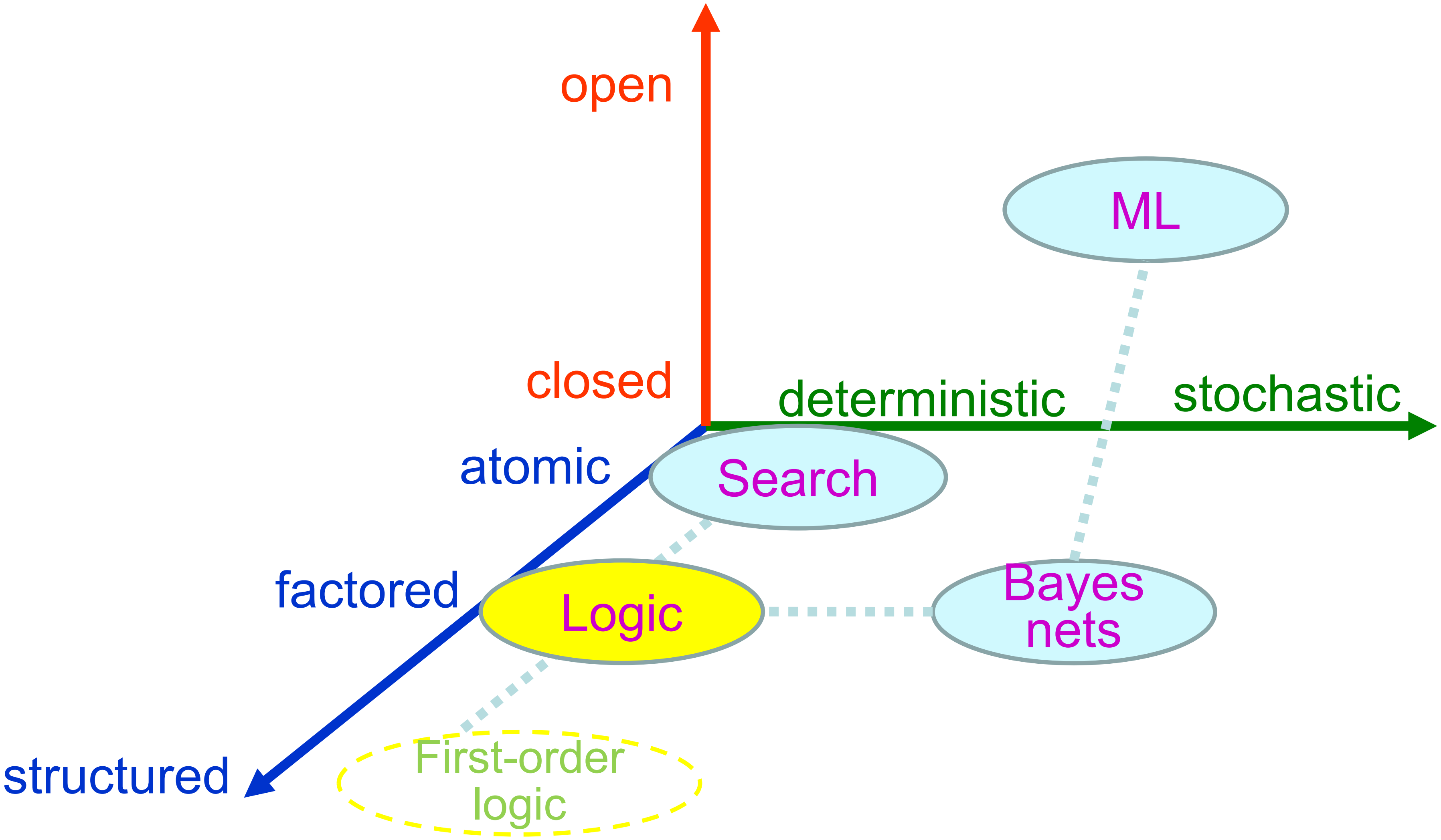
Artificial Intelligence

Section 7： **Logic Agent**

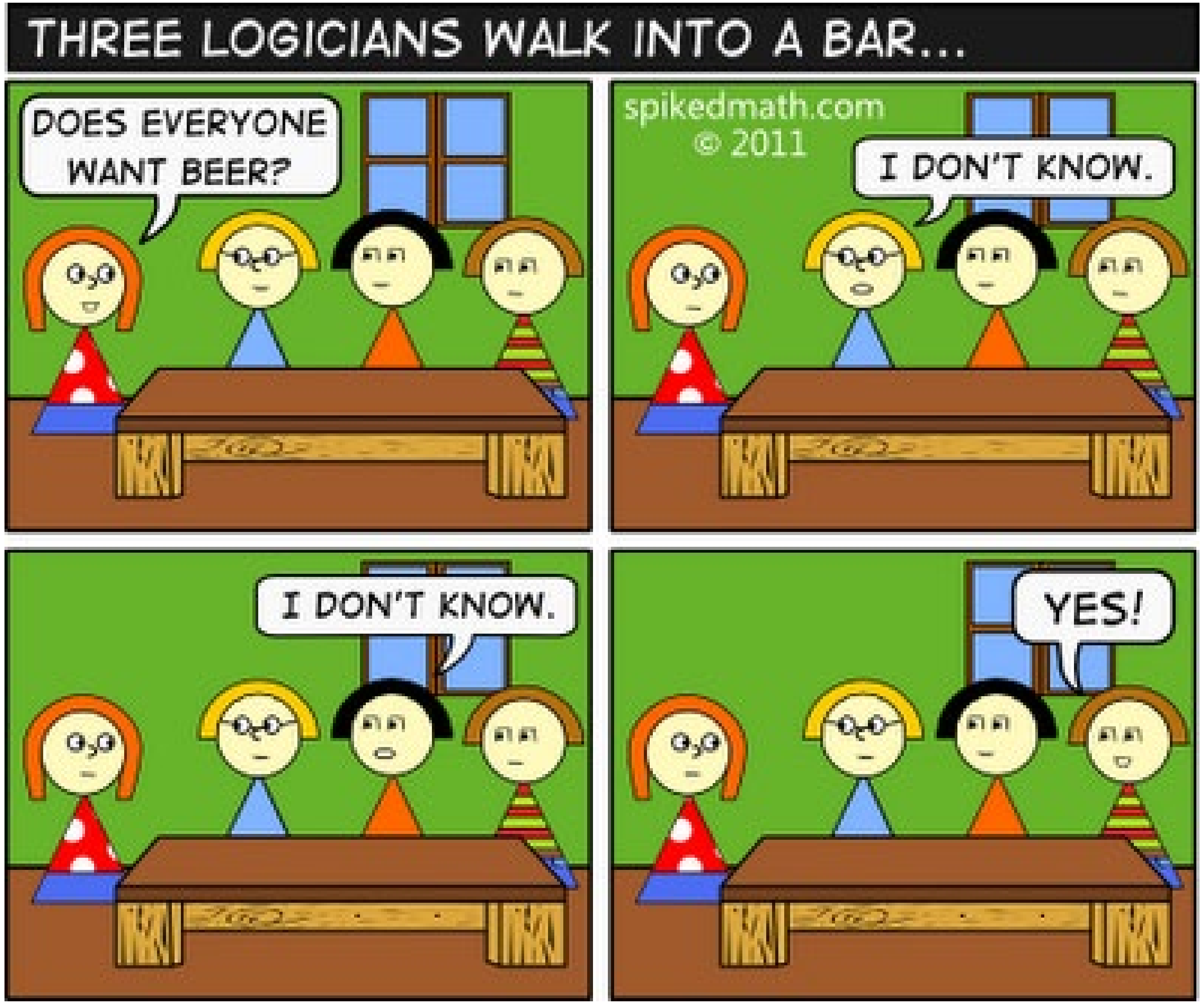
Outline of the course



Outline

- 7.1 基于知识的智能体
- 7.2 Wumpus world
- 7.3 知识的逻辑表示和推理
- 7.4 命题逻辑：一种简单的逻辑
- 7.5 命题逻辑定理证明
  - 模型检验、逻辑规则、归结原理

