PURDUE CS47100

INTRODUCTION TO AI

ANNOUNCEMENT

- Assignment 2 is out!
- Based on the poll results, the due date will be Sunday, October 16, the end of day
- Work on the programming part first and work on the written part as we cover the content in the next 4 classes!

RECAP: CSP

- General-purpose heuristics
  - Filtering: forward checking, constraint propagation, k-consistency
  - Ordering: minimum remaining value heuristic, degree heuristic, least constraining value heuristic
  - Structure: independent subproblems, tree structure, cutset conditioning

LOGIC AND REASONING

KNOWLEDGE BASE

- Knowledge Base (KB): A set of ***sentences*** in a formal language
- TELL the agent what it needs to know
- The agent can then ASK itself what to do
- The answers should follow from the KB

$\forall x\ Bird(x) \Rightarrow Fly(x)$ “For all objects that exist in the world, if the object is a bird, then the object can fly.”

Penguins can’t fly! Must be very careful when constructing KB!

purdue university

All · News · Images · Maps · Videos · More · Tools

About 131,000,000 results (0.80 seconds)

Purdue University

West Lafayette, IN · Public · 4-year

Overview · Admissions · Cost · Programs · Outcomes · Students

https://www.purdue.edu

Purdue University - Indiana's Land Grant University

Purdue University is a world-renowned, public research university that advances discoveries in science, technology, engineering and math.

Results from purdue.edu

Academics
Academic programs at Purdue prepare students to move the ...

Purdue Online
Our rigorous online programs allow you to earn a prestigious ...

About

purdue.edu

Purdue University is a public land-grant research university in West Lafayette, Indiana, and the flagship campus of the Purdue University system. The university was founded in 1869 after Lafayette businessman John Purdue donated land and money to establish a college of science, technology, and agriculture in his name. Wikipedia

Avg cost after aid	Graduation rate	Acceptance rate
$13K	81%	67%

Graduation rate is for first-time, full-time undergraduate more
From US Dept of Education · Learn more

Address: 610 Purdue Mall, West Lafayette, IN 47907

Phone: (765) 494-4600

Notable alumni

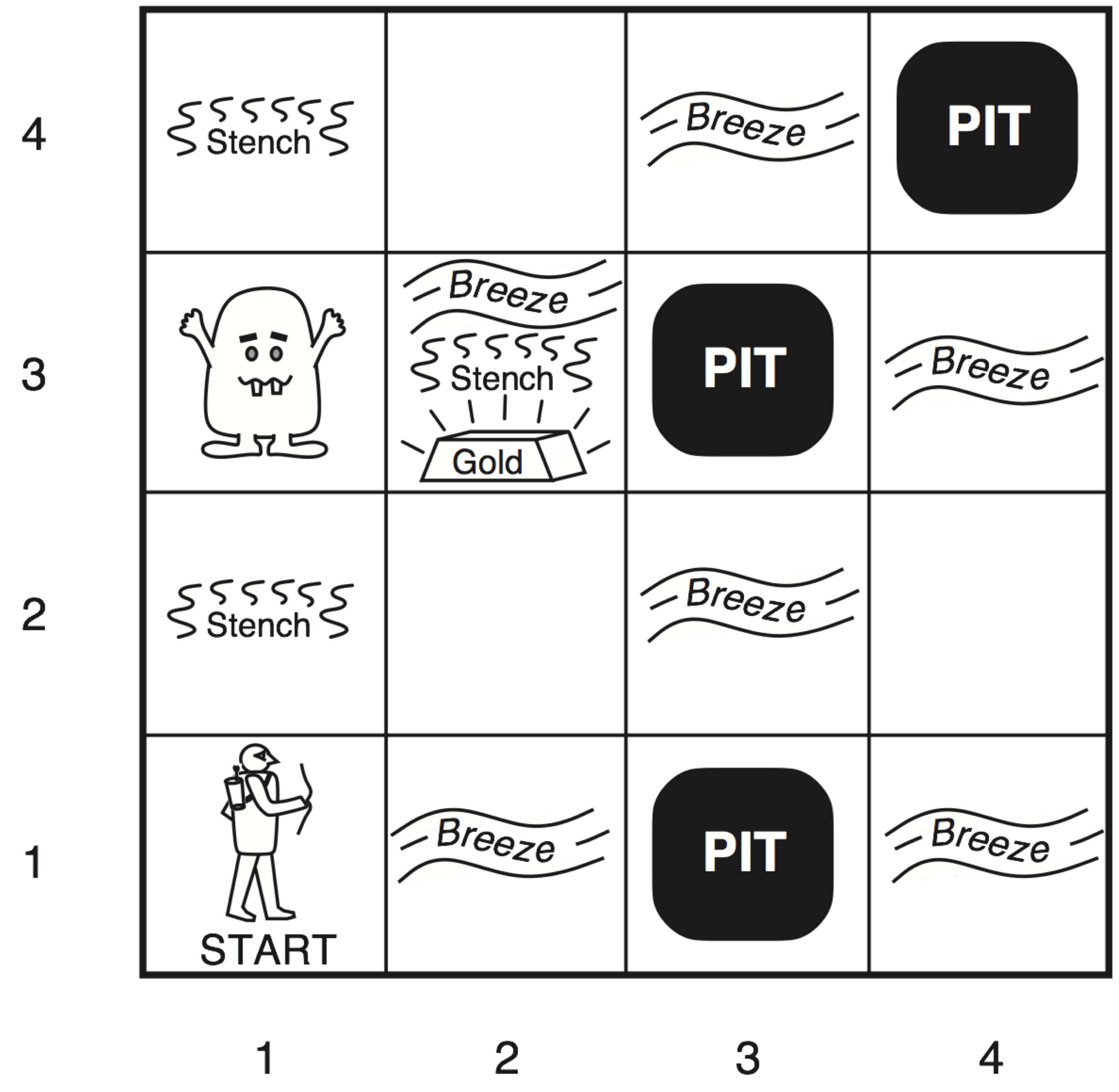
Neil Armstrong · Gene Cernan · Ted Allen · Gus Grissom

College facts

From US Dept of Education · Learn more

THE WUMPUS WORLD: TASK ENVIRONMENT CHARACTERISTICS

- **Performance Measure:**
  - Gold: +1000, Death: -1000; -1 per step, -10 for using 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
- **Sensors:** Breeze, Glitter, Smell
- **Actuators:**
  - Left turn, Right turn, Forward, Grab, Shoot

