Artificial Intelligence

Logical Agents and Propositional Logic

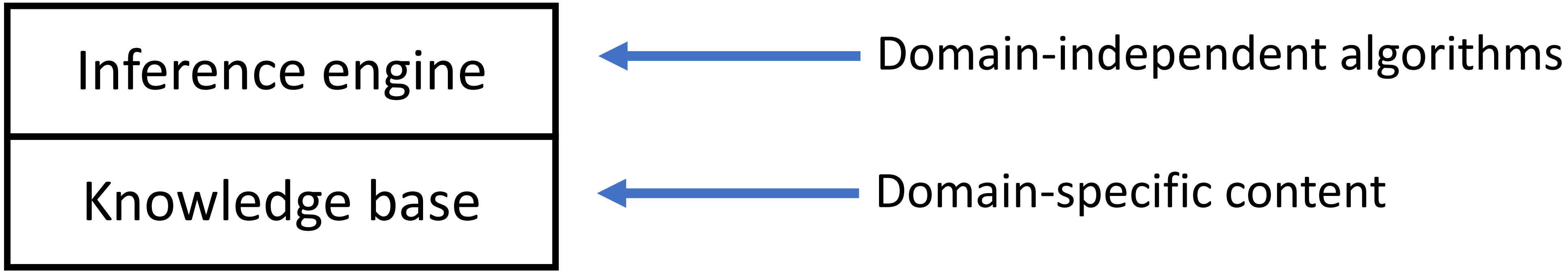
CS 444 – Spring 2020
Dr. Kevin Molloy
Department of Computer Science
James Madison University

JAMES MADISON UNIVERSITY

Outline

- Knowledge-based Agents
- Wumpus World
- Logic – Models and Entailment
- Propositional (Boolean Logic)
- Model checking: Inference by Enumeration

Knowledge Bases



Knowledge base = set of sentences in a **formal** language