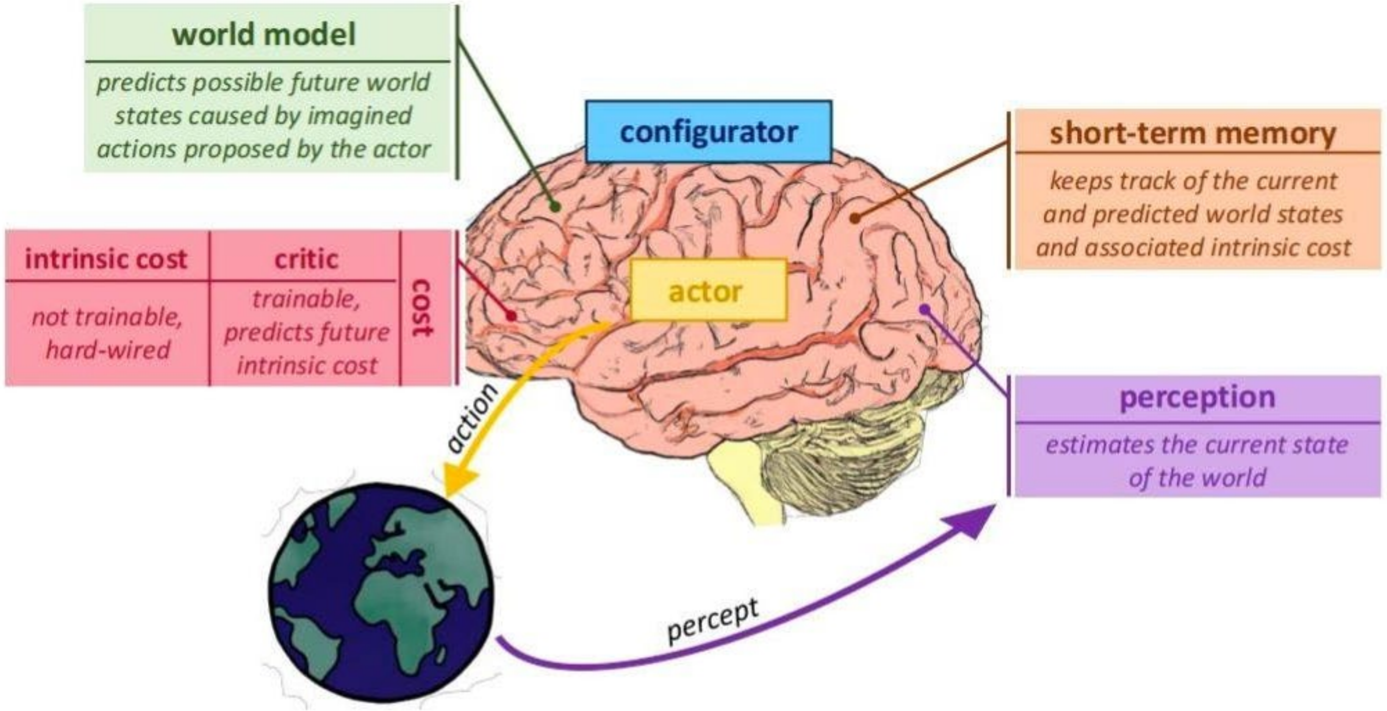
基于知识的智能体



THREE LOGICIANS WALK INTO A BAR...
DOES EVERYONE WANT BEER?
spikedmath.com © 2011
I DON'T KNOW.
I DON'T KNOW.
YES!

基于知识的智能体

世界模型(World Models)



https://haokan.baidu.com/v?pd=wisenatural&vid=4492493327783748355 03:30

基于知识的智能体

- Agents通过感知、学习和语言来获取知识
  - 行动的影响（“转移模型”）
  - 世界如何影响传感器（“传感器模型”）
  - 世界当前状态
- 可以追踪一个部分可见的世界
- 能够制定实现目标的计划
- …

Knowledge base
Inference engine

Domain-specific facts

基于知识的智能体

- 知识库***KB*** = set of sentences in a formal language

- 构建智能体的声明性方法：
  - Tell：Agent告诉知识库需要知道的一切
  - Ask：Agent询问知识库要执行什么动作？（回答应该是基于知识库的）

- 这两个过程都可能涉及推理，即从原有语句中推导出新的语句。还可以使得Agent具有学习能力

A knowledge-based agent

Function KB-AGENT(percept) **returns** an action

persistent: KB, a knowledge base

t, an integer, initially 0

TELL(KB, MAKE-PERCEPT-SENTENCE(percept, t))

action ← ASK(KB, MAKE-ACTION-QUERY(t))

TELL(KB, MAKE-ACTION-SENTENCE(action, t))

t←t+1

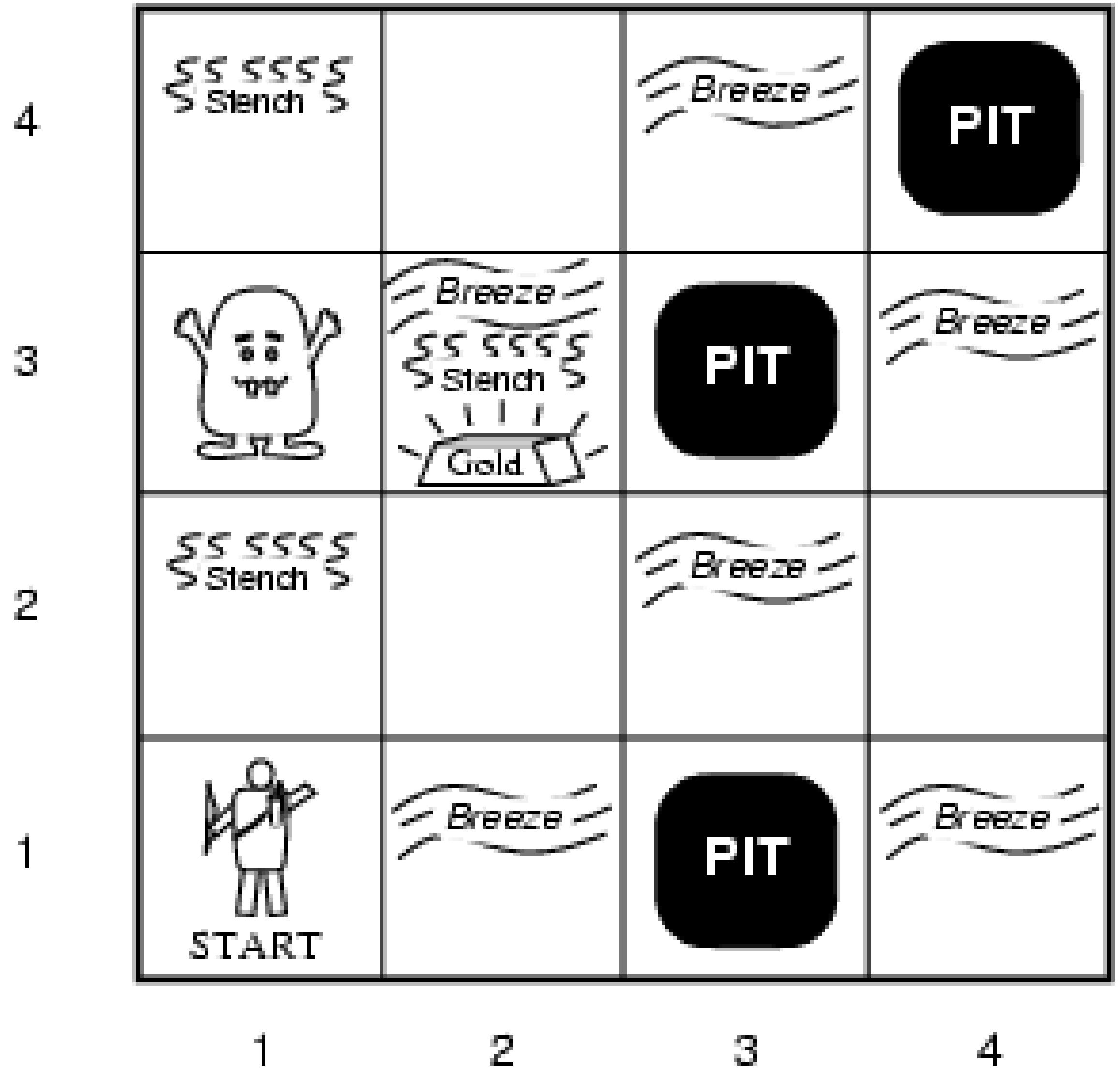
return action

- 智能体应该能够:
  - 表示状态、行为等等
  - 融入新的感知
  - 更新对周围世界的内部表示
  - 推断周围世界的隐藏属性
  - 推断出正确的行为

Outline

- 7.1 基于知识的智能体
- 7.2 Wumpus world
- 7.3 知识的逻辑表示和推理
- 7.4 命题逻辑：一种简单的逻辑
- 7.5 命题逻辑定理证明
    - 模型检验、逻辑规则、归结原理

Wumpus World



Wumpus世界环境：

由多个房间组成并连接起来的**山洞。**

在洞穴某处隐藏着一只**Wumpus**，会吃掉进入它房间的任何人。Wumpus周围有臭气**Stench**

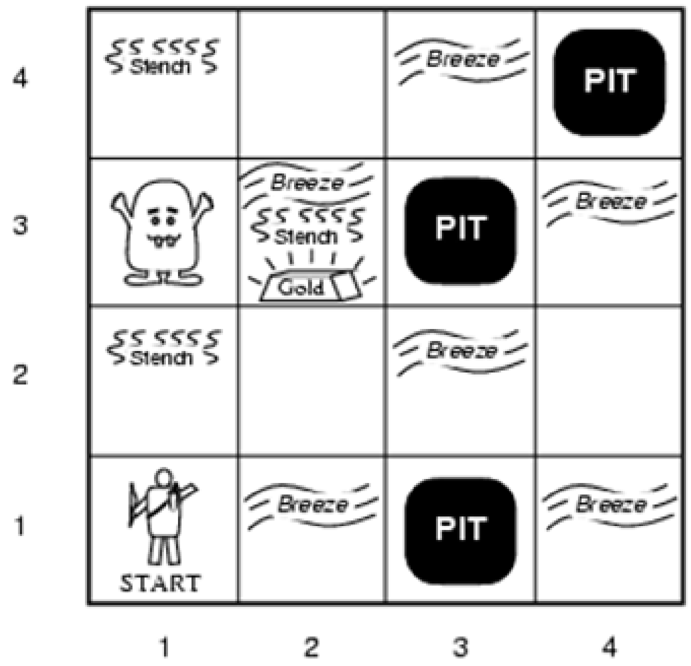
某些房间是**无底洞PIT**，任何人漫游到这些房间都会被无底洞吞噬。无底洞周围有微风**Breeze**