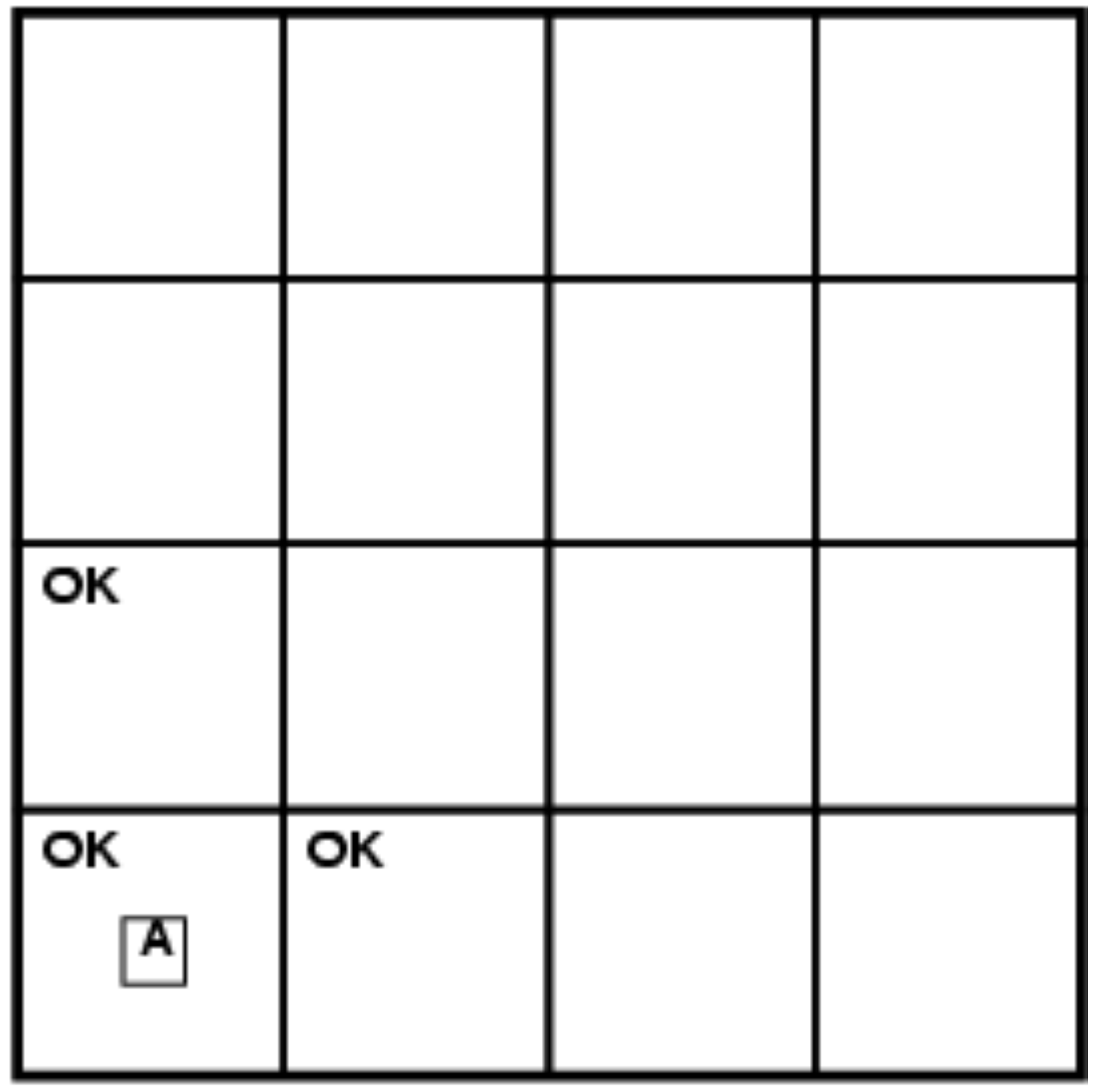


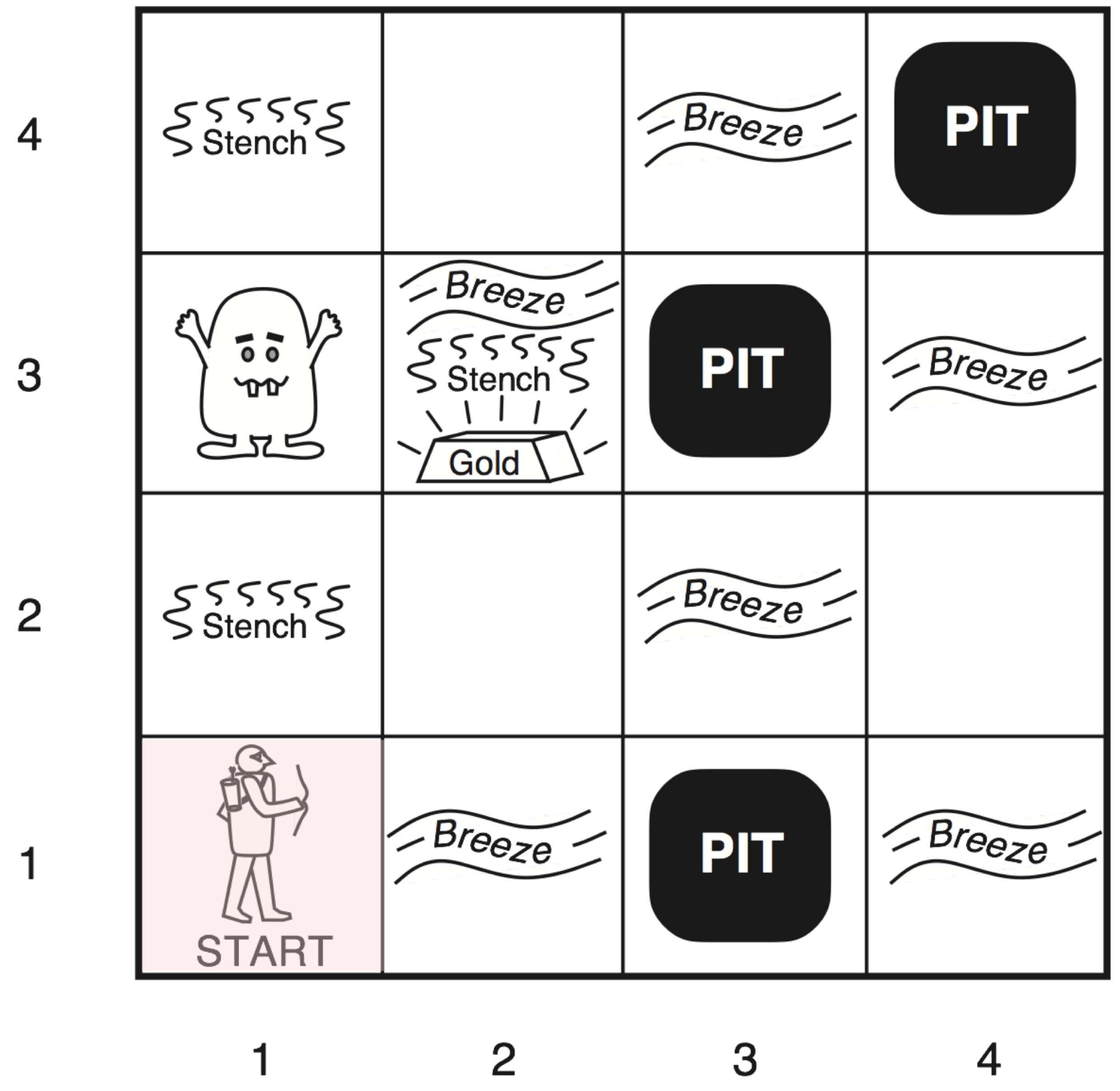
EXPLORING A WUMPUS WORLD

Squares adjacent to wumpus are smelly

Squares adjacent to pit are breezy

Glitter iff gold is in the same square



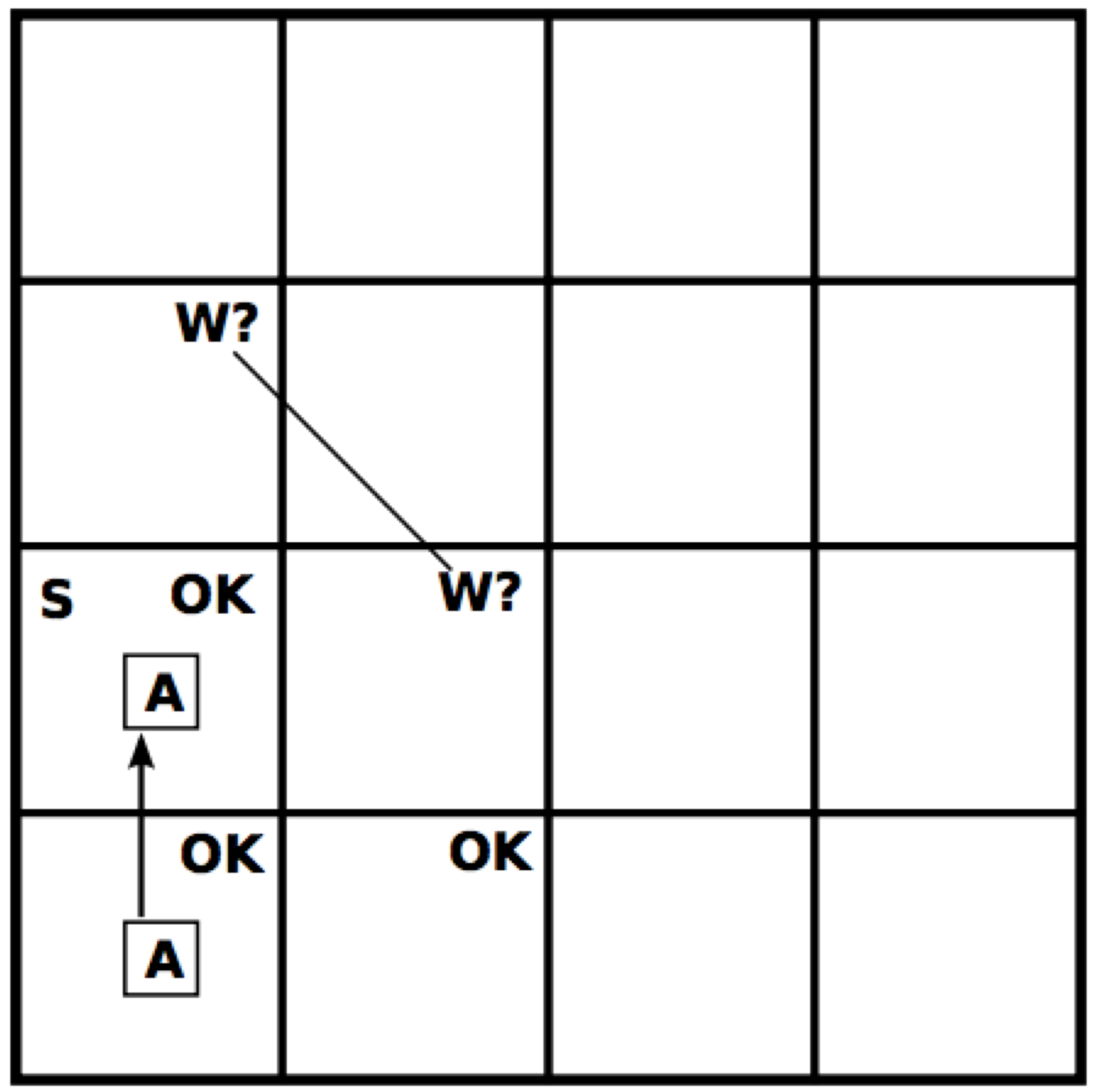


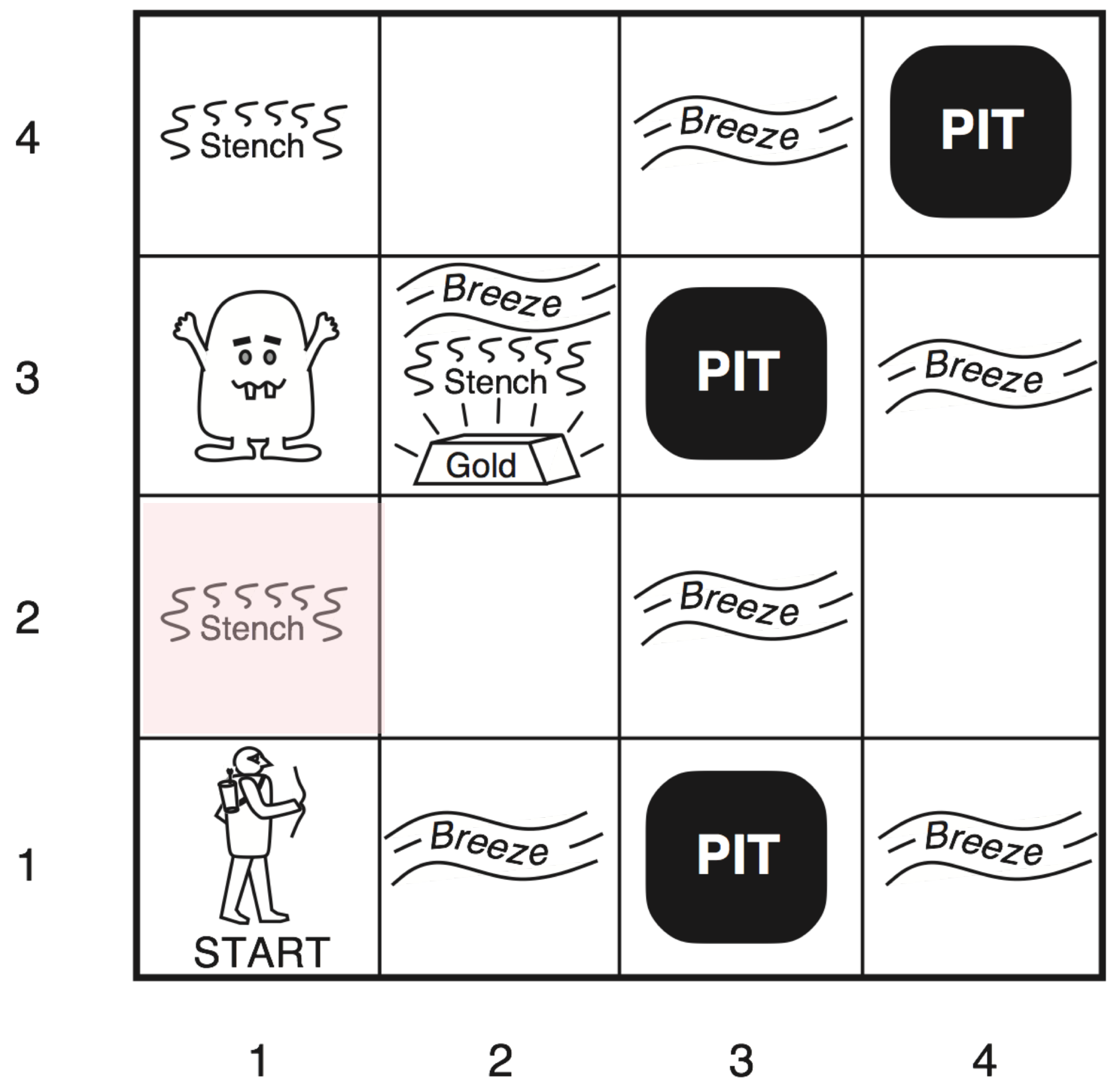
EXPLORING A WUMPUS WORLD

Squares adjacent to wumpus are smelly

Squares adjacent to pit are breezy

Glitter iff gold is in the same square



