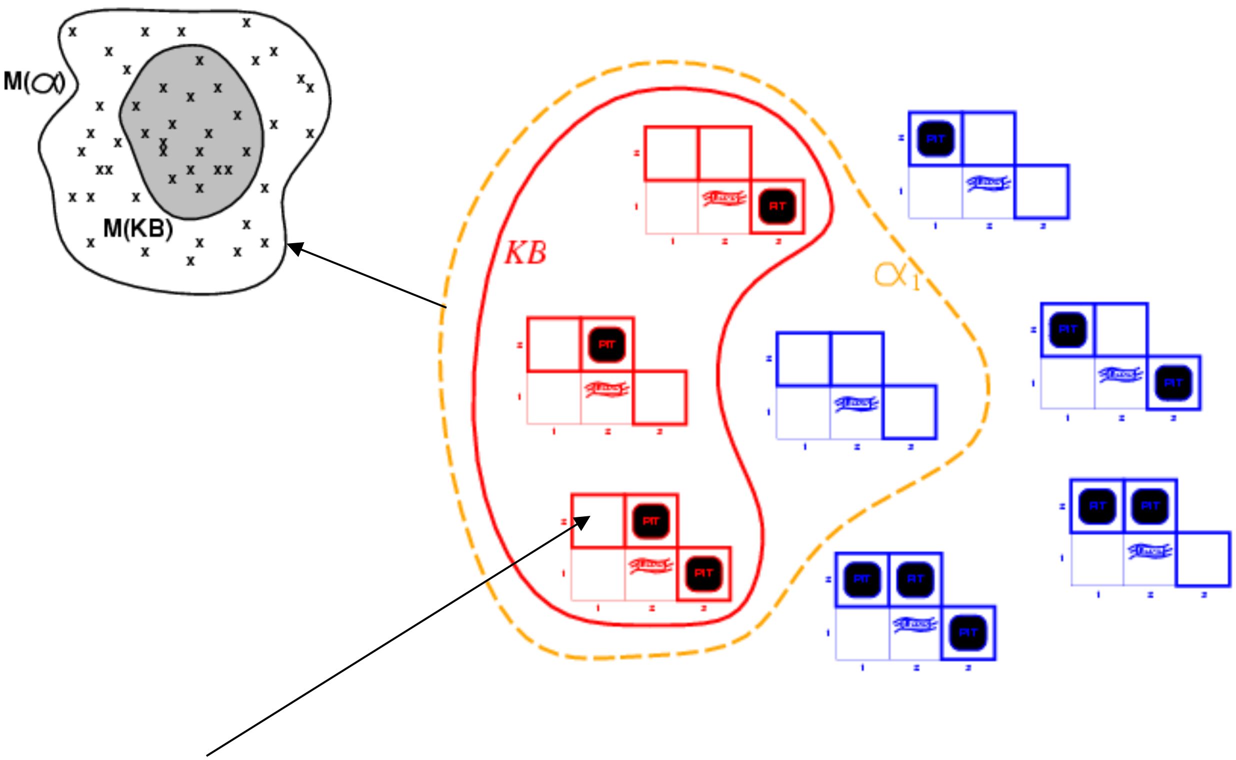
All possible models in this reduced Wumpus world. What can we infer?

Wumpus models



- *M(KB)* = all possible wumpus-worlds consistent with the observations and the "physics" of the Wumpus world.

Wumpus models



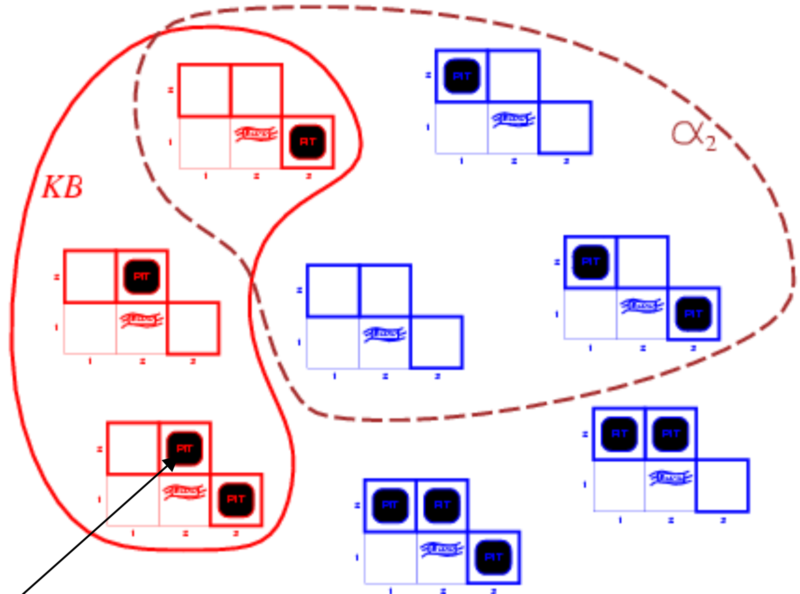
Now we have a query sentence, α_1 = "[1,2] is safe“

$KB \models \alpha_1$

M(KB) (red outline) is a subset of M(α_1) (orange dashed outline)

$\Rightarrow$ **α_1 is true in any world in which KB is true**

Wumpus models



Now we have another query sentence, α_2 = "[2,2] is safe"
$KB \not\models \alpha_2$,
M(KB) (red outline) is a **not** a subset of M(α_2) (dashed outline)
⇒ α_2 is false in some world(s) in which KB is true

Monotonicity

- Monotonicity: Each new sentence added to the knowledge base further constrains the set of models that holds.

- ⇒ if we can prove that some sentence is entailed by a set of sentences in the knowledge base, then adding new sentences to the knowledge base will never invalidate that proof.

Propositional logic

- Winter $\land$ NiceWeatherSunday $\Rightarrow$ Procrastinated

- Atomic sentence: a symbol that can take on the value true or false.
- Literal: atomic sentence, or negated atomic sentence
- Logical connectives: $\neg \lor \land \Rightarrow \Leftrightarrow$

Backus-Naur form

- Backus-Naur Form gives a recursive definition of syntax, the set of all legal sentences

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

Model and propositional logic

- model: true false values for every atomic sentence

- A world with the symbols isSnowing and isSunny would have the four models (true, true), (true, false), (false, true), and (false, false).

Propositional logic: semantics

- Take a model and sentence and evaluate to T/F. Easy for atomic sentences. For complex sentences, write some rules using **truth tables** and apply recursively.

P	Q	$P \wedge Q$
F	F	F
F	T	F
T	F	F
T	T	T

Propositional logic: semantics

- Definition of the **implies** connective:
- $P \Rightarrow Q$ is true in models for which either P is false, or both P and Q are true.

P	Q	$P \Rightarrow Q$
F	F	T
F	T	T
T	F	F
T	T	T

Truth tables for all logical connectives

P	Q	¬P	P∧Q	P∨Q	P⇒Q	P⇔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

Summary

- Model: assignment of values to variables
- Sentences: used to select a set of models (winter)
- Syntax: description of legal sentences
- Semantics: maps (sentence + model) to T/F
- Entailment: $\alpha \models \beta$. ("it is greater than 100 degrees" entails "it is greater than 32 degrees")
- Propositional logic with symbols and connectives

Next

- How to make inference?