生活在该环境的唯一希望是存在**发现一堆金子**的可能性。

Agent：可以移动到相邻房间，可以射杀Wumpus，但是只有一支箭。

Wumpus World PEAS description

- Performance measure
  - 黄金 +1000, 死亡 -1000
  - 每步 -1, 使用箭 -10
- Environment
  - 与怪兽相邻的方块是臭的
  - 无底黑洞旁边的广场有微风
  - 亮光和金子在同一个方块中
  - 如果你面对怪兽射击，会杀死它
  - 射击会消耗唯一的箭
  - 如果和金子在同一个方块中，抓取可以获得金子
  - 如果和金子在同一个方块中，放手可以扔掉金子
- Sensors: 臭味， 风， 发光， 撞击， 尖叫
- Actuators: 向左转， 向右转， 向前， 抓取， 放手， 射击



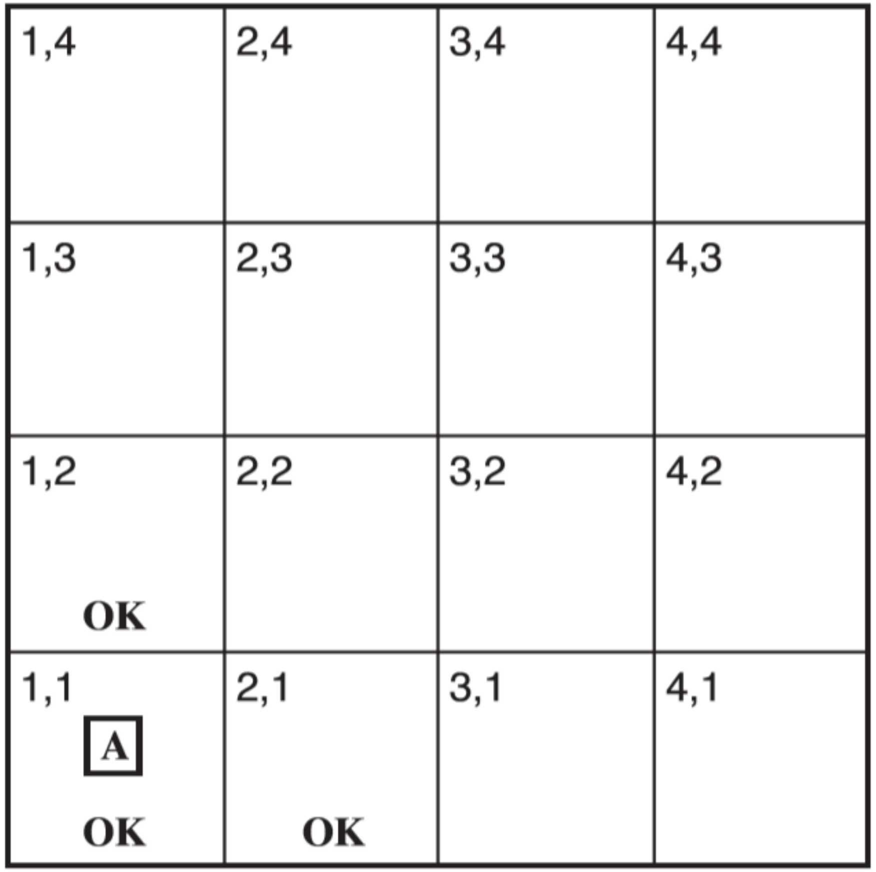
a percept
[Stench, Breeze, Glitter, Crash, Yell]

Wumpus world characterization

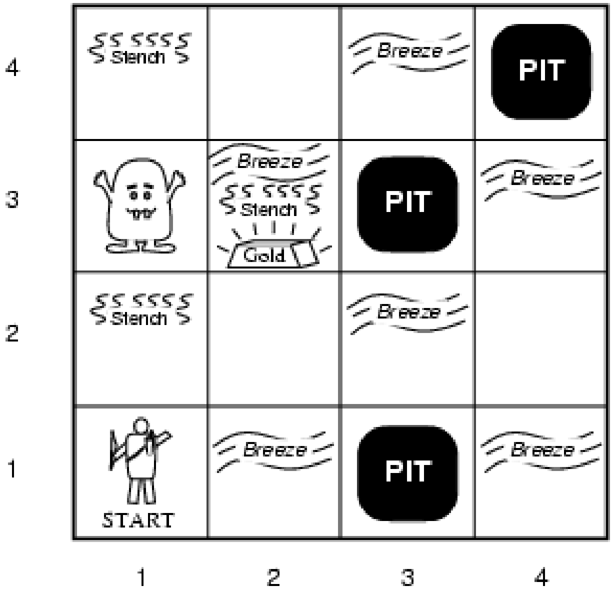
- Fully Observable No – only local perception
- Deterministic Yes – outcomes exactly specified
- Episodic No – sequential at the level of actions
- Static Yes – Wumpus and Pits do not move
- Discrete Yes
- Single-agent Yes – Wumpus is essentially a natural feature

Exploring a wumpus world

首先，定义感知(状态描述)： [Stench, Breeze, Glitter, Crash, Yell]



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

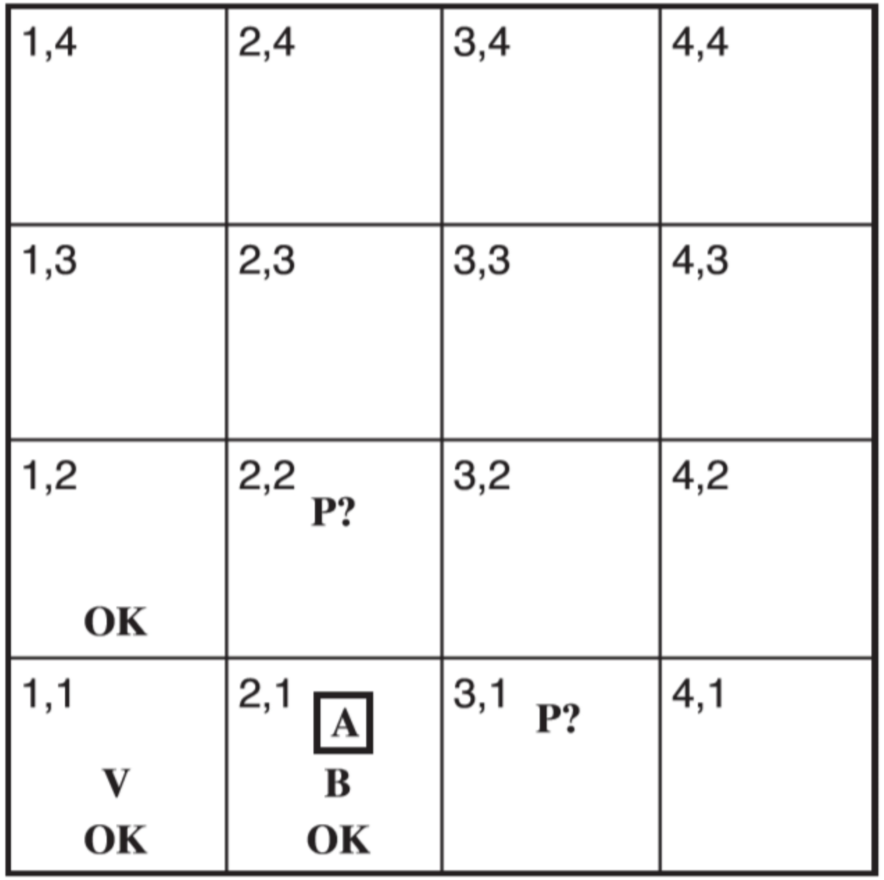


The first percept is [None,None,None,None,None]

conclude [1,2] and [2,1] are OK

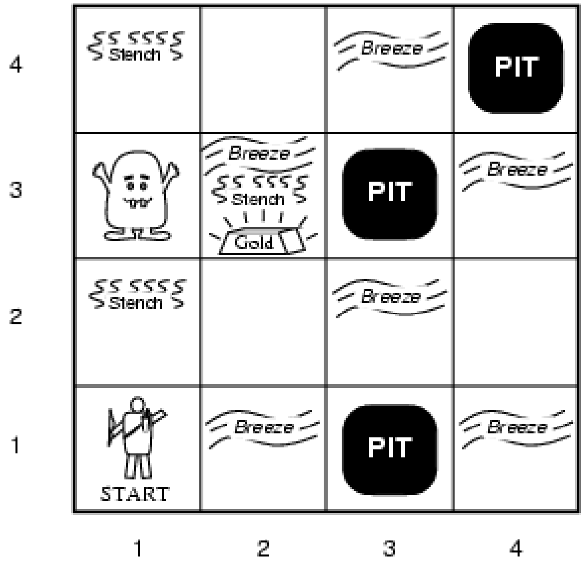
Action：[2,1]