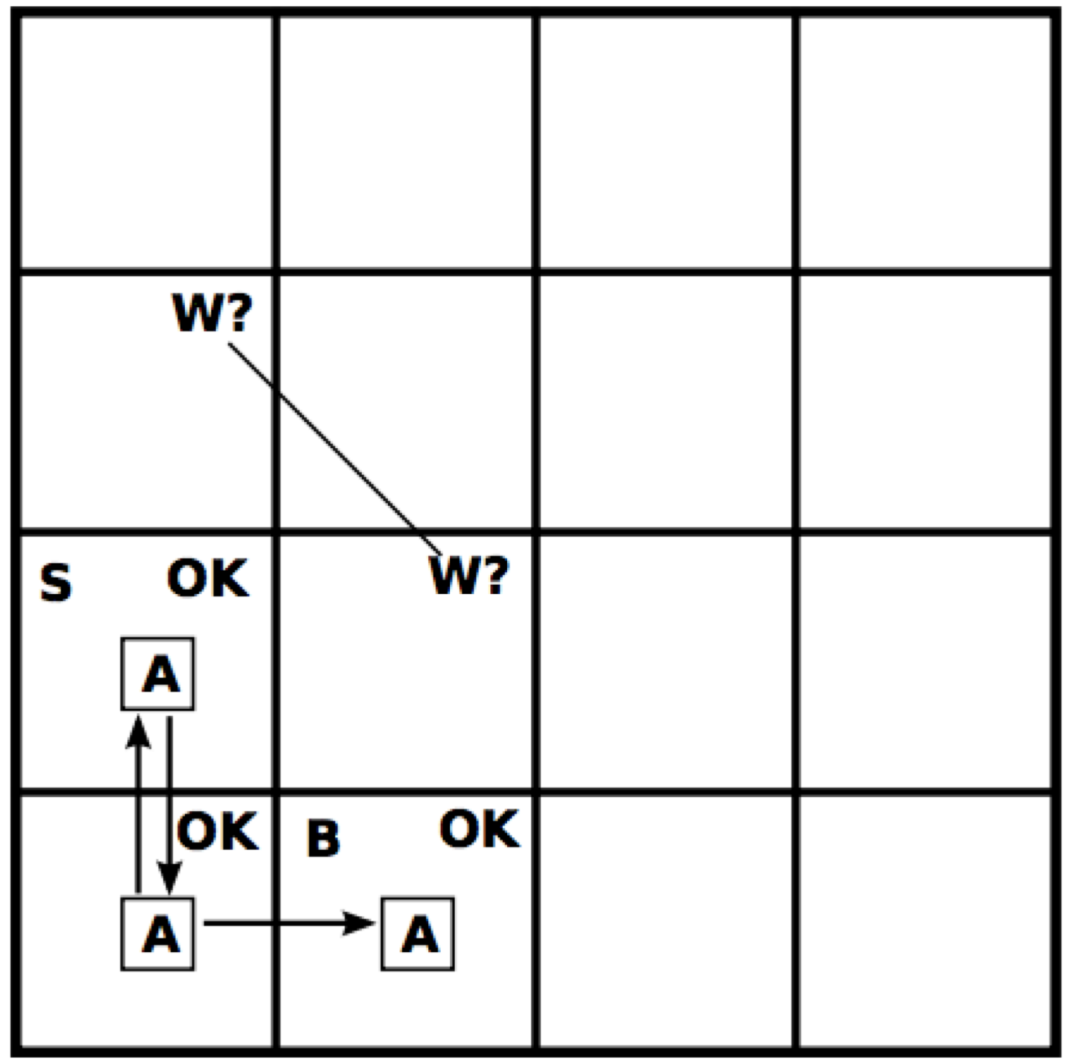


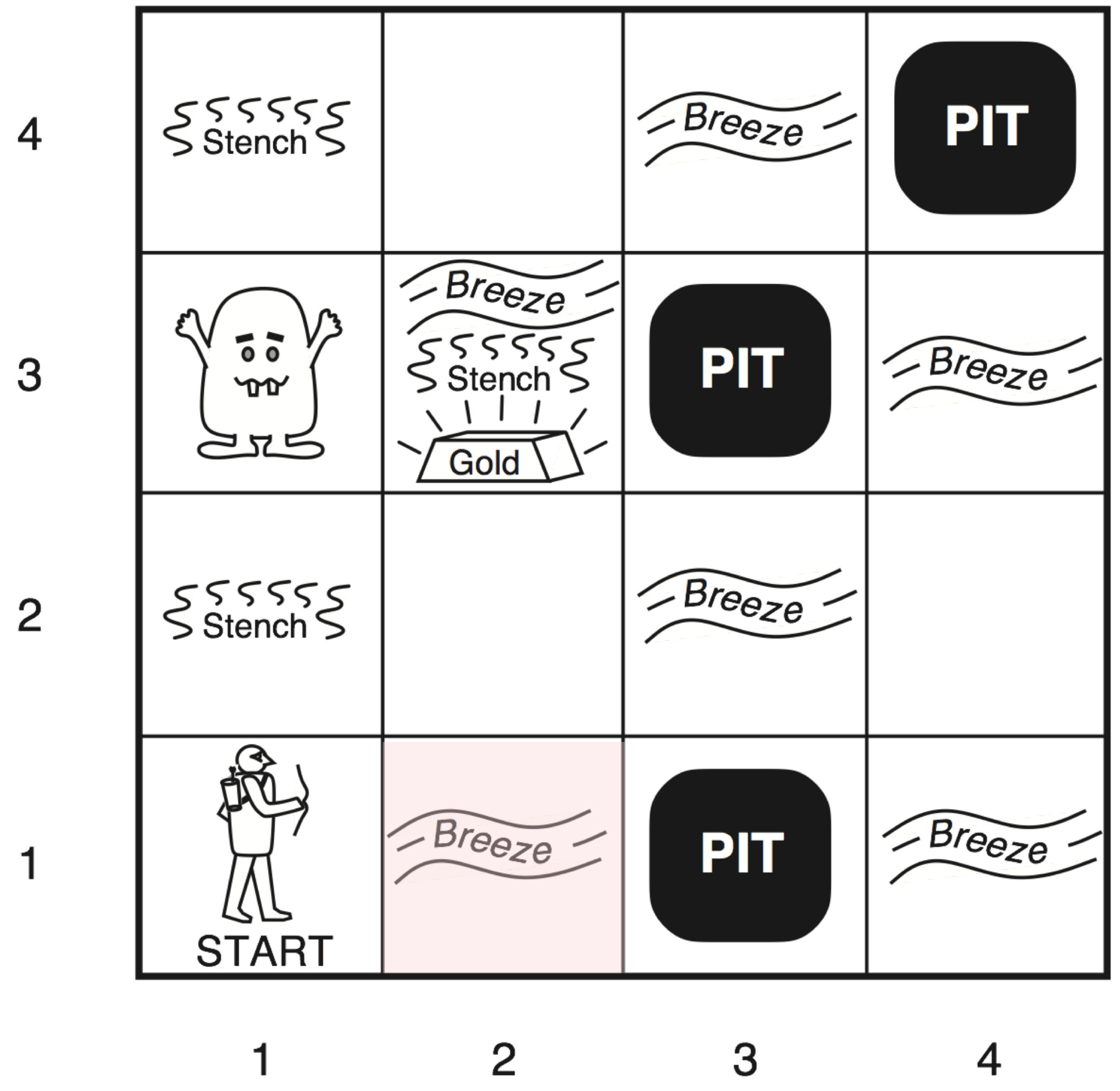
EXPLORING A WUMPUS WORLD

Squares adjacent to wumpus are smelly

Squares adjacent to pit are breezy

Glitter iff gold is in the same square



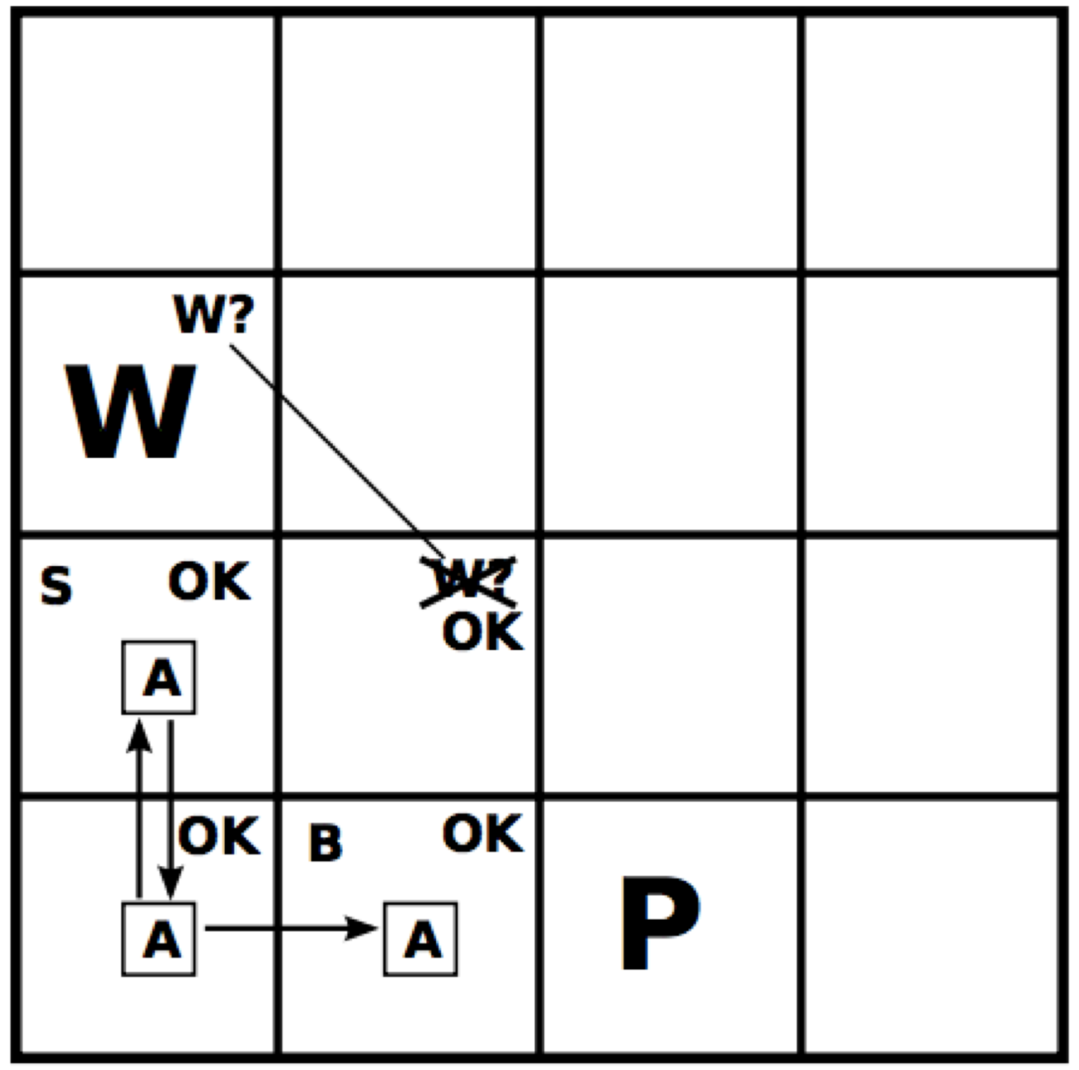


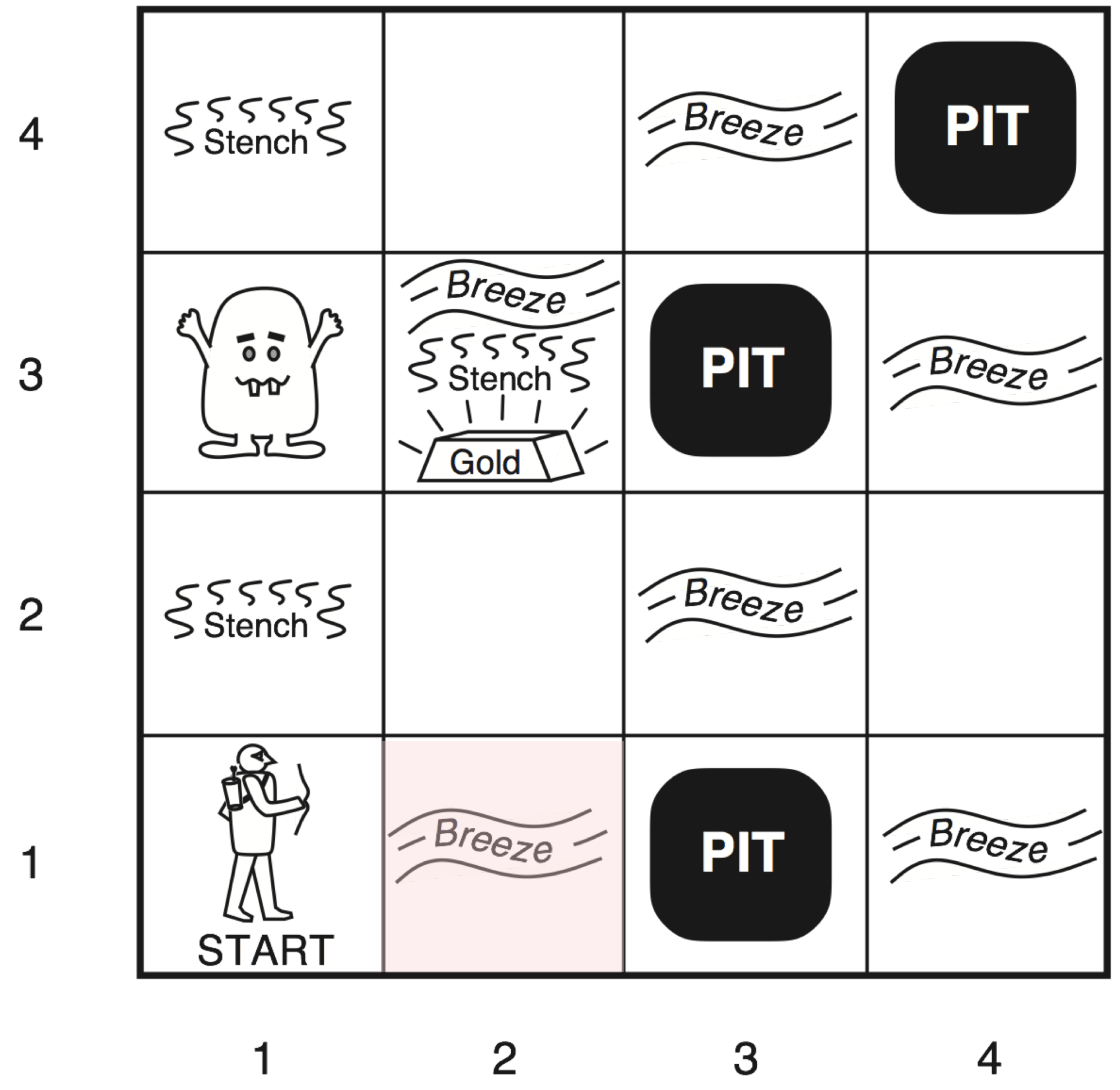
EXPLORING A WUMPUS WORLD

Squares adjacent to wumpus are smelly

Squares adjacent to pit are breezy

Glitter iff gold is in the same square