Declarative approach to build 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

A Simple Knowledge-based Agent

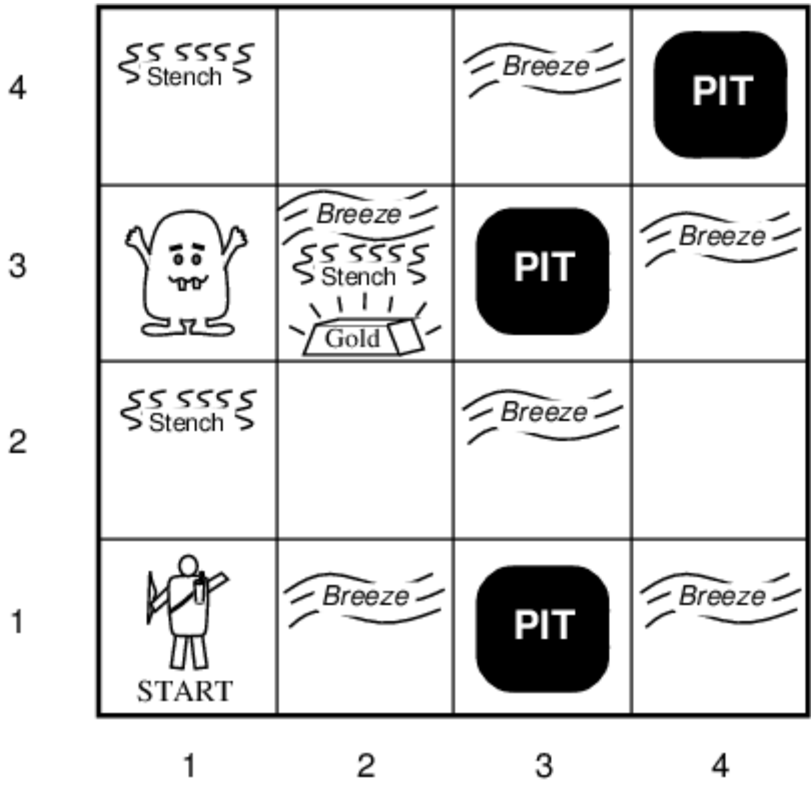
```
function KB-Agent(percept) returns an action
    Static:   KB, a knowledge base
              t, a counter, initially 0, that indicates time

    Tell(KB, Make-Percept-Sentence(percept, t))
    action ← Ask(KB, Make-Action-Query(t))
    Tell(KB, Make-Percept-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