Exploring a wumpus world



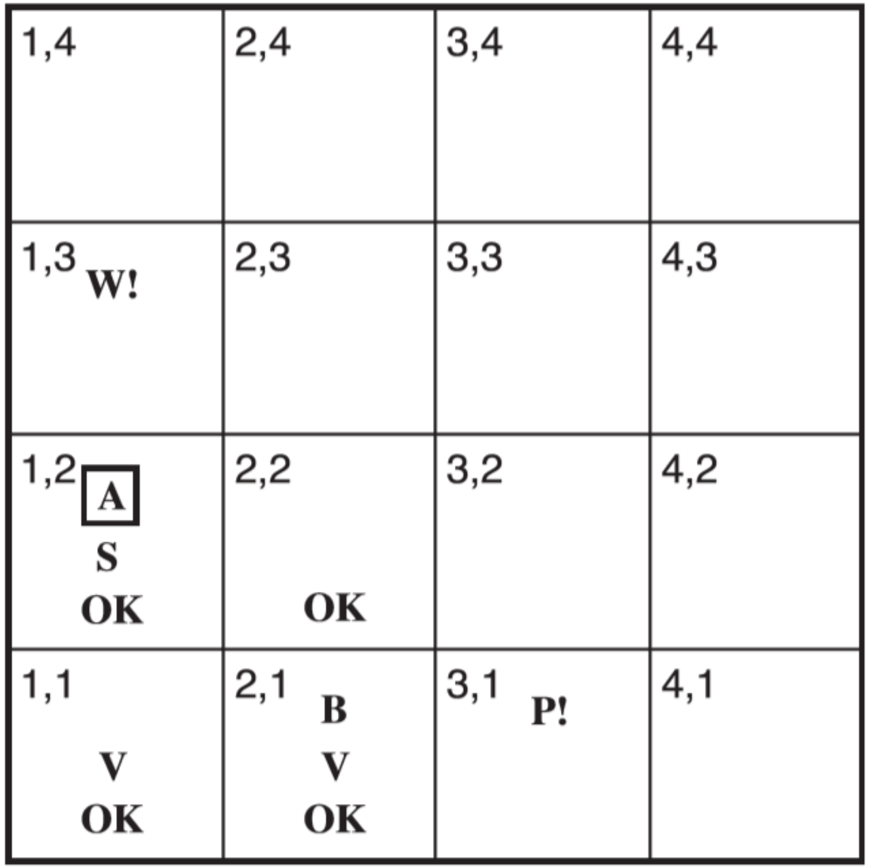
percept [None,Breeze,None,None,None] in [2,1]

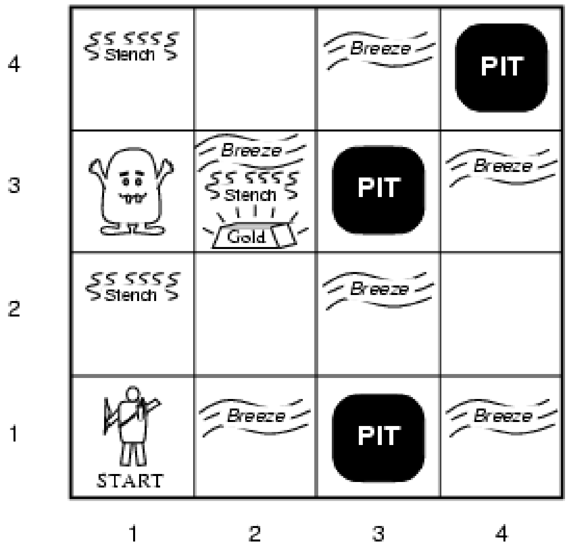
Action： [1,2]



there must be a pit in [2,2] or [3,1] or both

Exploring a wumpus world





wumpus is in [1,3].

the lack of a breeze in [1,2] implies no pit in [2,2]

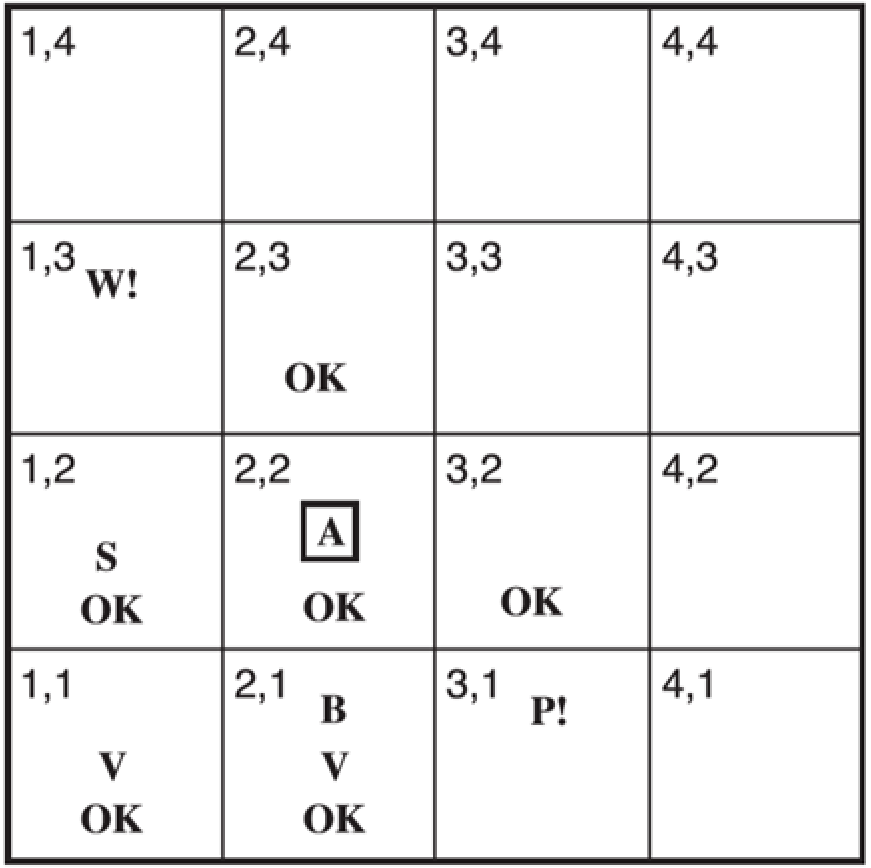
neither a pit nor a wumpus in [2,2], so it is OK

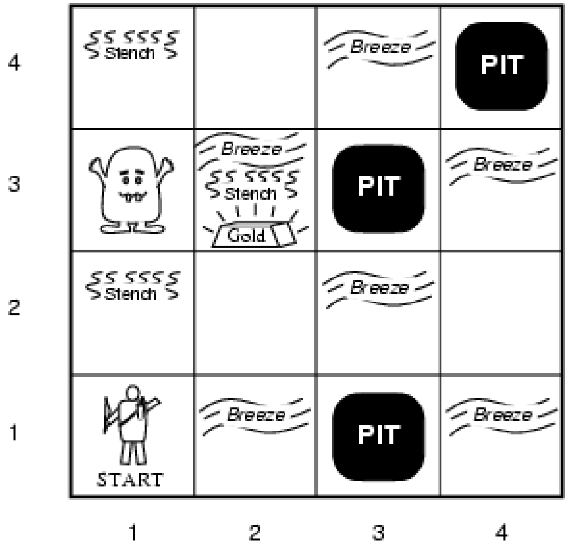
pit in [3,1]

percept [Stench,None,None,None,None] in [1,2]

Action： [2,2]