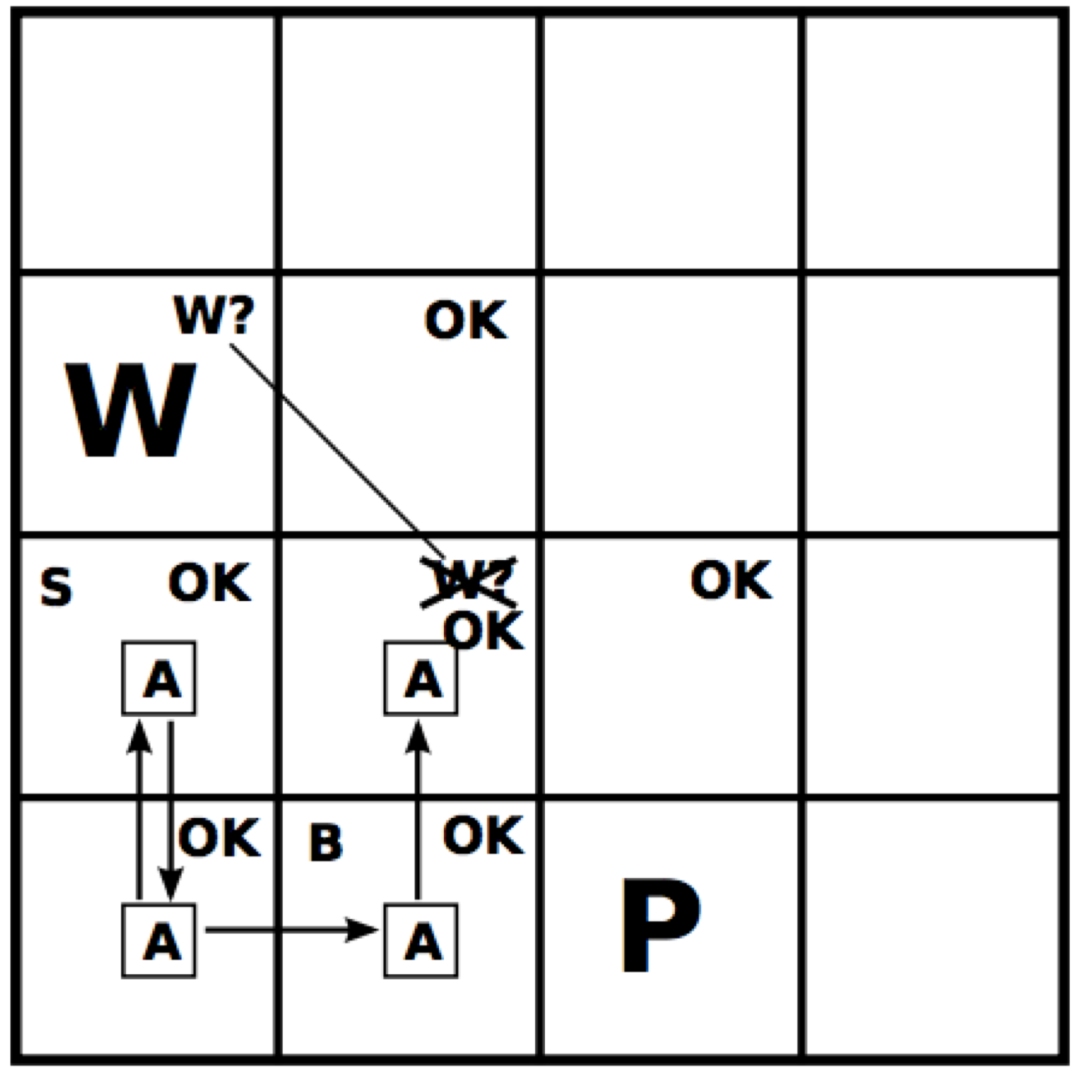


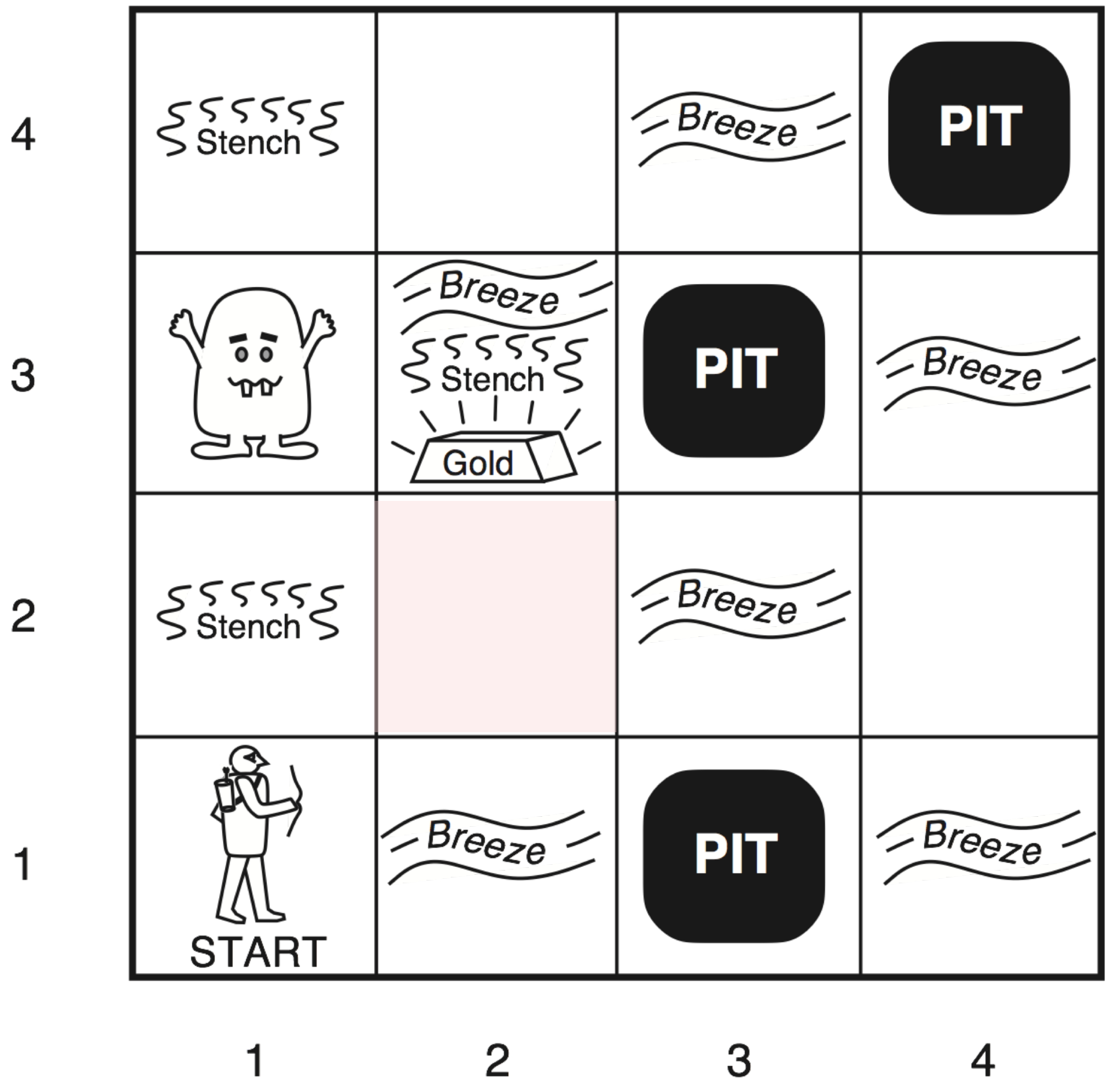
EXPLORING A WUMPUS WORLD

Squares adjacent to wumpus are smelly

Squares adjacent to pit are breezy

Glitter iff gold is in the same square





ENTAILMENT

- Entailment means one thing follows from another

$$KB \models \alpha$$

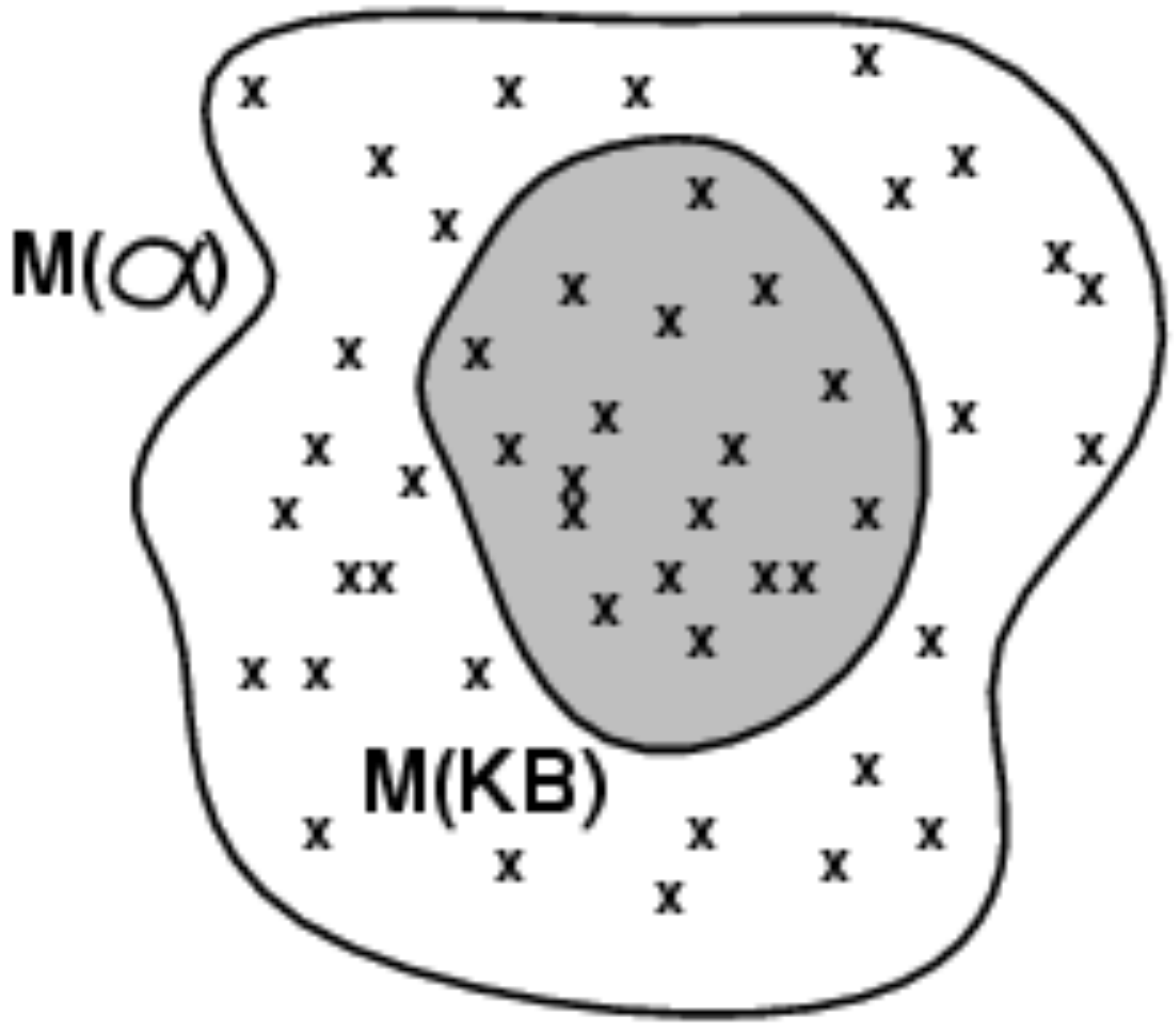
- Knowledge base KB entails sentence α if and only if α is true in all worlds where KB is true