- Squeal sound means Wumpus was killed

Actuators

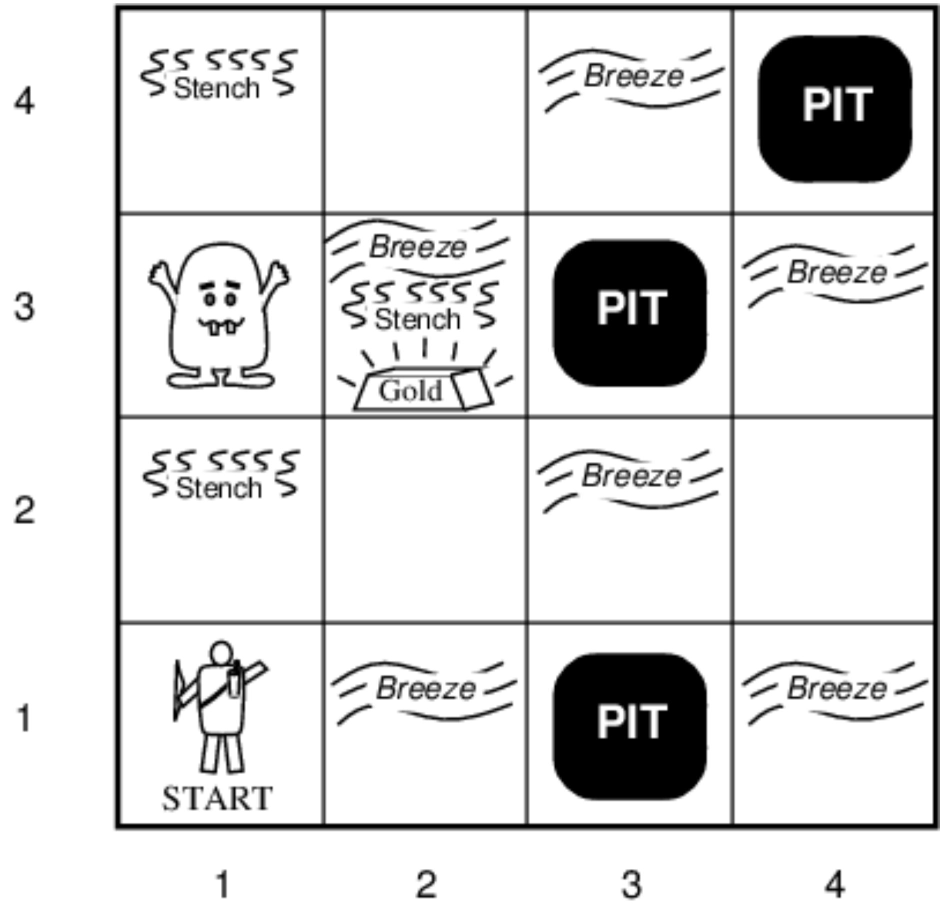
Left turn, right turn, forward, grab, release, shoot

Sensors

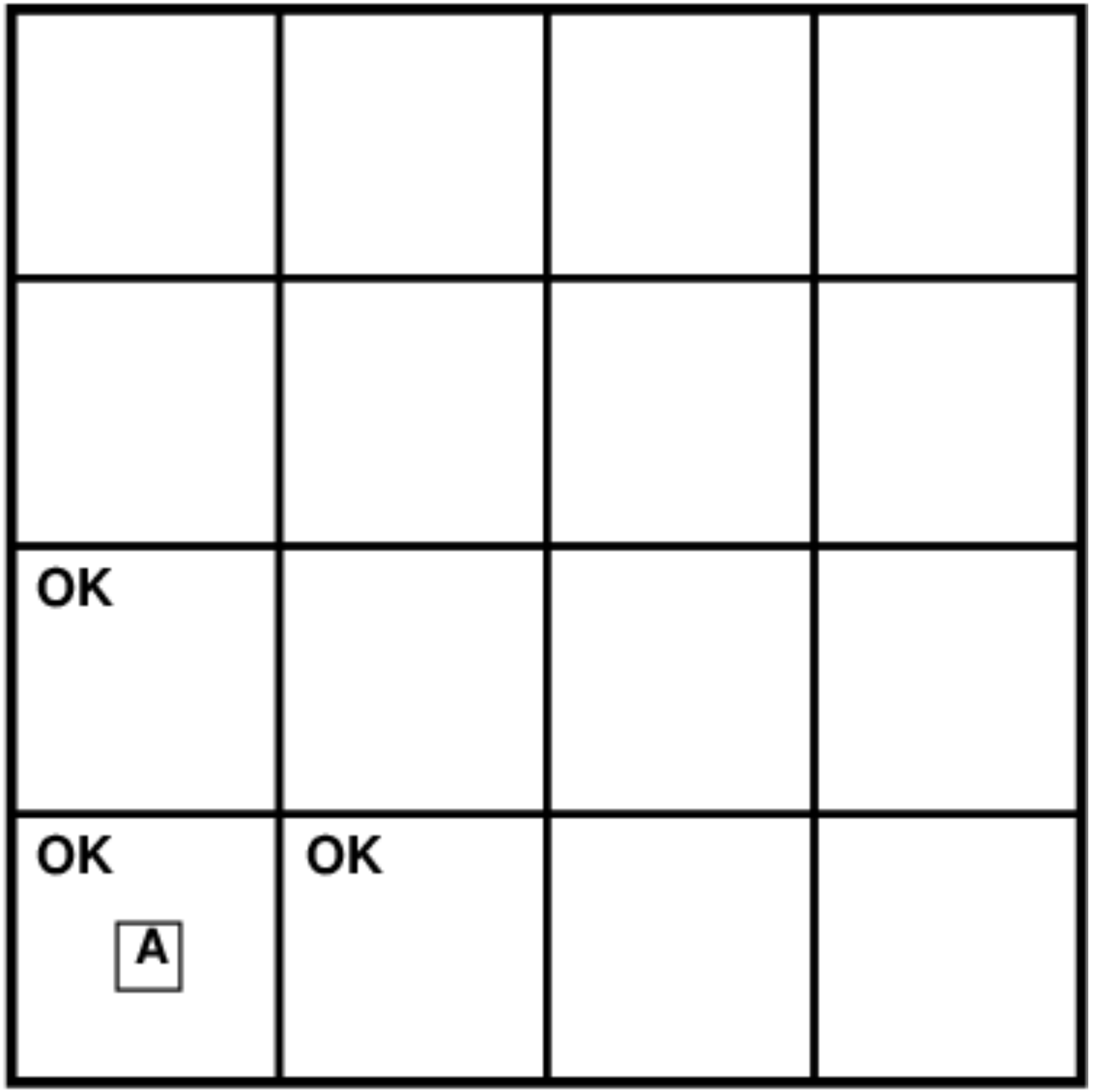
Breeze, glitter, smell

Wumpus World Characterizations

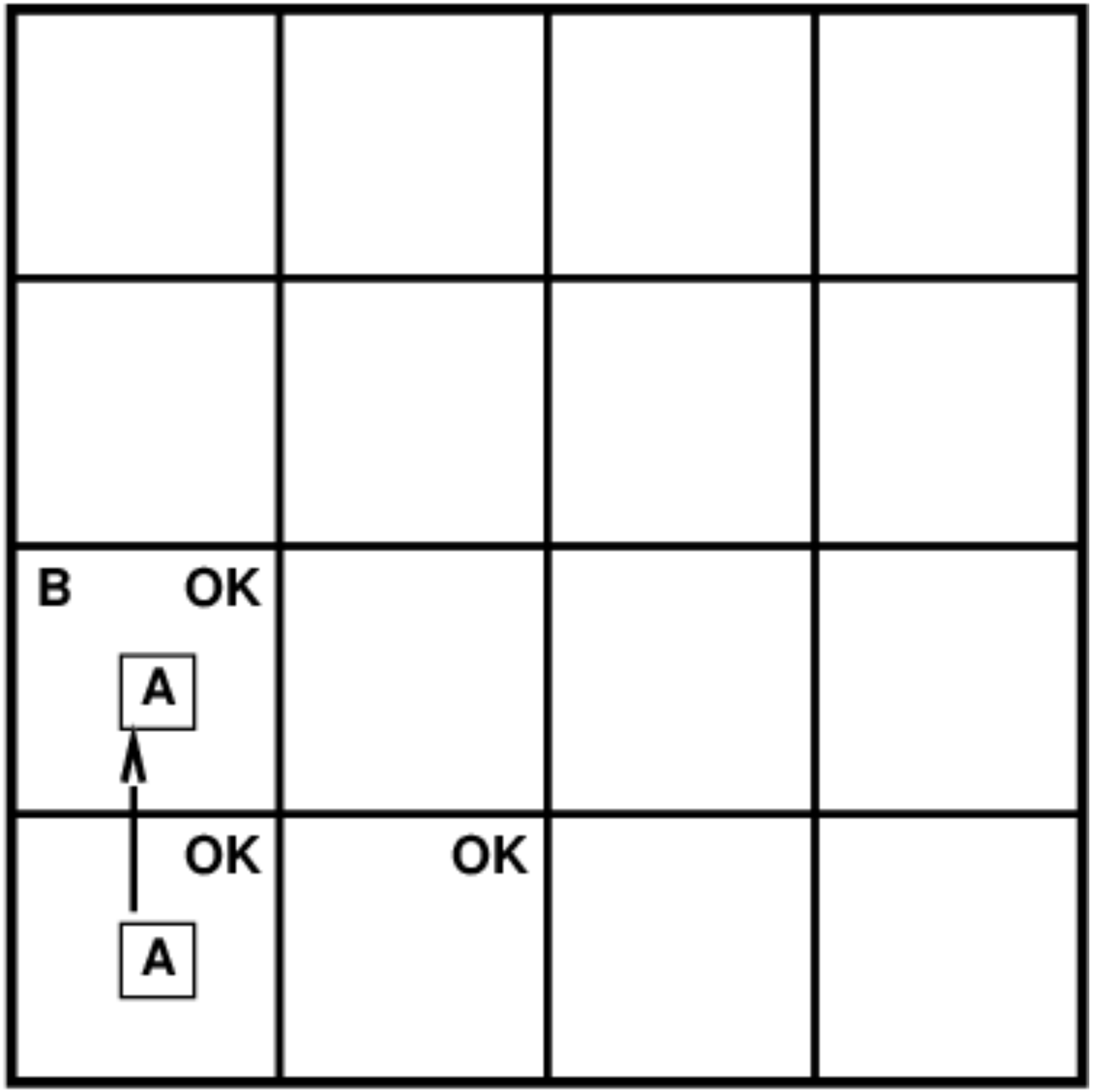
Observable?	Partially observable – only local perception
Deterministic?	Yes – outcomes exactly specified
Episodic?	No – sequential at the level of actions