Exploring a wumpus world





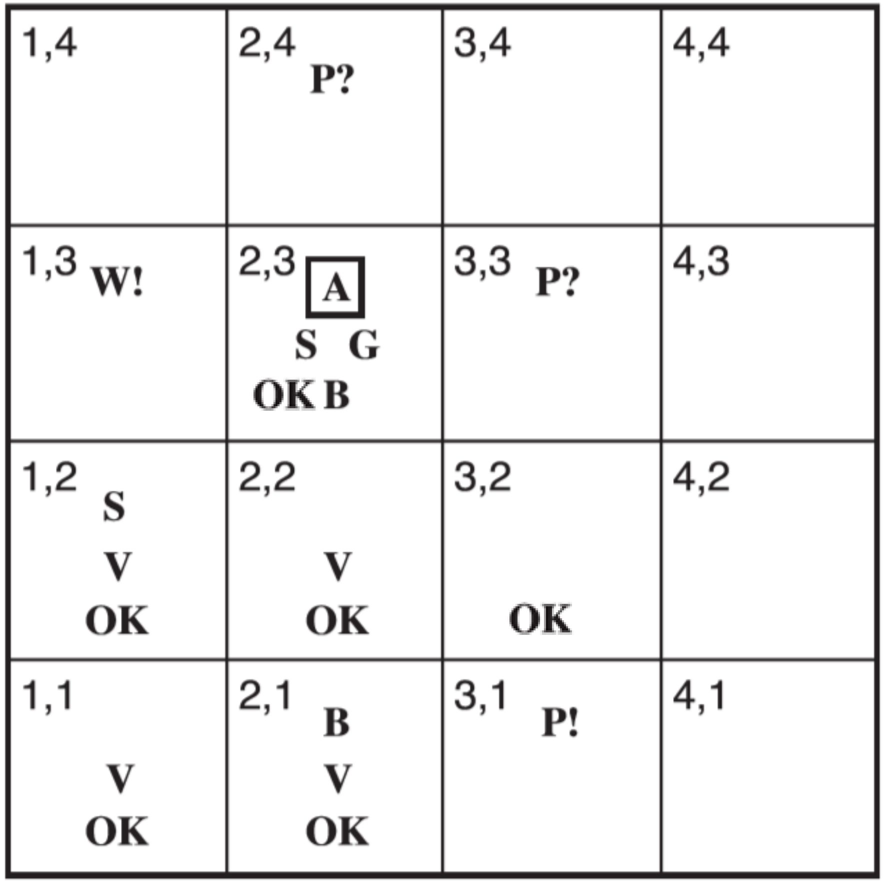
[2,3] is OK

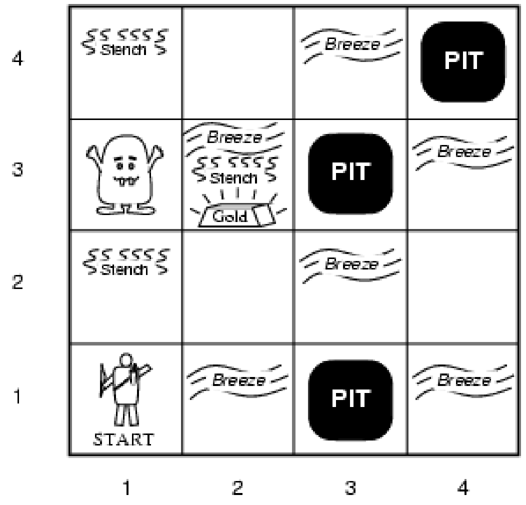
[3,2] is OK

percept [None,None,None,None,None] in [2,2]

Action： [2,3]

Exploring a wumpus world





percept [Stench,Breeze,Glitter,None,None] In [2,3]

the agent detects a glitter, so it should grab the gold and then return home.

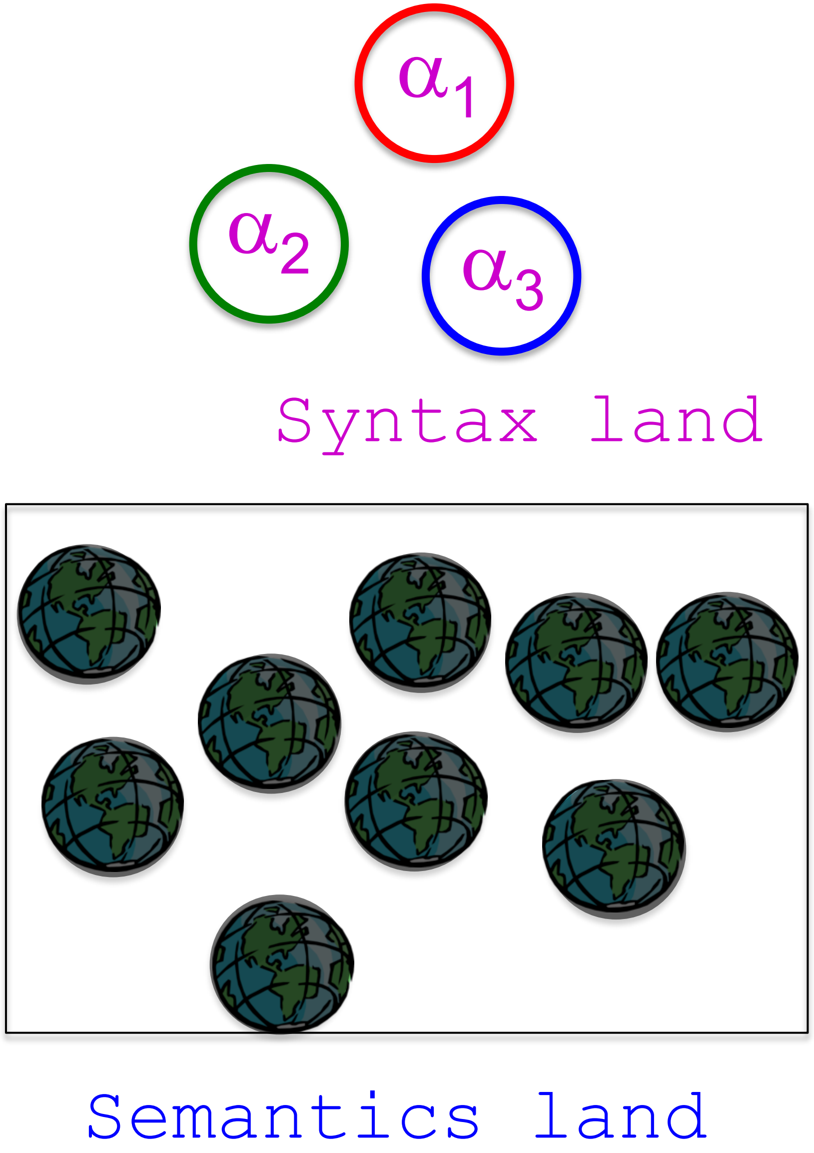
Outline

- 7.1 基于知识的智能体
- 7.2 Wumpus world
- 7.3 知识的逻辑表示和推理
- 7.4 命题逻辑：一种简单的逻辑
- 7.5 命题逻辑定理证明
  - 模型检验、逻辑规则、归结原理

什么是逻辑?

- **逻辑**: 处理事实的形式化系统，目的得到正确结论
  - 用来区分真和假的工具 – *Averroes (12世纪)*
- **语法**: 构造知识库有效句子的规则
  - 例如, x + 2 ≥ y 是有效的算术语句, ≥ x2y + 就不是
- **语义**: 句子的“含义”，逻辑语句和真实世界之间关系
  - 具体而言,语义决定了句子的真假
  - 例如

Sentence: x > y为真 World1: x = 5; y = 2 World2: x = 2; y = 3

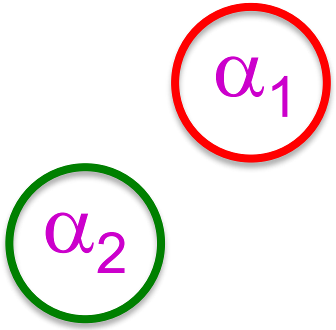


逻辑推理

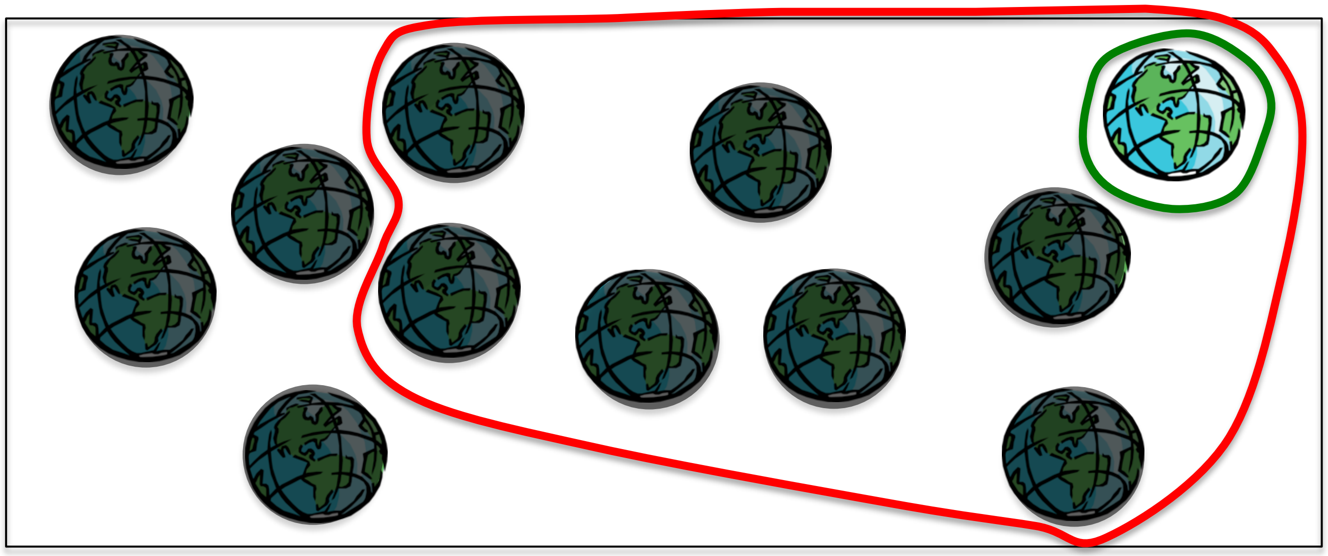
- **蕴含：** 意味着一样东西是从另一样东西中派生出来的
- KB蕴含句子α 当且仅当α 在所有KB为真的世界里为真

$$KB \models \alpha$$

- 例如, $\alpha_2 \models \alpha_1$ (Say α_2 is $\neg Q \wedge R \wedge S \wedge W$; α_1 is $\neg Q$)