    - $(x + y = 4) \models (4 = x + y)$
    - “I am hungry” entails “I am hungry or thirsty”
    - “There is no breeze here” entails “There is no pit in an adjacent square”

ENTAILMENT

- Models are formally structured possible worlds with respect to which truth can be evaluated
- M is a model of sentence α if α is true in M

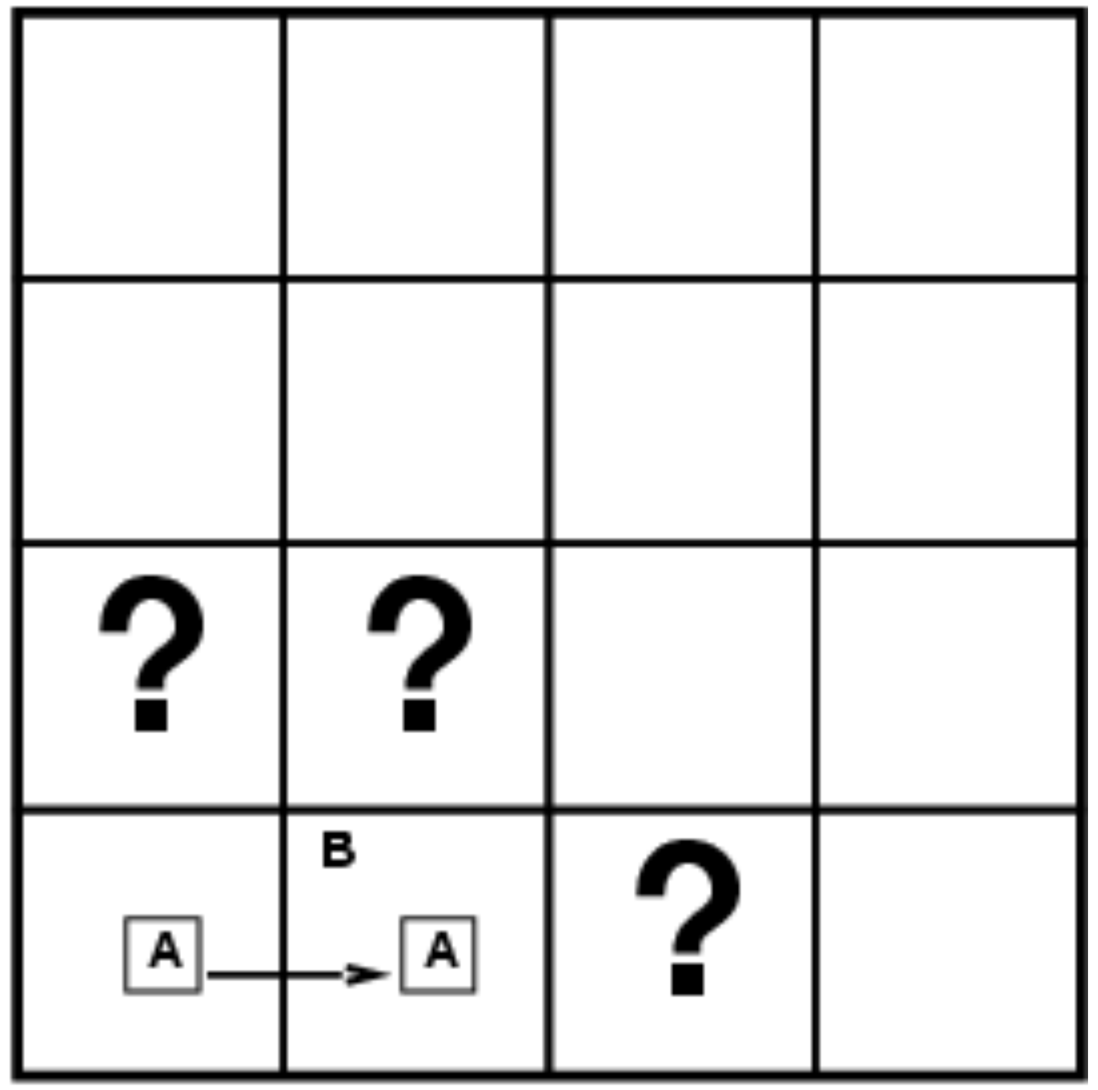
$$KB \models \alpha \text{ iff } M(KB) \subseteq M(\alpha)$$