Static ?	Yes – Wumpus and pits do not move
Discrete?	Yes
Single Agent?	Yes (Wumpus is essentially a natural feature)



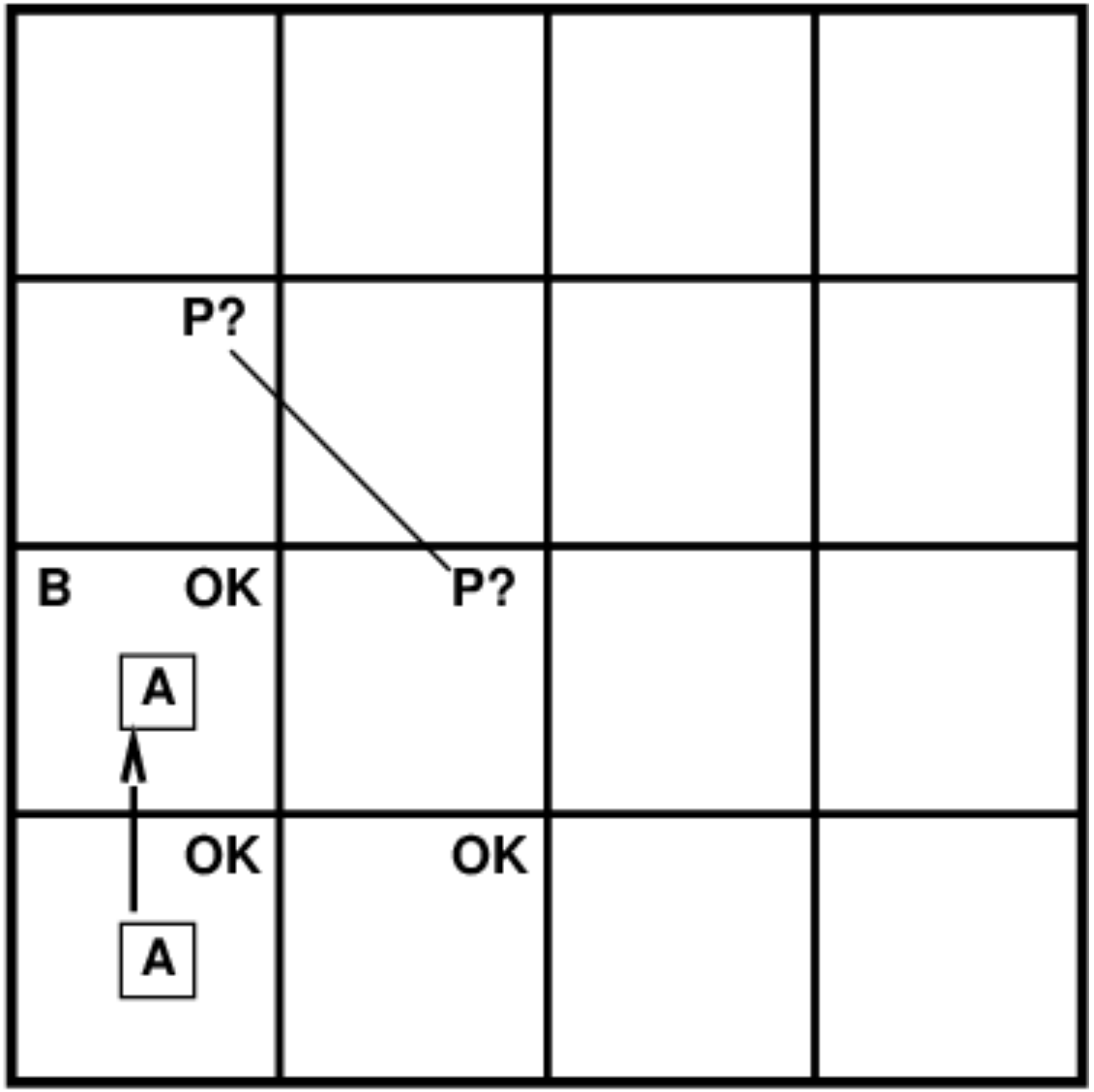
Exploring a Wumpus World



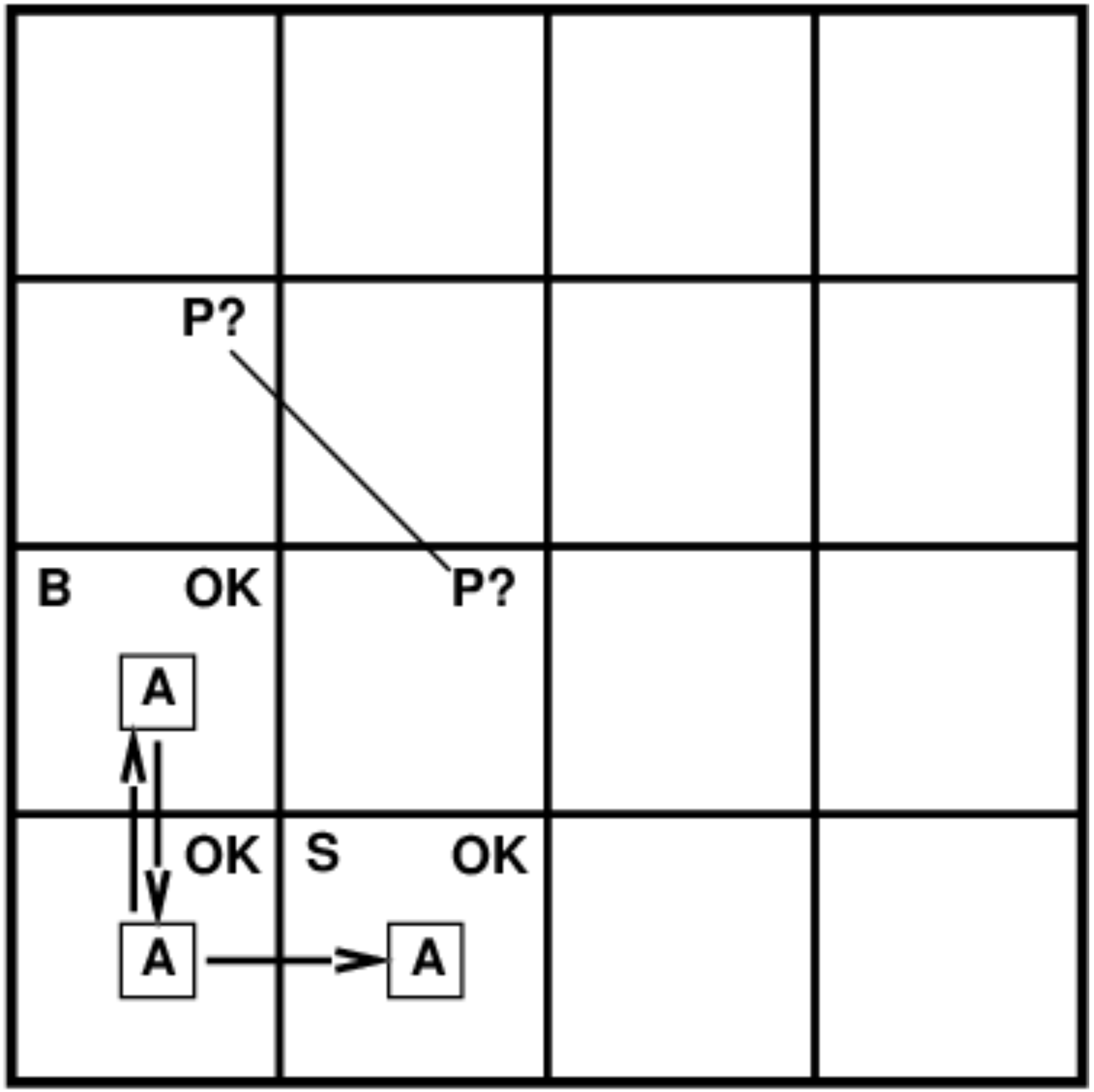
Exploring a Wumpus World



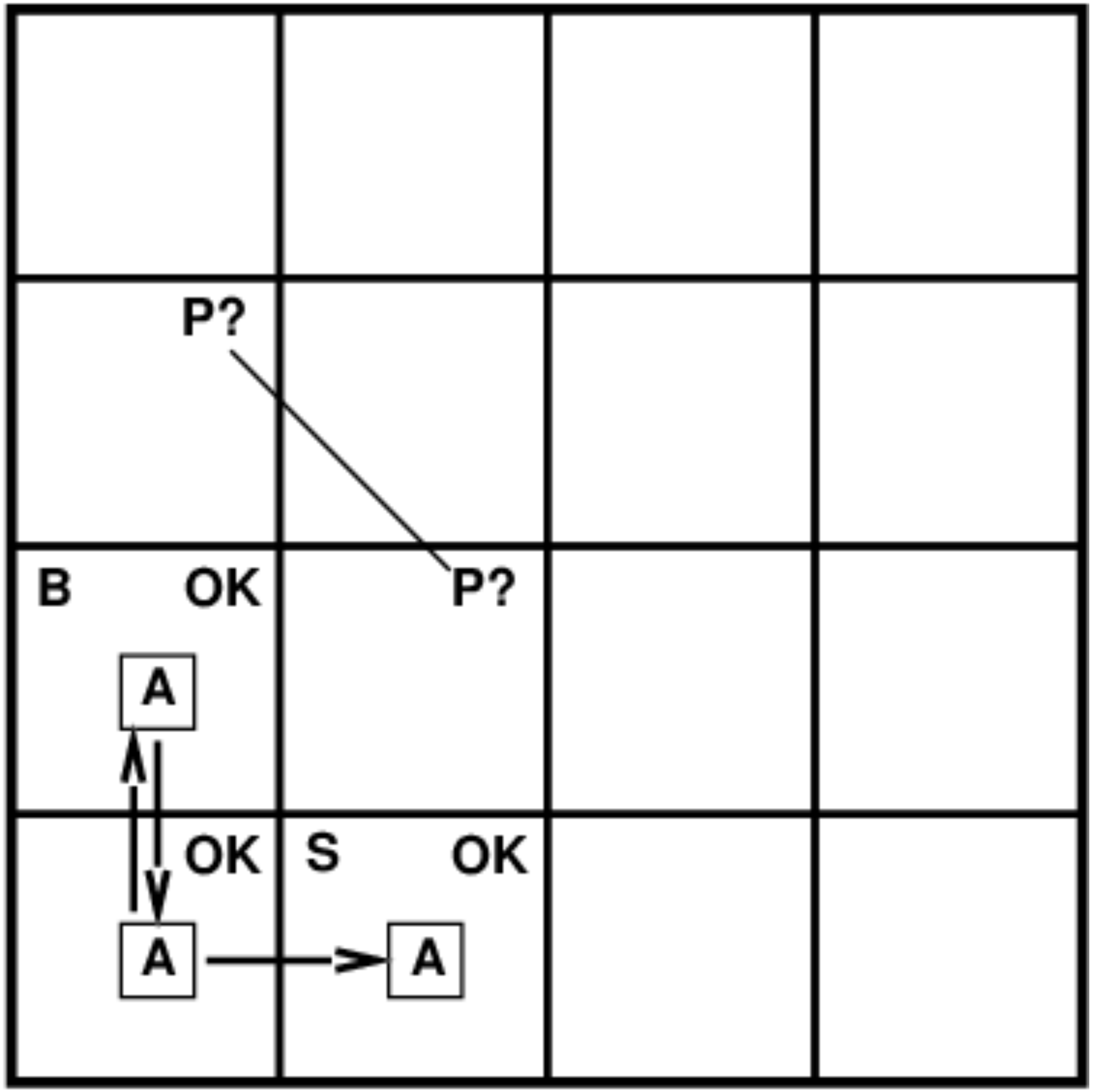
Exploring a Wumpus World



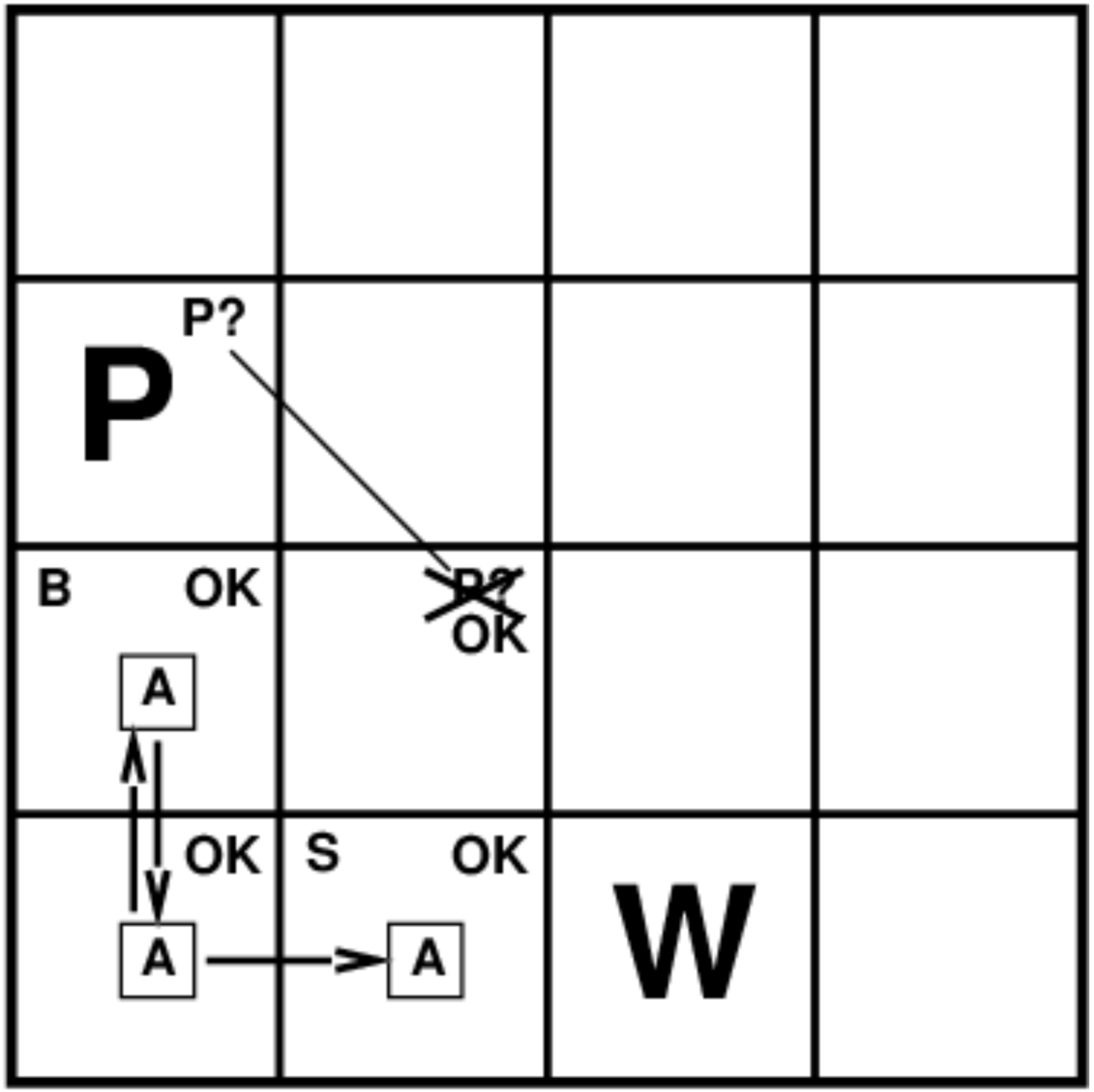
Exploring a Wumpus World



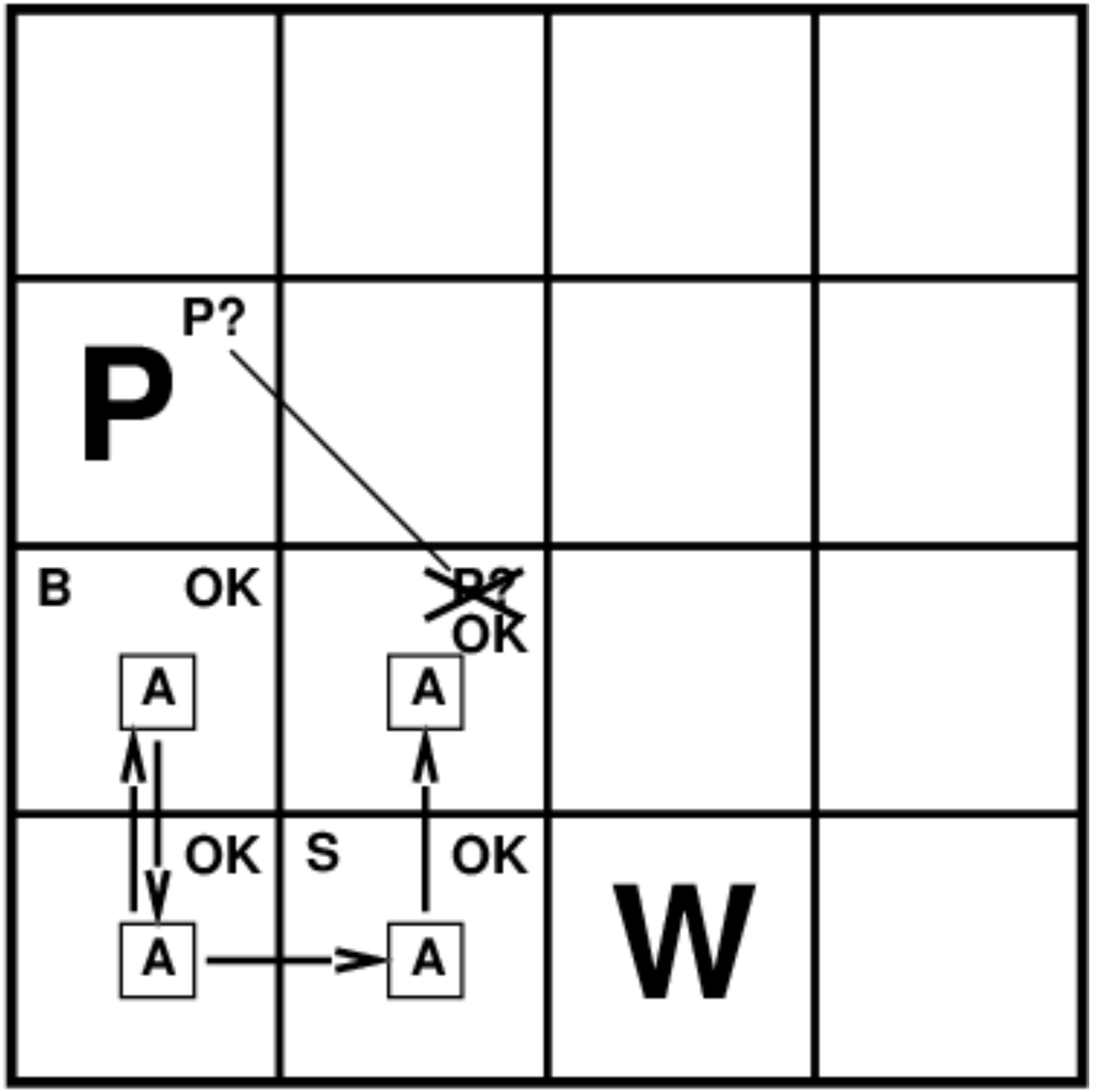