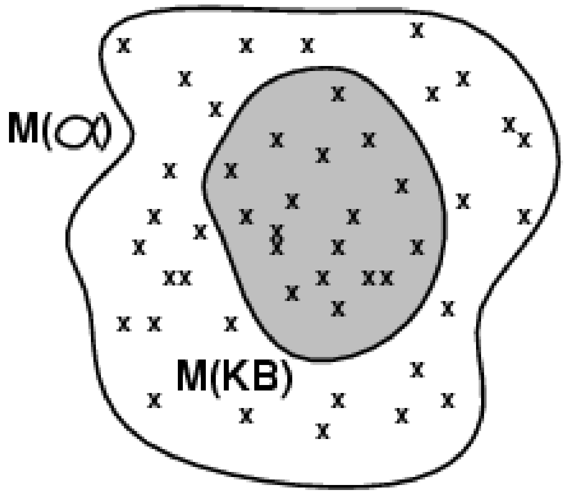
蕴含是基于语义的句子之间的关系



模型

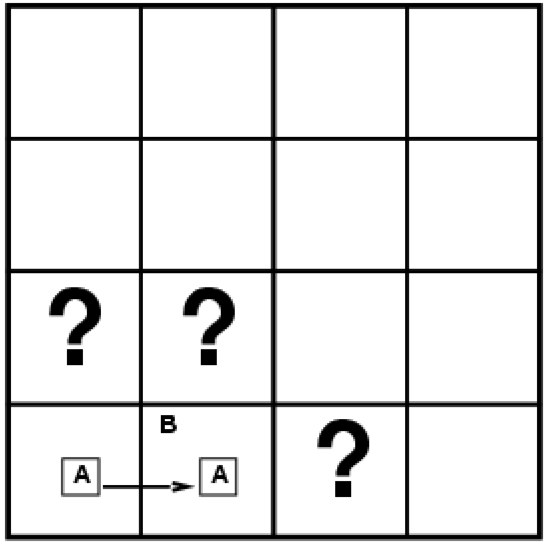
- 逻辑学家通常根据模型来思考，这些模型是可以评估真假的形式化的结构世界
- 如果α 在M中为真，我们说M是句子α 的模型