- In English: Every world where KB is true, α is also true

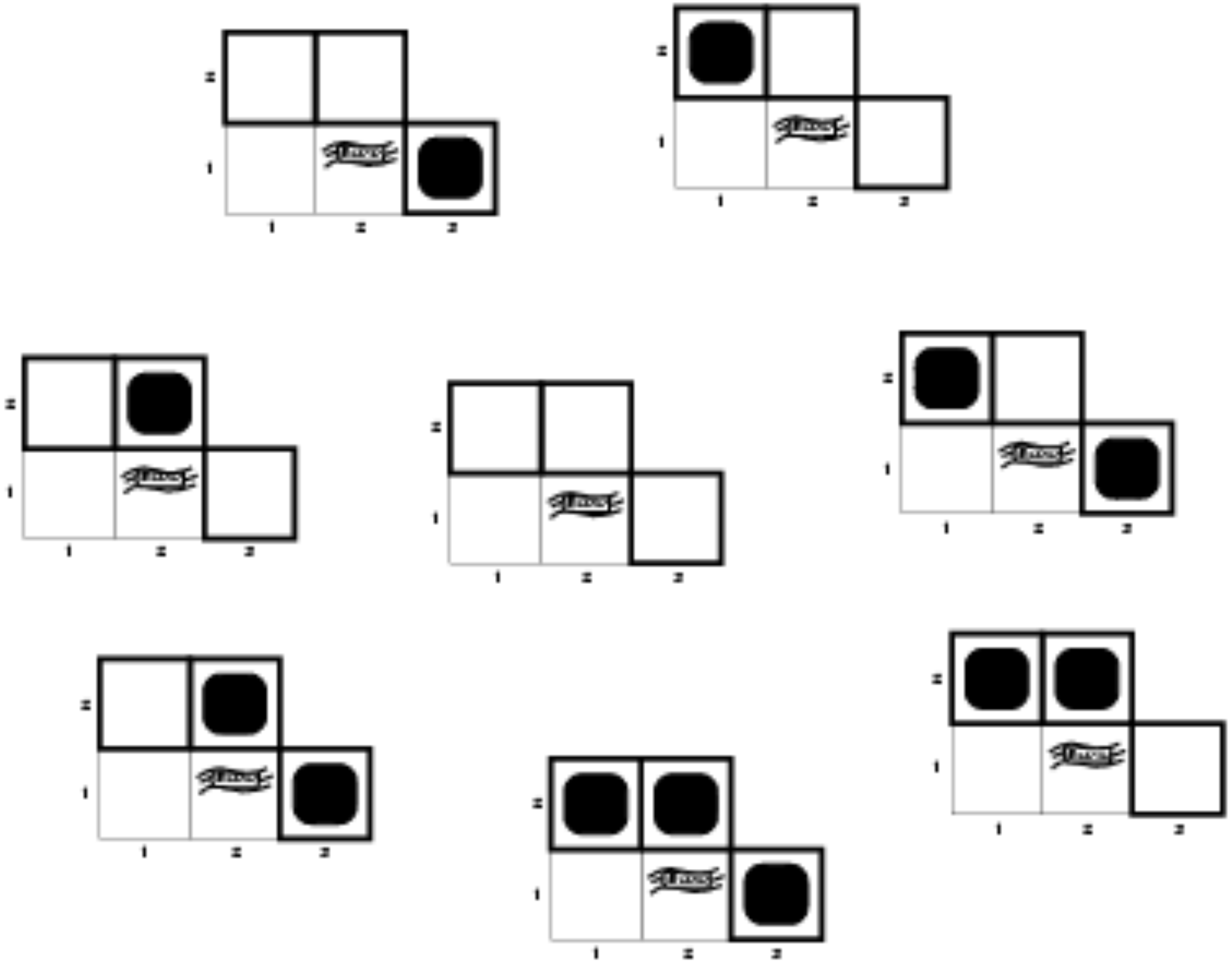


ENTAILMENT IN WUMPUS WORLD

- We detected nothing in square [1,1].
- Moved to the right
- There is a breeze in [2,1]
- What are the possibilities for the locations marked with ? ?
    - Consider only pits (don't worry about Wumpus for now)
    - Don't worry about reasoning yet – just list out the possibilities
    - There are three Boolean choices: $2^3 = 8$ *possible world models*