Exploring a Wumpus World



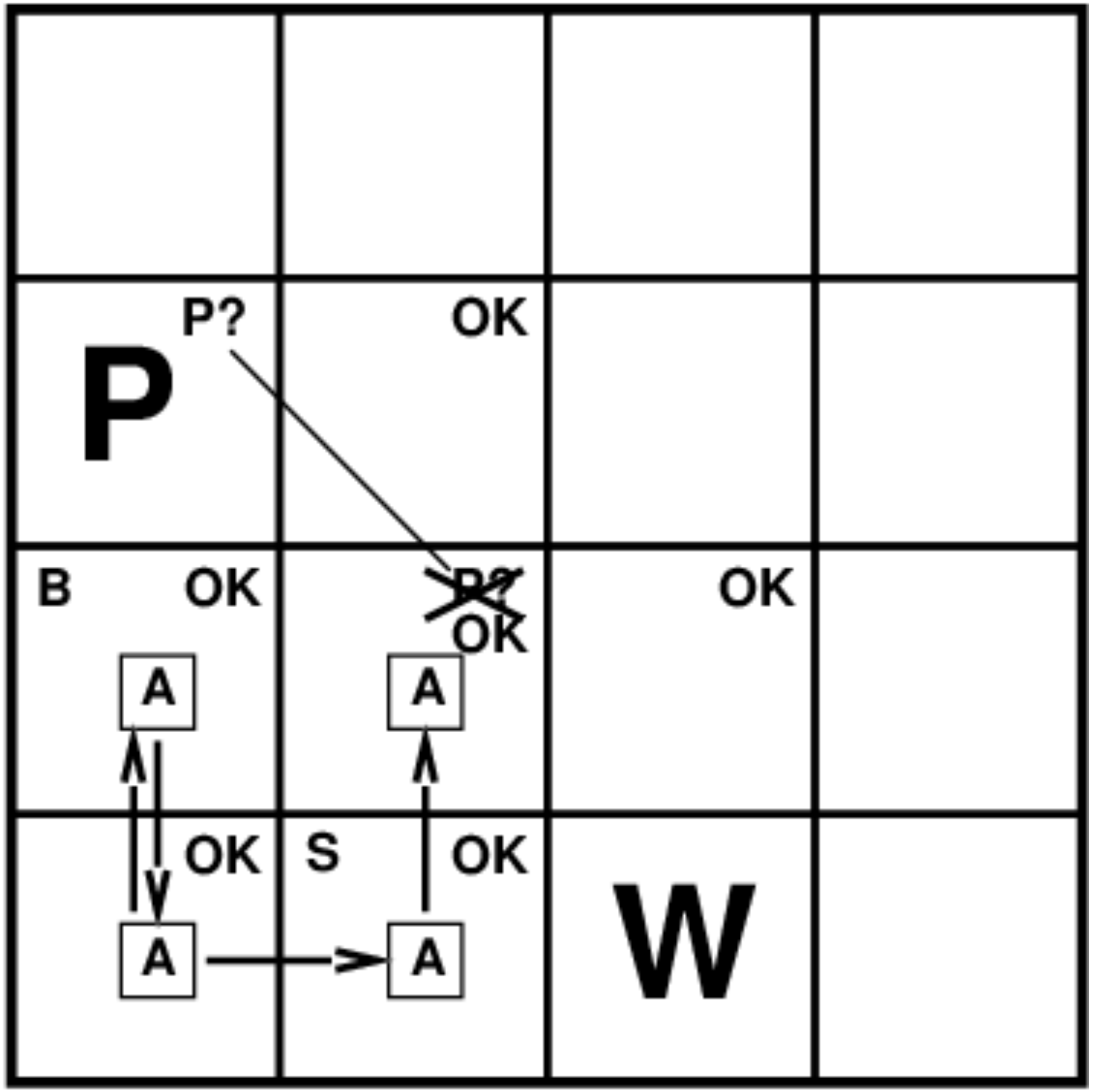
Exploring a Wumpus World



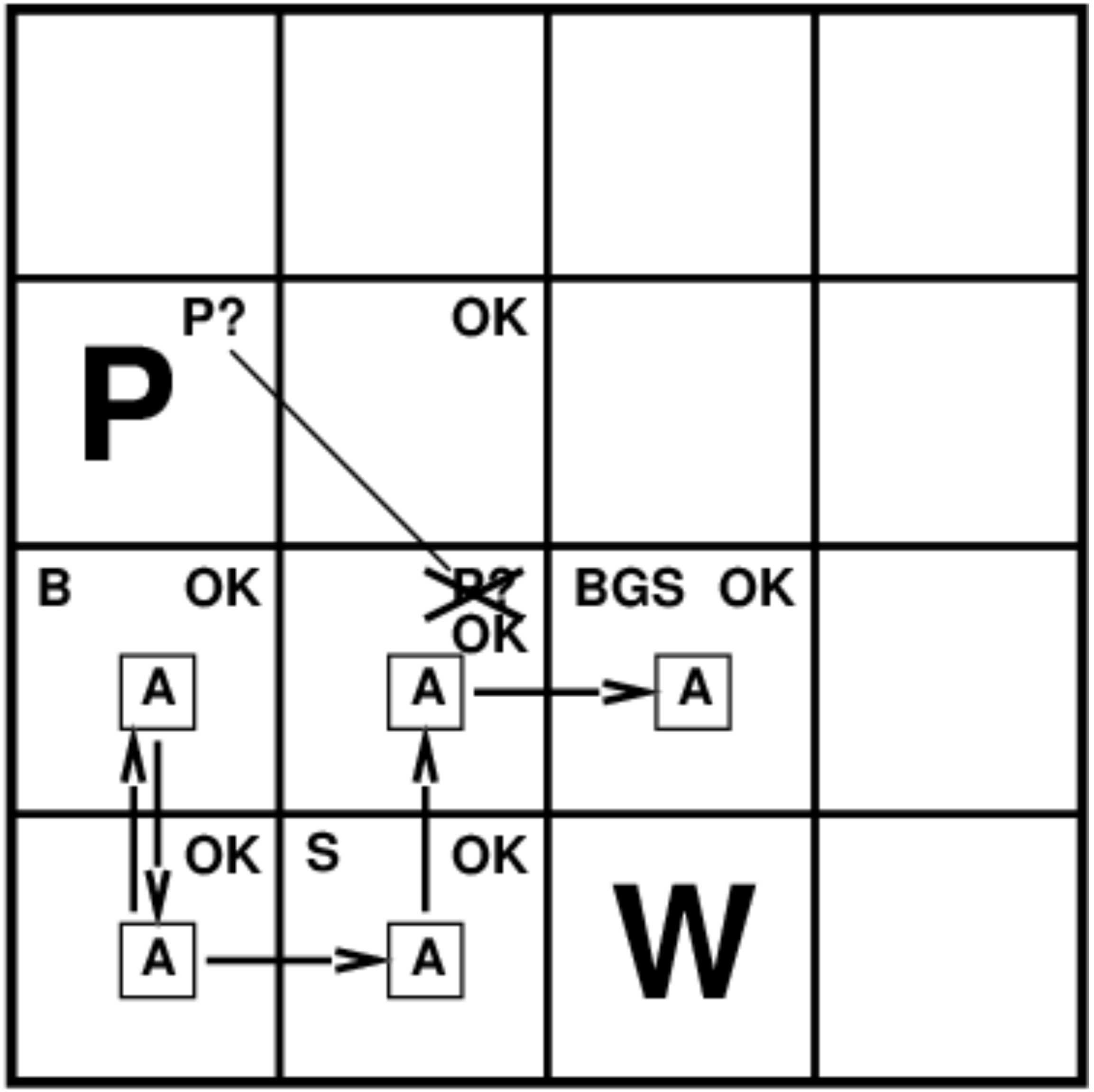
Exploring a Wumpus World



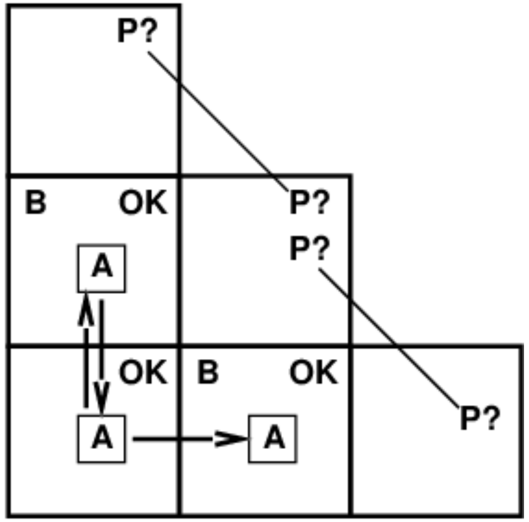
Exploring a Wumpus World



Exploring a Wumpus World



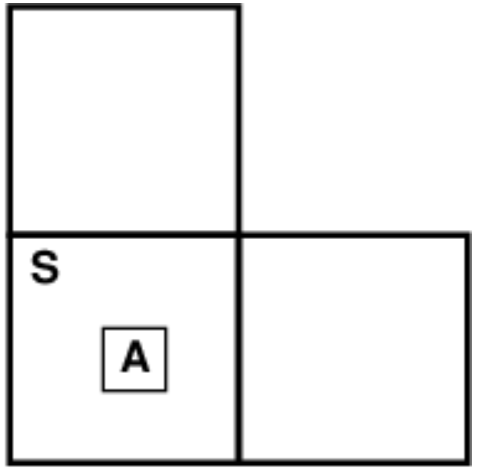
Sometimes No Safe Move Exists



Breeze in (1,2) and (2, 1)

⇨no safe actions

Assuming pits uniformly distributed, (2,2) has a pit with higher probability. How much?