- $M(\alpha)$ 是所有α的模型的集合
- 那么 KB $\models$ α iff $M(KB) \subseteq M(\alpha)$

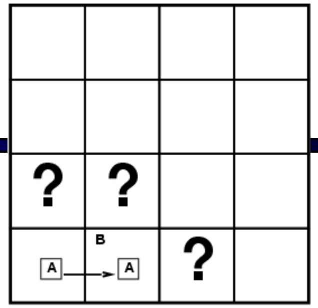


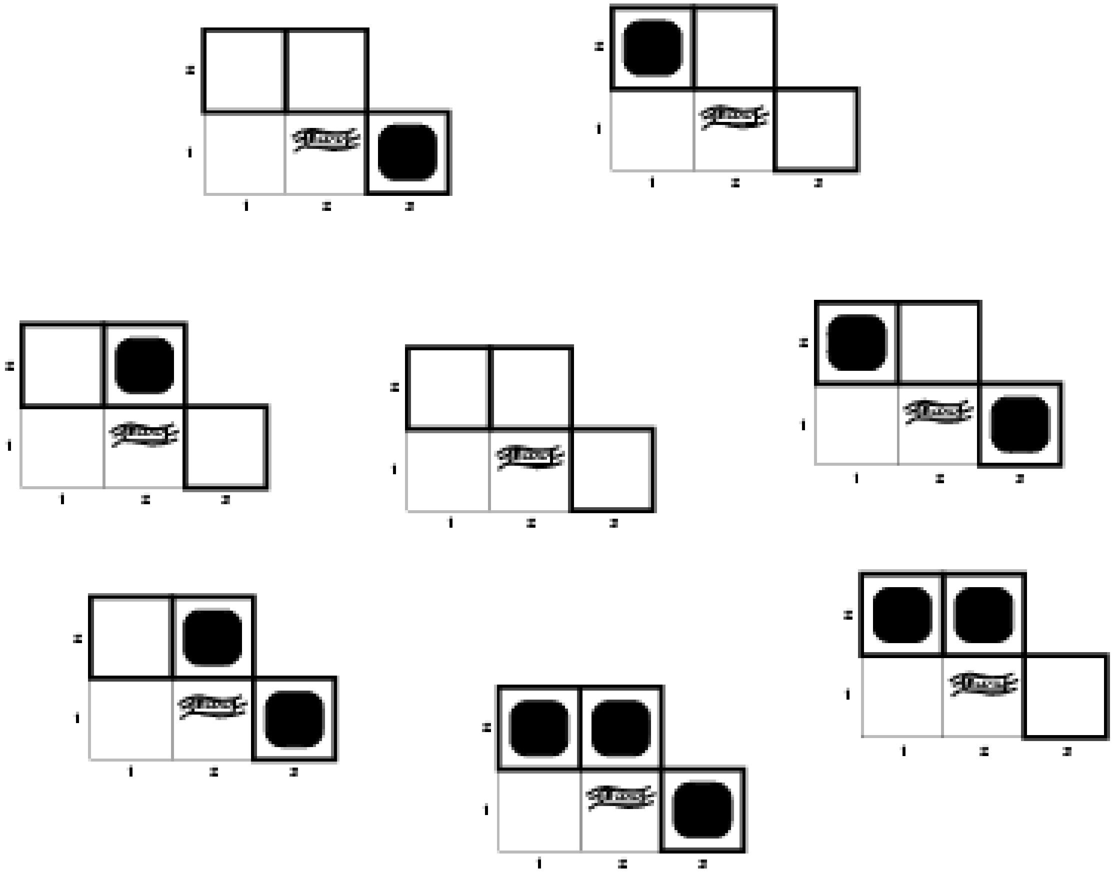
在怪兽世界中的蕴含规则

- 检测到[1,1]中没有东西, 向右移动, [2,1]中有微风
- 问题：是否有无底洞?
- 知识库KB： 8 个可能的情况（布尔选择）

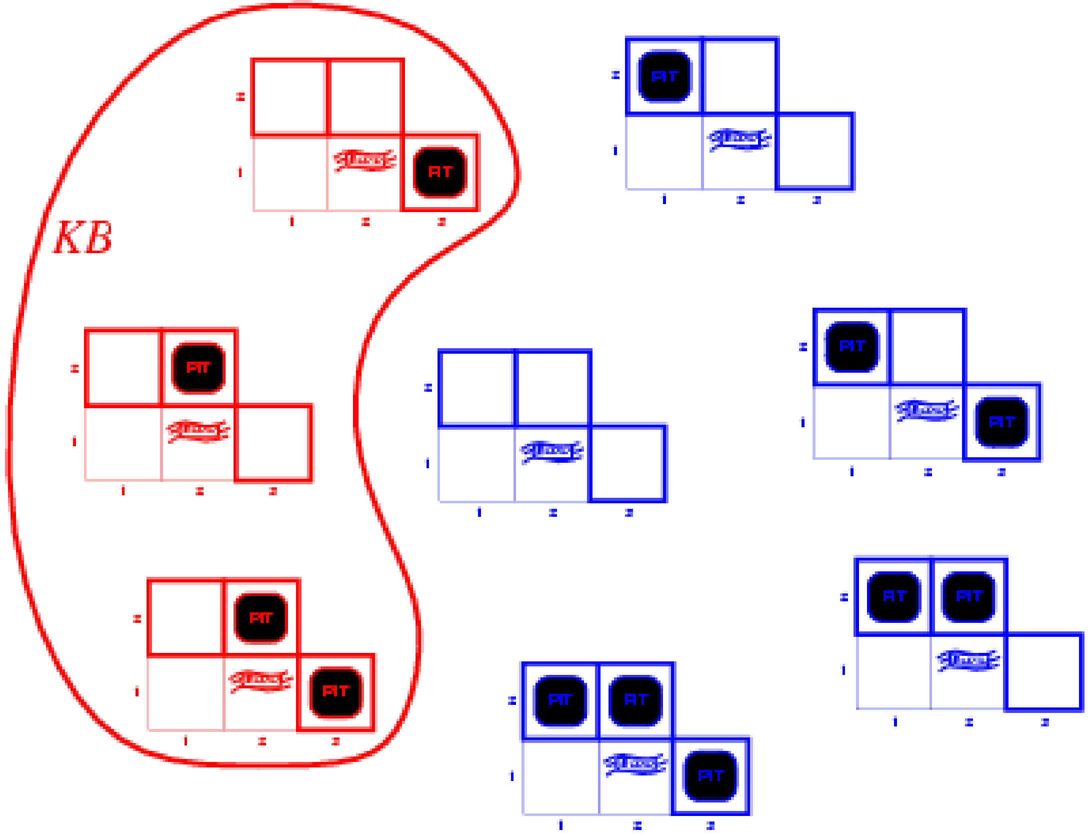


Wumpus models





Wumpus models



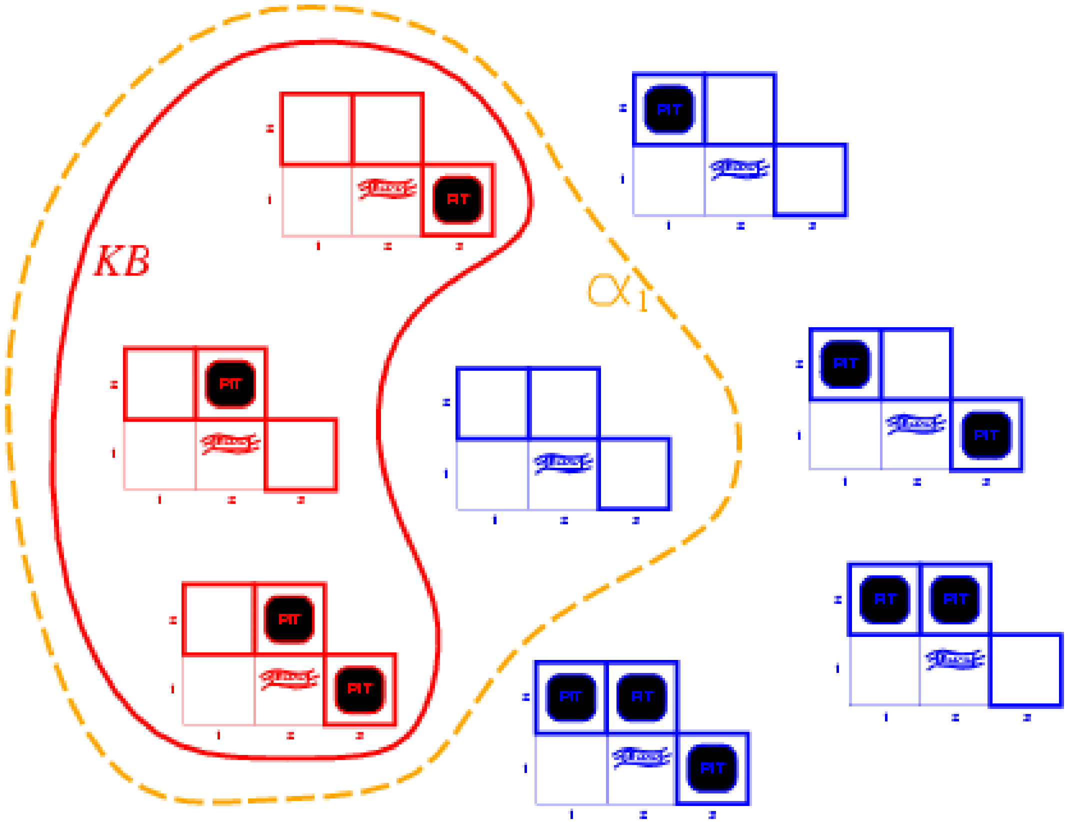
Observations：

[1,1]中没有东西

[2,1]中有微风

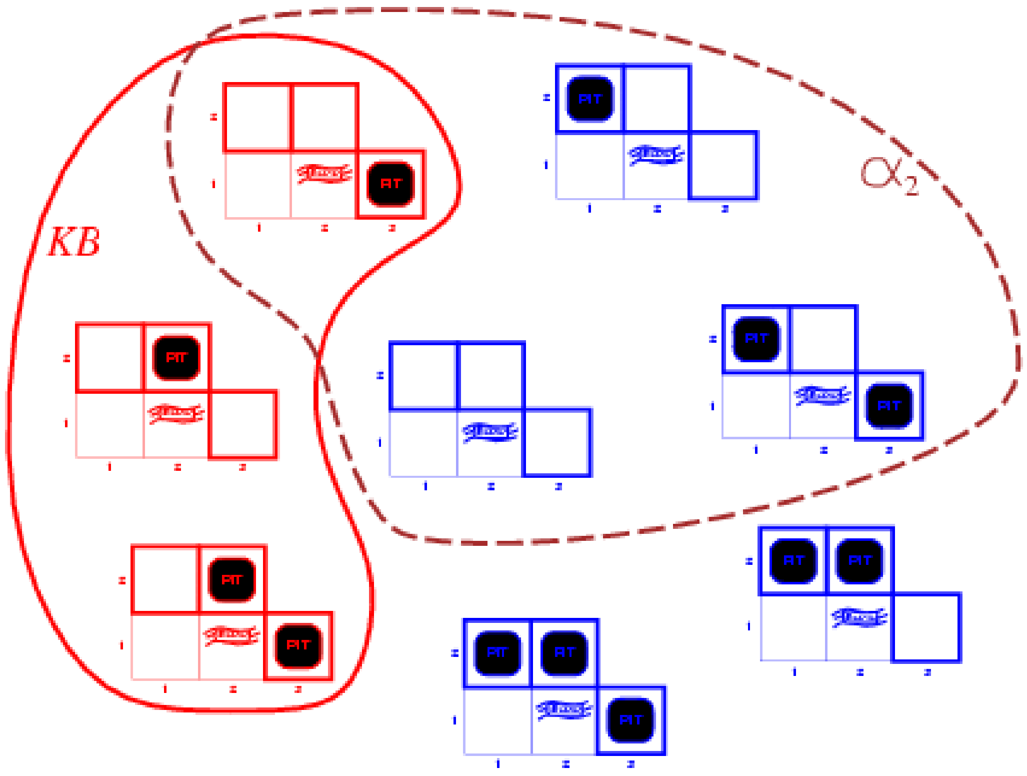
- *KB* = wumpus-world rules + observations

Wumpus models



- *KB* = wumpus-world rules + observations
- α_1 = "[1,2] is safe", 模型验证证明 ***KB*** $\models \alpha_1$

Wumpus models



- *KB* = wumpus-world rules + observations
- α_2 = “[2,2] is safe”, *KB* $\models \alpha_2$是否成立?
- ***KB*** $\not\models$ $\boldsymbol{\alpha_2}$： Agent无法得出[2,2]中没有无底洞

Outline

- 7.1 基于知识的智能体
- 7.2 Wumpus world
- 7.3 知识的逻辑表示和推理
- 7.4 命题逻辑：一种简单的逻辑
- 7.5 命题逻辑定理证明
  - 模型检验、逻辑规则、归结原理

命题

- **命题:** 能判断真假的陈述句。是具有唯一真值的陈述句或判断结果唯一的陈述句
- **命题的真值:** 判断的结果，真值的取值真与假二者取一
- **真命题:** 真值为真的命题
- **假命题:** 真值为假的命题
  - 注意: 感叹句、祈使句、疑问句都不是命题
  - 陈述句中的悖论以及判断结果不唯一确定的也不是命题

例 下列句子中那些是命题？

(1) 9 是无理数.

(2) $2 + 5 = 8$.

(3) $x + 5 > 3$.

(4) 你有铅笔吗？

(5) 这只兔子跑得真快呀！

(6) 请不要讲话！

(7) 我正在说谎话.

命题逻辑:语法

原子命题　　复合命题

$$
\begin{aligned}
\mathit{Sentence} &\rightarrow \mathit{AtomicSentence} \mid \mathit{ComplexSentence} \\
\mathit{AtomicSentence} &\rightarrow \mathit{True} \mid \mathit{False} \mid P \mid Q \mid R \mid \ldots \\
\mathit{ComplexSentence} &\rightarrow (\ \mathit{Sentence}\ ) \mid [\ \mathit{Sentence}\ ] \\
&\mid\ \neg\ \mathit{Sentence} \quad \text{取非} \\
&\mid\ \mathit{Sentence} \land \mathit{Sentence} \quad \text{合取} \\
&\mid\ \mathit{Sentence} \lor \mathit{Sentence} \quad \text{析取} \\
&\mid\ \mathit{Sentence} \Rightarrow \mathit{Sentence} \quad \text{蕴含} \\
&\mid\ \mathit{Sentence} \Leftrightarrow \mathit{Sentence} \quad \text{双向蕴含}
\end{aligned}
$$

OPERATOR PRECEDENCE : $\neg, \land, \lor, \Rightarrow, \Leftrightarrow$

Figure 7.7 A BNF (Backus–Naur Form) grammar of sentences in propositional logic, along with operator precedences, from highest to lowest.

$\neg, \land, \lor, \Rightarrow, \Leftrightarrow$ 逻辑连接词优先级从高到低

命题逻辑：语义

- **语义**定义了用于判定特定模型中语句真值的规则。
-
- **原子命题**容易计算:
  - 每个模型中：*True* is true and *False* is false
  - 每个命题的真值必须在模型中直接指定

E.g.　　P：4=4 ；　P=True

in wumpus-world

Q：[1，1]is safe;　Q=True

命题逻辑：语义

复合命题 有5条规则：(P and Q 是模型 *m*中的任意子句)

$\neg P$	is true	iff	P is false in *m*
$P \wedge Q$	is true	iff	both P is **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	iff	**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*

语义定义了用于判定模型（可能世界）中语句真值的规则

Truth tables for connectives

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*

Figure 7.8 Truth tables for the five logical connectives. To use the table to compute, for example, the value of $P \vee Q$ when P is true and Q is false, first look on the left for the row where P is *true* and Q is *false* (the third row). Then look in that row under the $P \vee Q$ column to see the result: *true*.

⇒ 的真值可能不太符合直觉： P implies Q 或者 if P then Q

例如, 5是偶数 ⇒ Sam很聪明 is true or false?