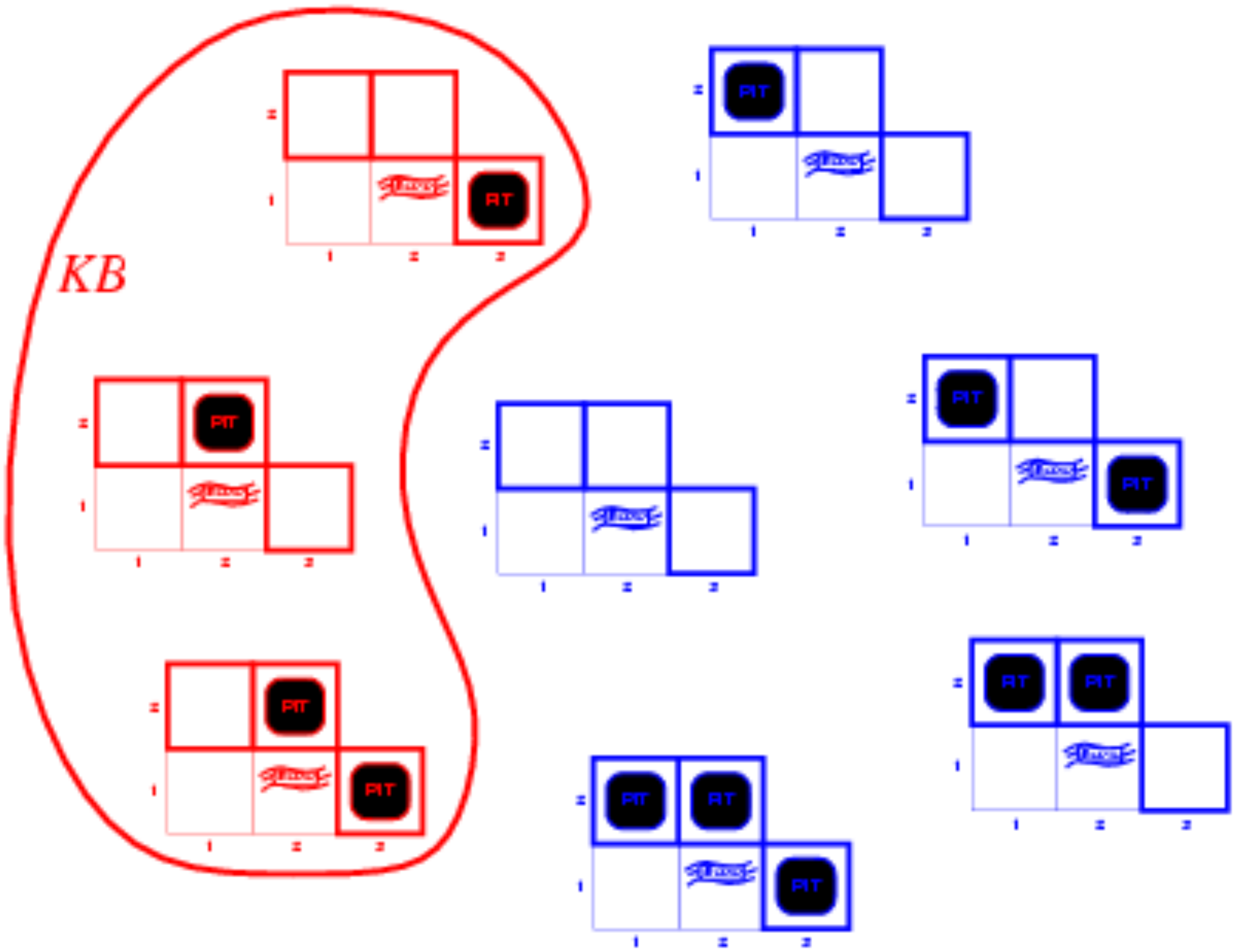


WUMPUS MODELS



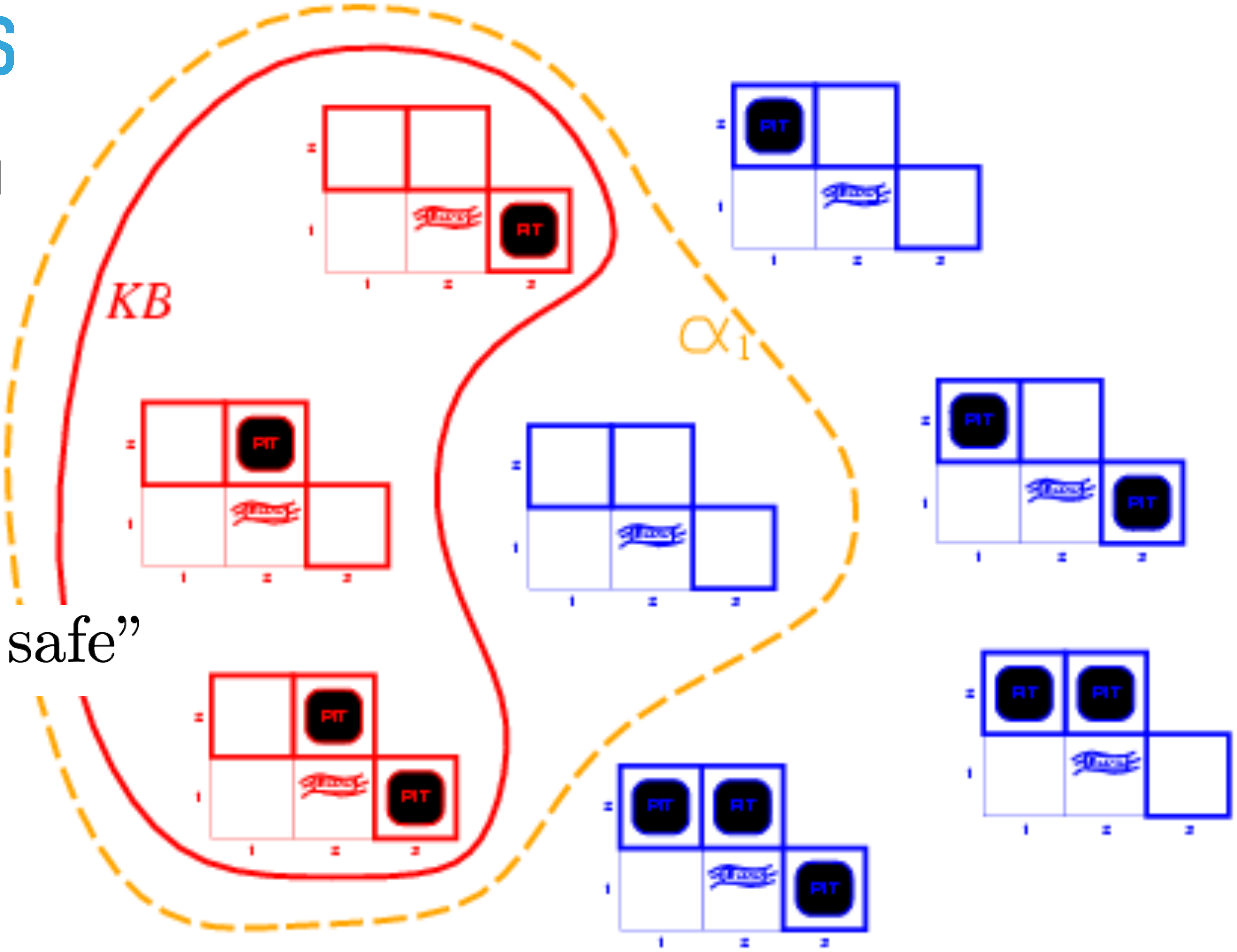
WUMPUS MODELS

KB = Wumpus world rules + observations



WUMPUS MODELS

KB = Wumpus world rules + observations

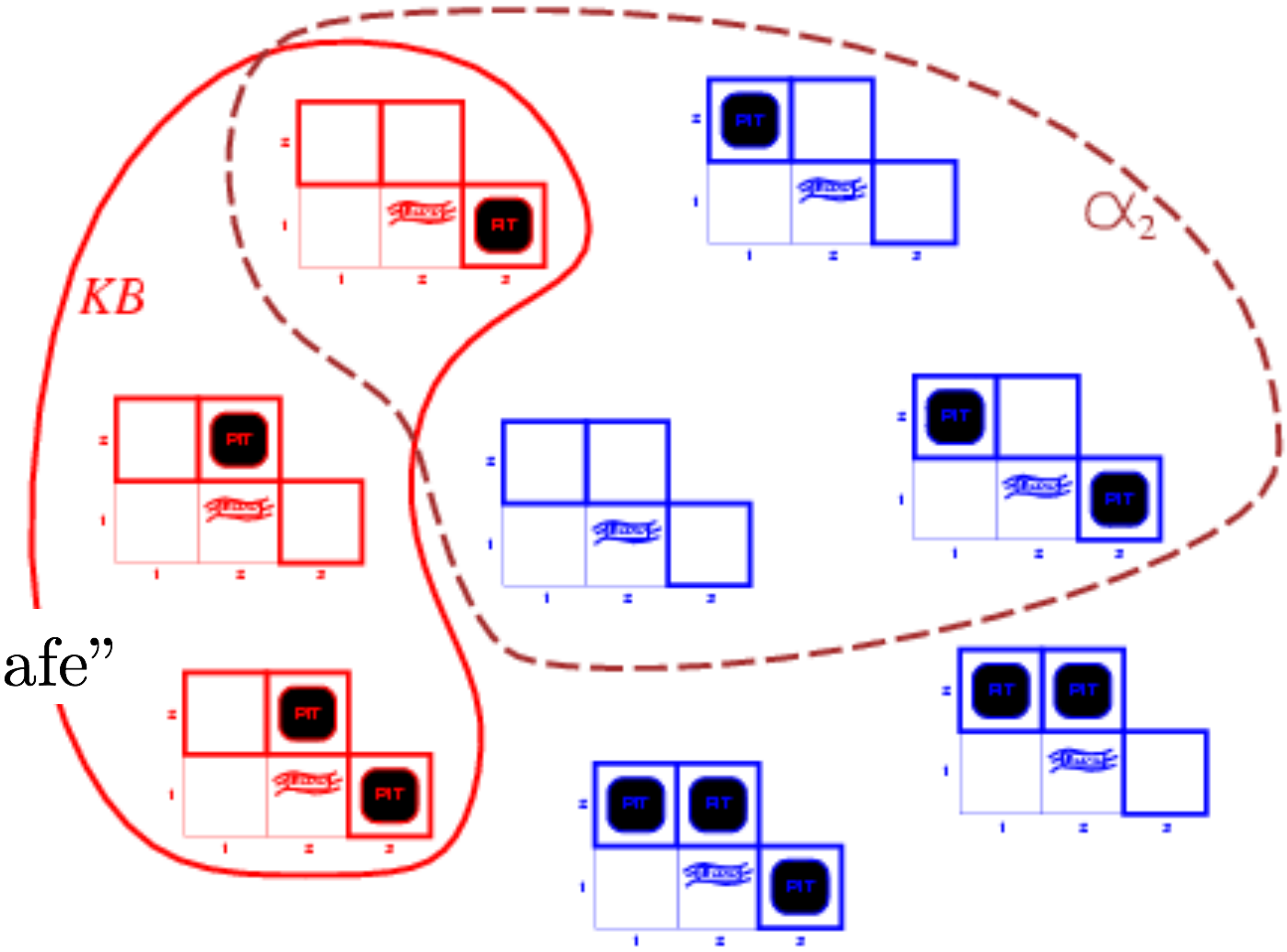


$$\alpha_1 = \text{“}[1,2] \text{ is safe”}$$

$$KB \models \alpha_1$$

WUMPUS MODELS

KB = Wumpus world rules + observations



$$\alpha_2 = \text{“[2,2] is safe”}$$

$$KB \not\models \alpha_2$$

INFERENCE

- KB $\vdash_i \alpha$: sentence α can be derived from KB by procedure i
- **Soundness**: i is sound if whenever KB $\vdash_i \alpha$, it is also true that KB $\models \alpha$
- **Completeness**: i is complete if whenever KB $\models \alpha$, it is also true that KB $\vdash_i \alpha$

- Preview: we will define a 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

PROPOSITIONAL LOGIC: SYNTAX

- Atomic sentences:
  - Proposition symbols $P_{1,1}$, $W_{2,2}$: A proposition that can be true or false
  - Always-true proposition or always-false proposition: True, False
- Complex sentences:
  - If S is a sentence, $\neg S$ is a sentence (**negation**)
  - If S1 and S2 are sentences, $S1 \wedge S2$ is a sentence (**conjunction**)
  - If S1 and S2 are sentences, $S1 \vee S2$ is a sentence (**disjunction**)
  - If S1 and S2 are sentences, $S1 \Rightarrow S2$ is a sentence (**implication**)
  - If S1 and S2 are sentences, $S1 \Leftrightarrow S2$ is a sentence (**biconditional**)

PROPOSITIONAL LOGIC: SEMANTICS

- Each model specifies true/false for each proposition symbol (P means pit at the given coordinates)
  E.g. $P_{1,2}$ $P_{2,2}$ $P_{3,1}$
  false true false

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

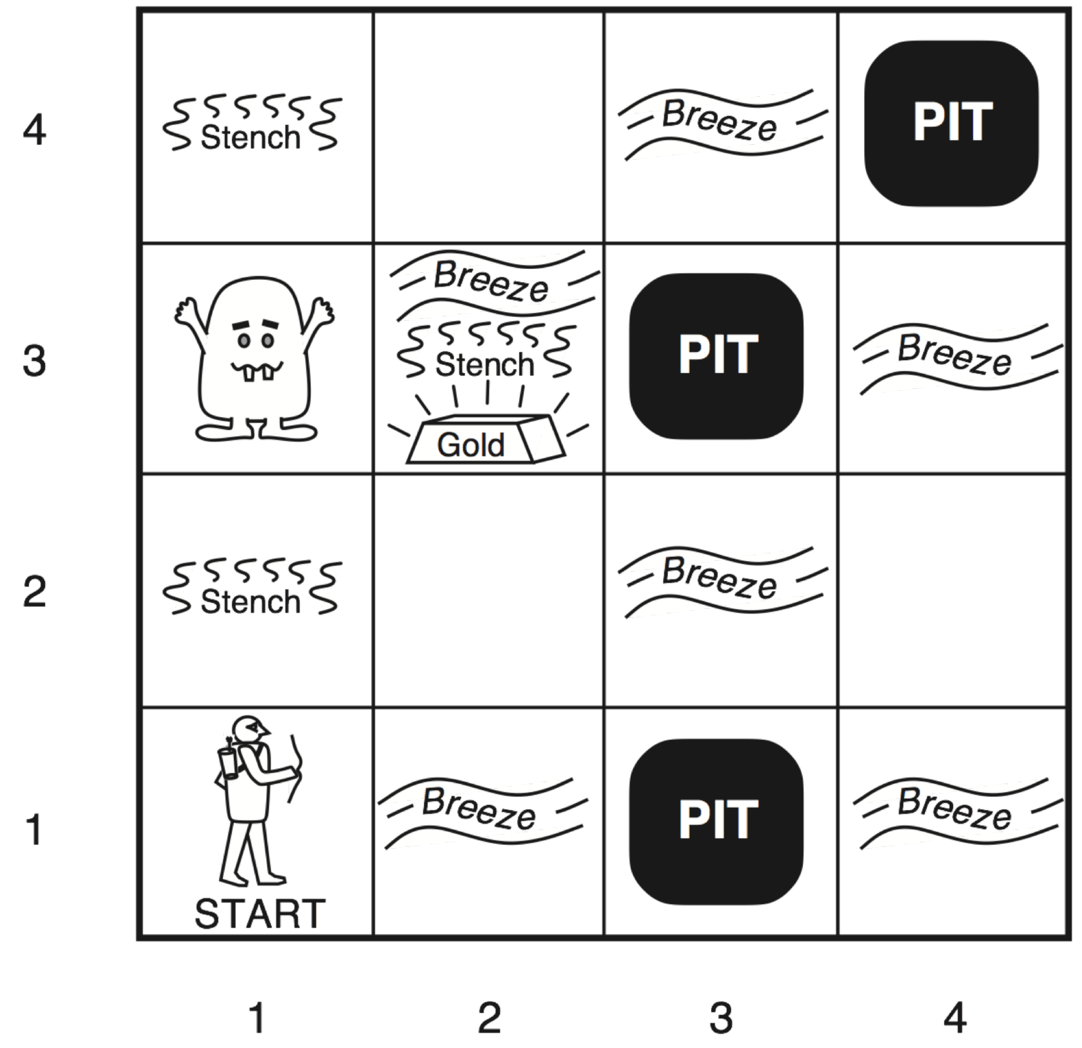
- Rules for evaluating truth value of complex sentences:
  - ¬S is true iff S is false
  - S1 ∧ S2 is true iff S1 is true **and** S2 is true
  - S1 ∨ S2 is true iff S1 is true **or** S2 is true
  - S1 ⇒ S2 is true iff S1 is false **or** S2 is true (is false iff S1 is true **and** S2 is false)
  - S1 ⇔ S2 is true iff S1⇒S2 is true **and** S2⇒S1 is true
- Simple recursive process evaluates an arbitrary sentence, e.g.,
  $\neg P_{1,2} \land (P_{2,2} \lor P_{3,1})$ = **true ∧ (false ∨ true) = true ∧ true = true**

TRUTH TABLE

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*

SENTENCES IN WUMPUS WORLD

- 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]