Smell in (1, 1)

⇨cannot move

Can use strategy of coercion:

shoot straight ahead

Wumpus was there ⇨ dead ⇨ safe

Wumpus wasn't there ⇨ safe

Logic in General

- **Logics** are formal languages for representing information such that conclusions can be drawn.
- **Syntax** determines how sentences are expressed in a particular logic/language
- **Semantics** define the “meaning of sentences”;
  - 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
  - $x + 2 \geq y$ is true iff the number $x + 2$ is no less than the number y
  - $x + 2 \geq y$ is true in a world where $x = 7, y = 1$
  - $x + 2 \geq y$ is false in a world where $x = 0, y = 6$

Types of Logic

- **Logics** are characterized by what they commit to as primitives
- Ontological commitment: what exists – facts? Objects? Time? Beliefs?
- Epistemological commitment: what states of knowledge?

Language	Ontological Commitment	Epistemological Commitment
Propositional logic	Facts	true/false/unknown
First-order logic	Facts, objects, relations	true/false/unknown
Temporal logic	Facts, objects, relations, time	true/false/unknown
Probability theory	Facts	Degree of belief
Fuzzy logic	Facts + degree of truth	Known internal value

Reasoning with Logic

- First order of business: fundamental concepts of logic representation. And reasoning
  - independent of any logic's particular form/type
  - **Entailment**

- Second order of business: Introduction to **propositional logic**

  Wumpus KB via propositional logic

- Third order of business: Drawing conclusions
  - **Inference** and theorem proving

Models

- Can use the term **model** in place of possible world
- Logicians typically think in terms of **models**, which are formally-structed worlds with respect to which truth can be evaluated
- Model = mathematical abstraction that fixes the truth/falsehood of every relevant sentence
- Possible models are just all possible assignments of variables in the environment
- We say that a model ***m*** "satisfies" sentence α if α "is true in" **m**

  Or: "**m** is a model of α"

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

Models and Entailment

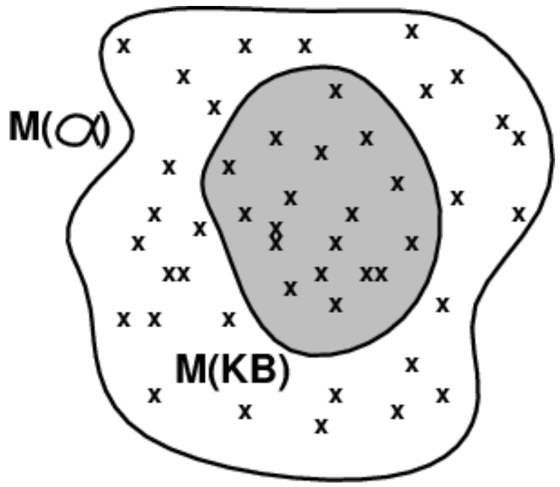
Entailment means that one thing follows from another:

$$KB \vDash \alpha$$

Knowledge base **KB** entails sentence α

iff α is true in all worlds/models where **KB** is true

KB $\vDash \alpha$ iff **M(KB)** $\subseteq$ **M(**α**)**



e.g., KB contains “Giants won” and “Red won” entails ”Giants or Reds won”

x + y = 4 entails **4 = x + y**

Entailment is a relationship between sentences (i.e., **syntax**) that is based on **semantics**.

Note: brains process **syntax** (or some sort).

Quick Exercise

Given two sentences α and β, what does this mean:

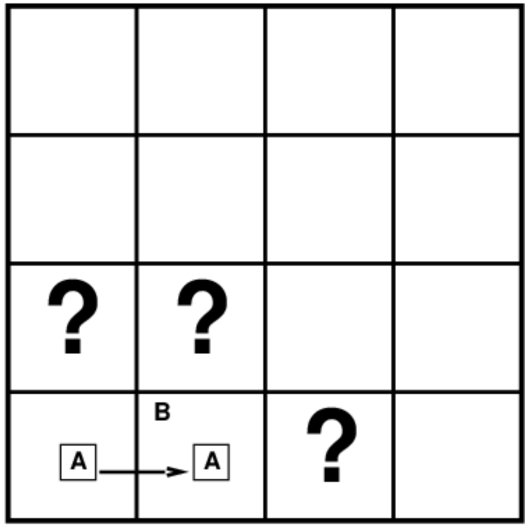
$$\alpha \vDash \beta$$

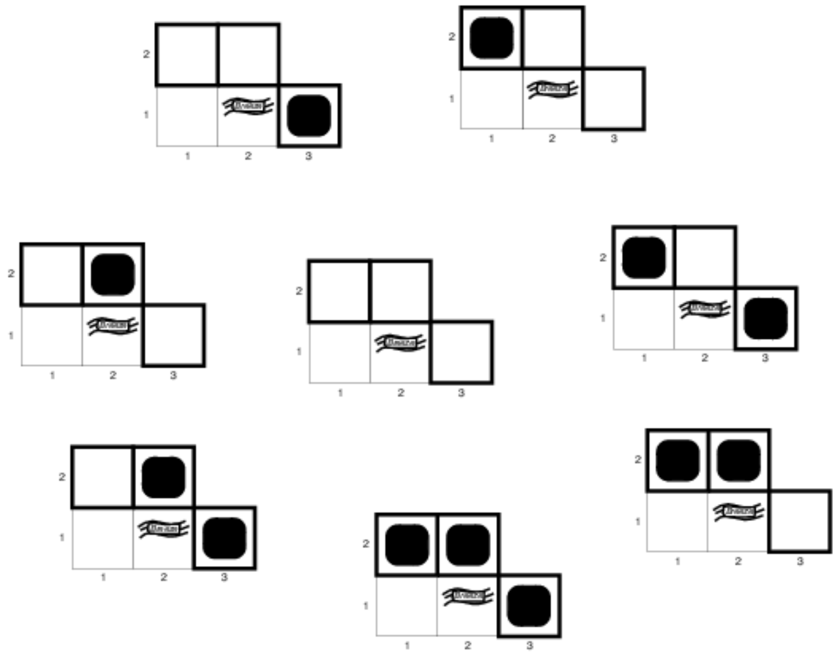
- α entails β
- **M(α) ⊆ M(β)**
- β is satisfied in all models of α.
- β may be satisfied in other models (as well).
- α is stronger assertion than β.

Hands on: Entailment in the Wumpus World

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

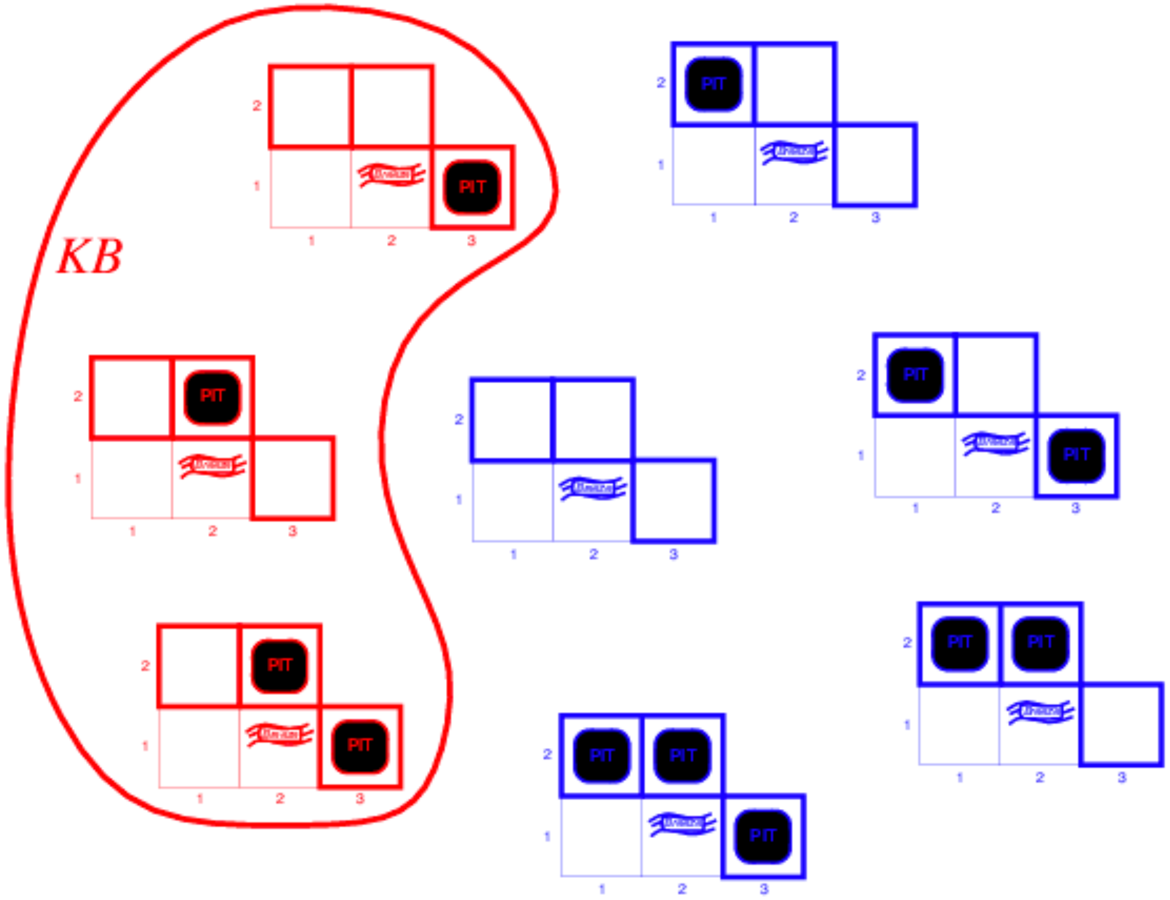
Consider possible models for ?s assuming only pits





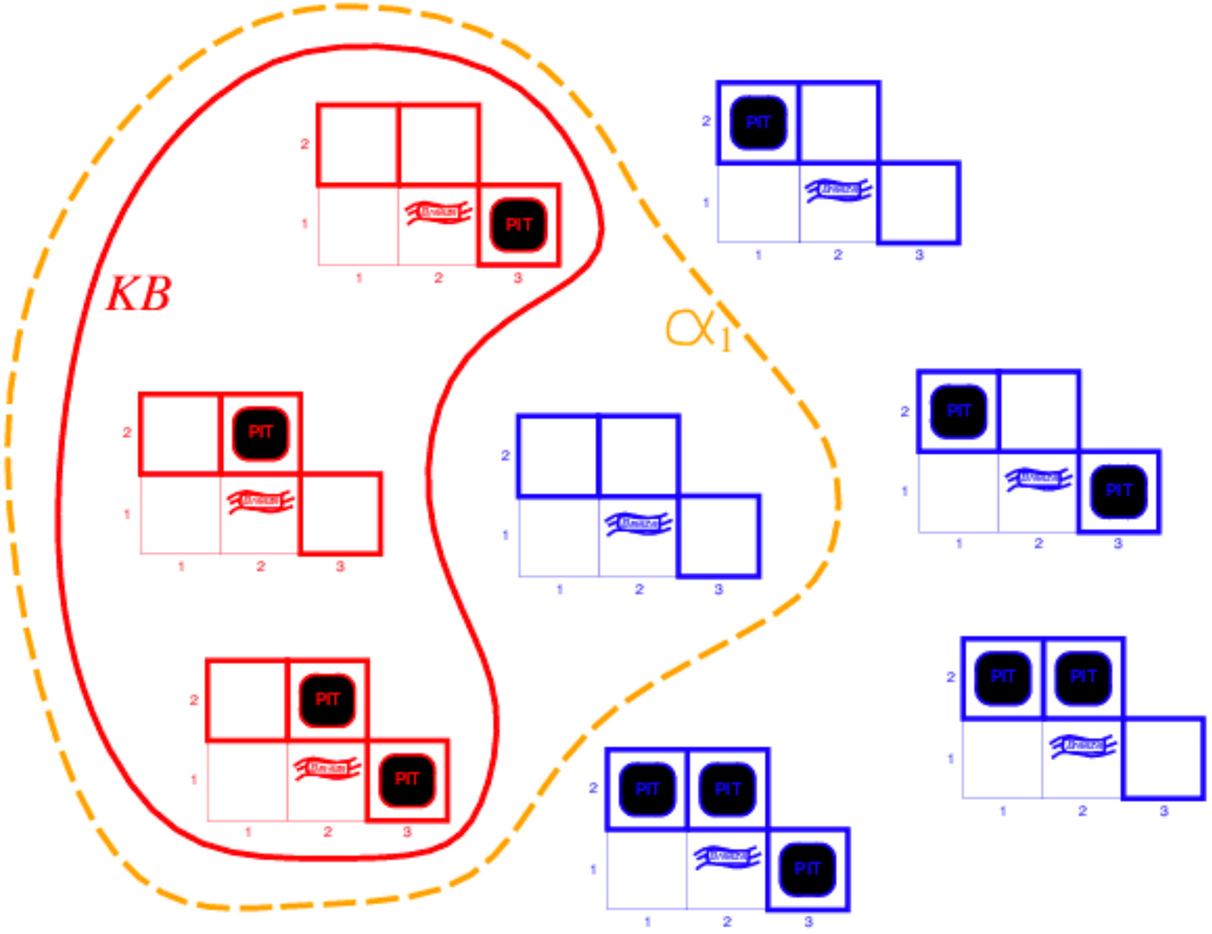
3 Boolean choices ⇨ 8 possible models

Wumpus Models



KB = Wumpus-world rules + observations

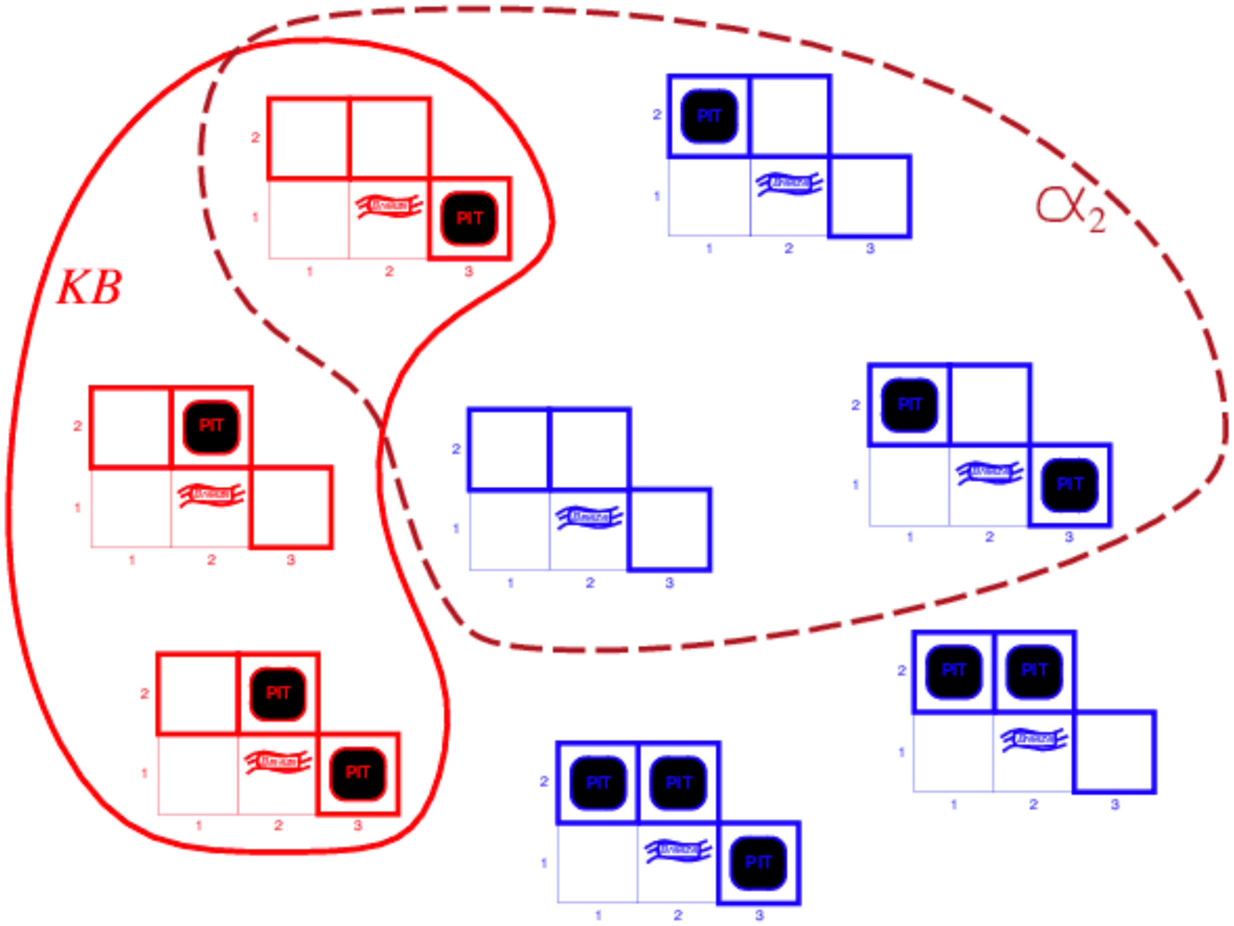
Wumpus Models



KB = Wumpus-world rules + observations, α_1 = “[1,2] is safe”

KB ⊨ α_1, proved by **model checking**

Wumpus Models



KB = Wumpus-world rules + observations, α_2 = "[2,2] is safe",

KB $\nvDash \alpha_2$

From Entailment to Logical Inference

Entailment can be used to derive logical conclusions

i.e., **carry out logical inference**

A straightforward algorithm to carry out inference:

Model Checking

Model checking enumerates all possible models to check that α is true in all models where **KB** is true. i.e**., M(KB) ⊆ M(α)**

To understand entailment and inference: haystack and needle analogy

Consequences of ***KB*** are a haystack, α is a needle.