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

- Pits cause breeze in adjacent squares

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

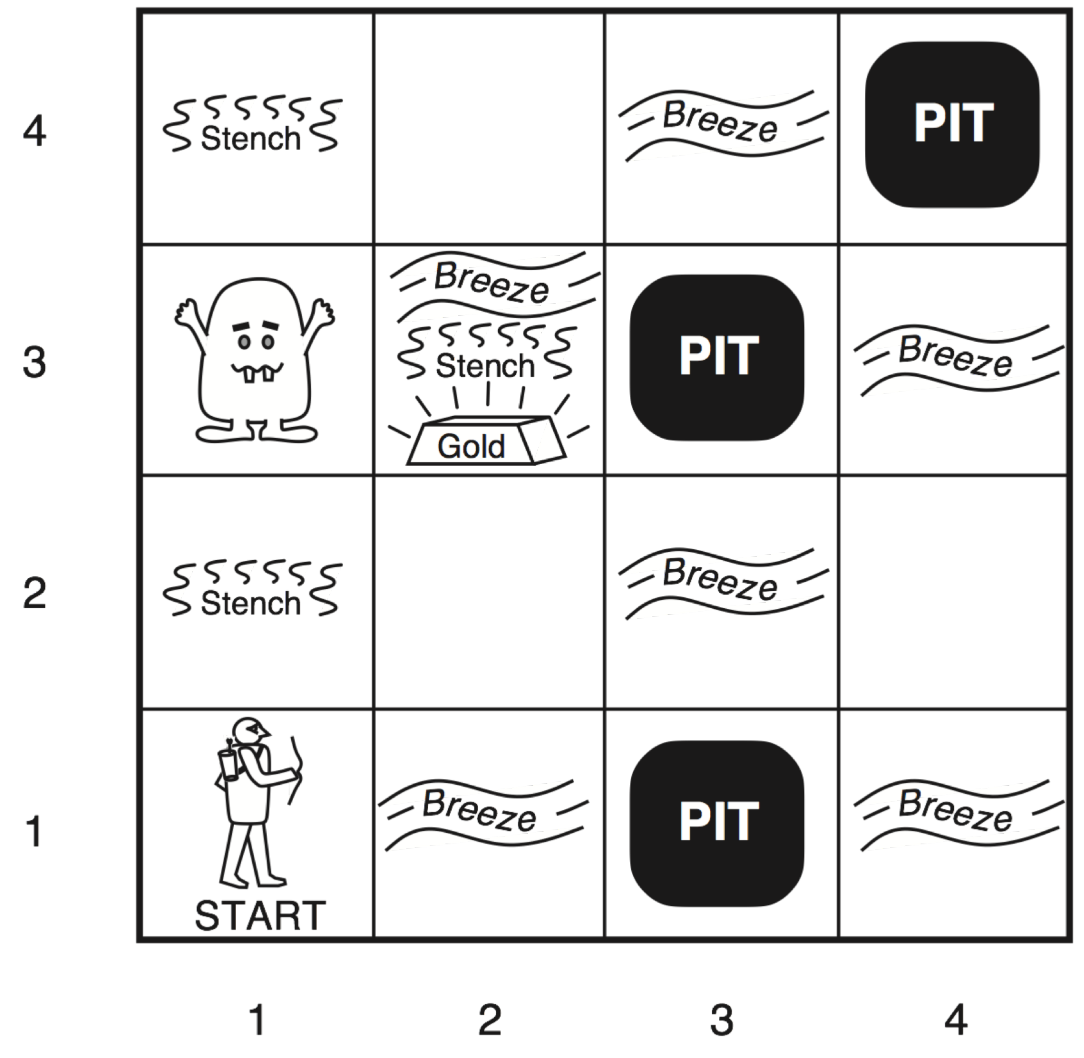
$$B_{2,1} \Rightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$$

SENTENCES IN WUMPUS WORLD

- 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]

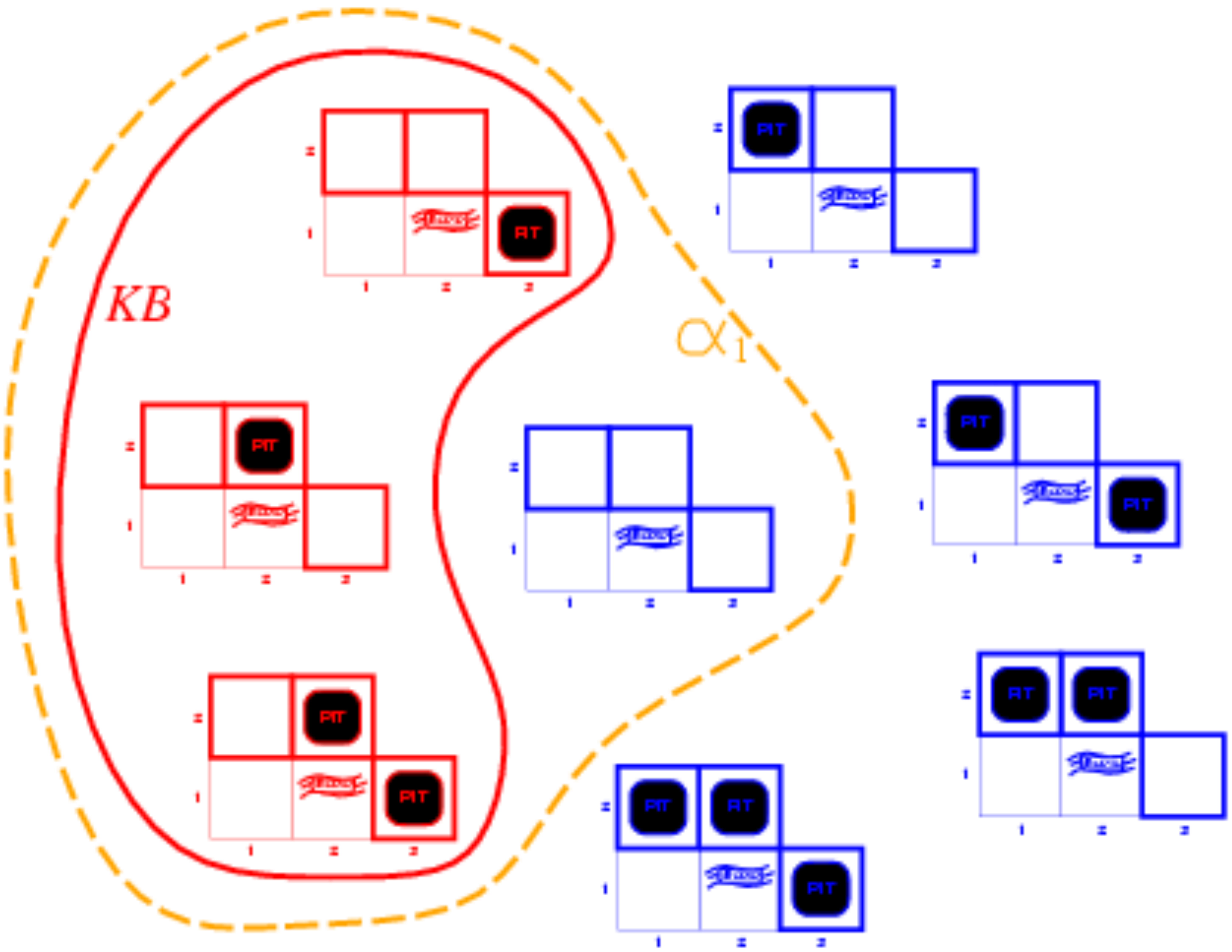
$$\neg P_{1,1} \quad \neg B_{1,1} \quad B_{2,1}$$

- Pits cause breeze in adjacent squares



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

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


KB
α_1

INFERENCE METHOD: ENUMERATION

```
function TT-ENTAILS?(KB, α) returns true or false
   inputs: KB, the knowledge base, a sentence in propositional logic
           α, the query, a sentence in propositional logic

   symbols ← a list of the proposition symbols in KB and α
   return TT-CHECK-ALL(KB, α, symbols, [ ])
```

```
function TT-CHECK-ALL(KB, α, symbols, model) returns true or 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))
```

Just check every single possible assignment of variables in both KB and α

TRUTH TABLES FOR INFERENCE

$B_{1,1}$	$B_{2,1}$	$P_{1,1}$	$P_{1,2}$	$P_{2,1}$	$P_{2,2}$	$P_{3,1}$	KB	α_1
false	*false*	*false*	*false*	*false*	*false*	*false*	*false*	*true*
false	*false*	*false*	*false*	*false*	*false*	*true*	*false*	*true*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
false	*true*	*false*	*false*	*false*	*false*	*false*	*false*	*true*
false	*true*	*false*	*false*	*false*	*false*	*true*	<u>*true*</u>	<u>*true*</u>
false	*true*	*false*	*false*	*false*	*true*	*false*	<u>*true*</u>	<u>*true*</u>
false	*true*	*false*	*false*	*false*	*true*	*true*	<u>*true*</u>	<u>*true*</u>
false	*true*	*false*	*false*	*true*	*false*	*false*	*false*	*true*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
true	*true*	*true*	*true*	*true*	*true*	*true*	*false*	*false*

RECAP: LOGIC

- **Logics** are formal languages for representing information such that conclusions can be drawn
- **Sentence** represents an assertion about the world that can be true or false
- **Knowledge base** is a set of sentences in a representation language
- **Syntax** defines legal sentences in the language (i.e., how to build up complex sentences from atomic sentences)
- **Semantics** define the "meaning" of sentences (i.e., rules for determining truth of a sentence over "possible worlds")
- **Model** is a mathematical abstraction of a possible world, which fixes the truth/falsehood of every relevant sentence

RECAP: ENTAILMENT AND INFERENCE

- **Entailment**: the idea that a sentence follows logically from another sentence
- **Inference**: procedure/algorithm to derive conclusions
  - Like finding needle in a haystack
  - Haystack: set of all consequences of KB
  - Needle: α
- Desirable properties of inference algorithms
  - **Soundness**: Only entailed sentences are derived
  - **Completeness**: Any entailed sentence can be derived

HOW TO DETERMINE ENTAILMENT?

- Methods can be divided into (roughly) two kinds:
  - **Theorem proving: Application of inference rules**
    - Legitimate (sound) generation of new sentences from old
    - Proof = a sequence of inference rule applications
      Can use inference rules as operators in a standard search algorithm
    - Typically require transformation of sentences into a normal form
  - **Model checking**
    - Truth table enumeration (always exponential in n)
    - Improved backtracking, e.g., Davis-Putnam-Logemann-Loveland (DPLL)
    - Heuristic search in model space (sound but incomplete), e.g., min-conflicts-like hill-climbing algorithms

LOGICAL EQUIVALENCE

- Two sentences are **logically equivalent** iff they are true in the same set of models: $\alpha \equiv \beta$ iff $\alpha \vDash \beta$ and $\beta \vDash \alpha$

$$
\begin{array}{rcll}
(\alpha \wedge \beta) & \equiv & (\beta \wedge \alpha) & \text{commutativity of } \wedge \\
(\alpha \vee \beta) & \equiv & (\beta \vee \alpha) & \text{commutativity of } \vee \\
((\alpha \wedge \beta) \wedge \gamma) & \equiv & (\alpha \wedge (\beta \wedge \gamma)) & \text{associativity of } \wedge \\
((\alpha \vee \beta) \vee \gamma) & \equiv & (\alpha \vee (\beta \vee \gamma)) & \text{associativity of } \vee \\
\neg(\neg\alpha) & \equiv & \alpha & \text{double-negation elimination} \\
(\alpha \Rightarrow \beta) & \equiv & (\neg\beta \Rightarrow \neg\alpha) & \text{contraposition} \\
(\alpha \Rightarrow \beta) & \equiv & (\neg\alpha \vee \beta) & \text{implication elimination} \\
(\alpha \Leftrightarrow \beta) & \equiv & ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) & \text{biconditional elimination} \\
\neg(\alpha \wedge \beta) & \equiv & (\neg\alpha \vee \neg\beta) & \text{De Morgan} \\
\neg(\alpha \vee \beta) & \equiv & (\neg\alpha \wedge \neg\beta) & \text{De Morgan} \\
(\alpha \wedge (\beta \vee \gamma)) & \equiv & ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) & \text{distributivity of } \wedge \text{ over } \vee \\
(\alpha \vee (\beta \wedge \gamma)) & \equiv & ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) & \text{distributivity of } \vee \text{ over } \wedge
\end{array}
$$