Entailment = needle in haystack

Inference = finding it

We need inference procedures to derive α from a given KB.

Inference

KB $\vdash_i$ α = sentence α can be derived from *KB* by procedure *i*

Soundness: inference procedure ***i*** is sound if

whenever ***KB*** $\vdash_i$ α, it is also true that KB $\models$ α (does not make stuff up)

Completeness: inference procedure ***i*** is complete if

whenever **KB** $\models$ α, it is also true that **KB** $\vdash_i$ α (finds the needle in haystack)

Preview: we will define a logic (first-order logic) which is expressive enough to say almost anything of interest, and for which there exists a sound and complete inference procedure.

That is, the procedure will answer any question whose answer follows from what is known by the **KB**.

First step, propositional logic.

Propositional Logic: Syntax

Propositional logic is the simplest logic – illustrates basic ideas

Atomic sentences consist of a single propositional symbol

e.g., Propositional symbols $\mathbf{P_1}$, $\mathbf{P_2}$, etc. are atomic sentences

Each such symbol stands for a proposition that can be true or false.

e.g., W1,3 stands for proposition that Wumpus is in [1,3]

Two propositions with fixed meaning: *True* and *False*

Complex sentences build over atomic ones via connectives:

negation, conjunction, disjunction, implication, biconditional

Propositional Logic: Syntax

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

A (**positive**) literal is an atomic sentence; a (negative) literal is a negated atomic sentence

If S_1 and S_2 are sentences $S_1 \land S_2$ is a sentence (**conjunction**)

S_1 and S_2 are called conjuncts

If S_1 and S_2 are sentences $S_1 \lor S_2$ is a sentence (**disjunction**)

S_1 and S_2 are called disjuncts

If S_1 and S_2 are sentences, $S_1 \Longrightarrow S_2$ is a sentence (**implication/ conditional**)

S_1 is called premise/antecedent; S_2 is called conclusion or consequent

Implication also known as **rule** or **if-then** statement

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

Propositional Logic: Semantics (Backus-Naur Form (BNF)

BNF is an ambiguous formal grammar for propositional logic

Sentence → AtomticSentence | ComplexSentence

AtomicSentence → True | False | P | Q | …

ComplexSentence → (Sentence) | [Sentence] | ¬Sentence | Sentence ∧ Sentence …

We add operator precedence to disambiguate

Operator precedence (from highest to lowest): ¬, ∧, ∨, ⟹, ⇔

JAMES MADISON UNIVERSITY

Propositional Logic: Semantics

Each model specifies true/false for each propositional symbol

e.g.,

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

This specific model: $m_1 = \{P_{1,2} = \text{true}, P_{2,2} = \text{true}, P_{3,1} = \text{false})$

$2^3 = 8$ possible models, feasible to enumerate

Rules for evaluating truth with respect to a model m:

$\neg S$	is true iff	S	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
i.e.	is false iff	S_1	is true **and**	S_2	is false
$S_1 \Leftrightarrow S_2$	is true iff	$S_1 \Rightarrow S_2$	is true **and**	$S_2 \Rightarrow S_1$	is true

$\neg P_{1,2} \wedge (P_{2,2} \vee p_{3,1})$ = true $\wedge$ (false $\vee$ true) = true $\wedge$ true = true

JAMES MADISON UNIVERSITY

Truth Table for Connectives

P	Q	¬P	$P \wedge Q$	$P \vee Q$	$P \Longrightarrow Q$	$P \Leftrightarrow Q$
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

Wumpus World Sentences in Propositional Logic

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

Let $B_{i,j}$ be true if agent is in [i, j] and perceives a breeze

Let $W_{i,j}$ be true if there is a Wumpus in [i, j]

Let $S_{i,j}$ be true if agent is in [i, j] and perceives a stench

… you can define other atomic sentences

Percept sentences part of KB:

No pit, no breeze in [1, 1], but breeze perceived when in [2, 1]

$R_1 : \neg P_{1,1}$; $R_4 : \neg B_{1,1}$ $R_5 : B_{2,1}$

Rules in KB: “Put cause breezes in adjacent squares” eqv. to “Square is breeze **iff** adjacent pit”.

$R_2 : B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

$R_3 : B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

Truth Tables for Inference: Model Checking by Enumeration

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$		R_1	R_2	R_3	R_4	R_5		KB
false false … false	false false … true	False False … false	False False … false	False False … false	False False … false	False True … false		true True … true	True True … true	true False … false	true True … true	false False … true		False False … false
False False False	true True true	False False False	False False false	False False false	False True true	True False true		True True true	True True true	True True true	True True true	True True true		True True true
False … true	True … true	False … true	False … true	True … true	False … true	False … true		True … false	False … true	False … true	True … false	True … true		False … false

Enumerate rows (different assignments to symbols); rows are possible models

If KB is true in a row/model, check that α is true, if not, entailment does not hold

If entailment not broken over all rows where *KB* is true, then α is entailed

Inference by Enumeration

```
function TT-Entails?(KB, α) returns true/false
    Inputs:     KB (the knowledge base), a sentence in proposition logic
                α, the query, a sentence in propositional logic
    symbols ← a list of the propositional symbols in KB and α
    Return TT-Check-All(KB, α, symbols,[])
```

```
function TT-Check-All(KB, α, symbols, []) returns true/false
    if Empty?(symbols) then
        if PL-TRUE?(KB, model) then return PL-TRUE(α, model)
        else return true;
    else do
        P ← First(symbols); rest ←Rest(symbols)
        return TT-Check-All(KB, α, rest, Extend(P, true, model)) and
                    TT-Check-All(KB, α, rest, Extend(P, false, model))
```

$O(2^n)$ for *n* symbols, problem is co-NP-complete.

Proof Methods

Proof methods divide into (roughly) two kinds:

Model checking:

- Truth table enumeration (always exponential in n)
- Improved backtracking, e.g., Davis-Putnam-Logemann-Loveland
- Backtracking with constraint propagation, backjumping
- Heuristic search in model space (sound but incomplete) e.g., min-conflicts, etc.

Theorem Proving/Deductive Systems: Application of inference rules

- Legitimate (sound) generation of new sentences from old
- Proof = a sequence of inference rule applications
- Typically requires translation of sentence into a **normal form**

End of first part of Chap 7

Logical Equivalence

Two sentences are logically equivalent iff true in same models:

$\alpha \equiv \beta$ if and only if a $\vDash \beta$ and $\beta \vDash \alpha$

$(\alpha \wedge \beta)$

Proof = a sequence of inference rule applications

Typically requires translation of sentence into a normal form