⇒ 为假的唯一条件：P为真而Q为假

Outline

- 7.1 基于知识的智能体
- 7.2 Wumpus world
- 7.3 知识的逻辑表示和推理
- 7.4 命题逻辑：一种简单的逻辑
- 7.5 命题逻辑定理证明
  - 模型检验、逻辑规则、归结原理

推理: 判断逻辑蕴涵

- Method 1: ***model-checking*（模型检验）**
  - 一种简单的枚举推理
  - 枚举所有的模型(可能世界), 并验证语句在所有模型中为真
- Method 2: Application of inference rules **（逻辑规则）**
- Method 3: ***theorem-proving*（定理证明）**

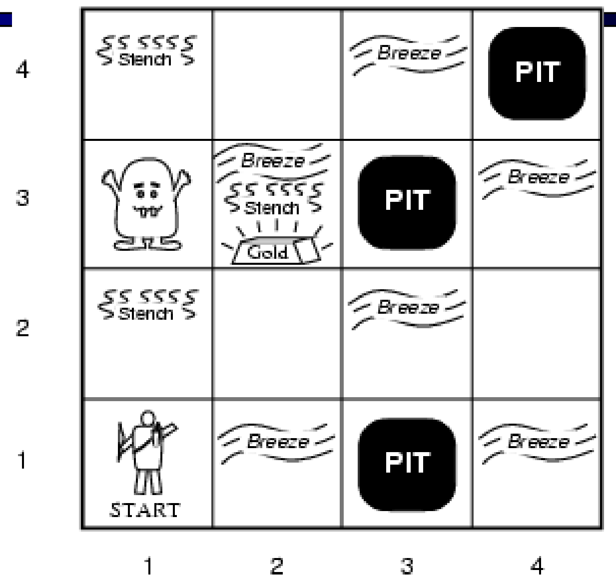
Wumpus world sentences

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].

R_1: $\neg P_{1,1}$

R_2: $\neg B_{1,1}$

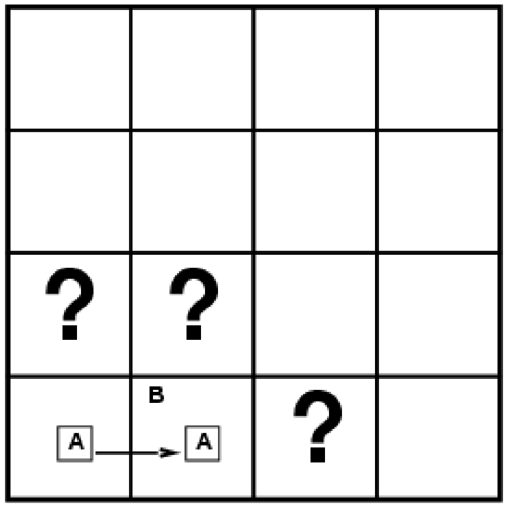
R_3: $B_{2,1}$



- "Pits cause breezes in adjacent squares"

R_4: $B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$

R_5: $B_{2,1} \Leftrightarrow (P_{1,1} \lor P_{2,2} \lor P_{3,1})$



此时的知识库KB由以上五条语句组成

KB=$R_1 \land R_2 \land R_3 \land R_4 \land R_5$

Truth tables for inference

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

Figure 7.9 A truth table constructed for the knowledge base given in the text. KB is true if R_1 through R_5 are true, which occurs in just 3 of the 128 rows (the ones underlined in the right-hand column). In all 3 rows, $P_{1,2}$ is false, so there is no pit in [1,2]. On the other hand, there might (or might not) be a pit in [2,2].

$KB \models \neg P_{1,2}$

R_6: $\neg P_{1,2}$

With seven symbols, there are 2^7 =128 possible models; <u>in three of these, KB is true</u>

通过枚举推理

- 用于判断蕴涵 KB ⊨ α 的真值表枚举推理算法

```
function TT-ENTAILS?(KB, α) returns true or false
    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))
```

如果KB在模型model中为真，PL-Ture?(KB, model)返回真

- 对所有模型来说，深度优先的枚举（递归枚举）是完备的

- 对于n个符号, 时间复杂度为$O(2^n)$, 空间复杂度为$O(n)$

推理: 判断逻辑蕴涵

- Method 1: ***model-checking*（模型检验）**
  - 枚举所有的模型(可能世界), 并验证语句在所有模型中为真
- Method 2: Application of inference rules **（推理规则）**
  - 在知识库的语句上，直接应用推理规则，构建目标语句的证明
- Method 3: ***theorem-proving*（定理证明）**

7.5.1 推理规则

- **Modus Ponens:**

  假言推理规则

$$\frac{\alpha \Rightarrow \beta, \qquad \alpha}{\beta}$$

- **And-Elimination:**

  消去合取词

$$\frac{\alpha \wedge \beta}{\alpha}$$

- **Other rules:** 逻辑等价

例如：双向蕴涵 $\dfrac{\alpha \Leftrightarrow \beta}{(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)}$ and $\dfrac{(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)}{\alpha \Leftrightarrow \beta}$

逻辑等价性

- 任意两个语句是逻辑等价的 iff 它们在相同模型中互相蕴涵:

α ≡ ß iff α ╞ β and β ╞ α

交换律

$$(\alpha \wedge \beta) \equiv (\beta \wedge \alpha) \quad \text{commutativity of } \wedge$$
$$(\alpha \vee \beta) \equiv (\beta \vee \alpha) \quad \text{commutativity of } \vee$$

结合律

$$((\alpha \wedge \beta) \wedge \gamma) \equiv (\alpha \wedge (\beta \wedge \gamma)) \quad \text{associativity of } \wedge$$
$$((\alpha \vee \beta) \vee \gamma) \equiv (\alpha \vee (\beta \vee \gamma)) \quad \text{associativity of } \vee$$
$$\neg(\neg\alpha) \equiv \alpha \quad \text{double-negation elimination}$$

$$(\alpha \Rightarrow \beta) \equiv (\neg\beta \Rightarrow \neg\alpha) \quad \text{contraposition}$$

假言易位 逆否命题的逻辑等价

$$(\alpha \Rightarrow \beta) \equiv (\neg\alpha \vee \beta) \quad \text{implication elimination}$$

消去蕴含词

$$(\alpha \Leftrightarrow \beta) \equiv ((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)) \quad \text{biconditional elimination}$$
$$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta) \quad \text{de Morgan}$$
$$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta) \quad \text{de Morgan}$$

De Morgan定律

分配律

$$(\alpha \wedge (\beta \vee \gamma)) \equiv ((\alpha \wedge \beta) \vee (\alpha \wedge \gamma)) \quad \text{distributivity of } \wedge \text{ over } \vee$$
$$(\alpha \vee (\beta \wedge \gamma)) \equiv ((\alpha \vee \beta) \wedge (\alpha \vee \gamma)) \quad \text{distributivity of } \vee \text{ over } \wedge$$

Wumpus world sentences

$R_1: \neg P_{1,1}$

$R_2: \neg B_{1,1}$

$R_3: B_{2,1}$

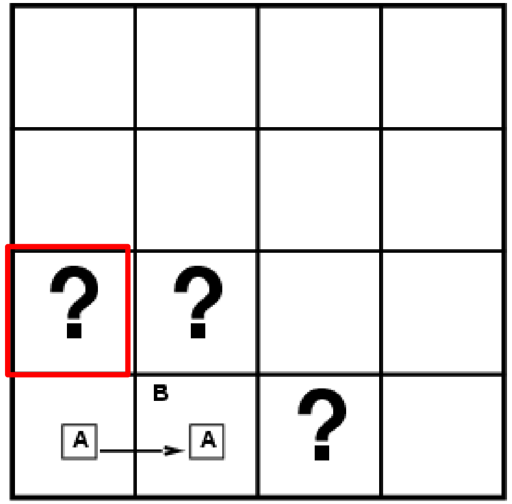
$R_4: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

$R_5: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

证明：KB $\models \alpha$

KB：$R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5$

α : $\neg P_{1,2}$



消去双向蕴涵to R_4

$R_6: (B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

Wumpus world sentences

$R_1: \neg P_{1,1}$

$R_2: \neg B_{1,1}$

$R_3: B_{2,1}$

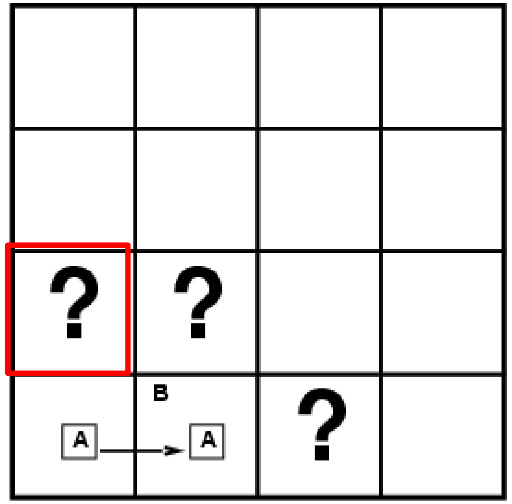
$R_4: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

$R_5: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

证明：KB $\models$ α

KB： $R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5 \wedge R_6$

α : $\neg P_{1,2}$



$R_6 : (B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

消去合取词 to R_6

$R_7 : (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}$

Wumpus world sentences

$R_1: \neg P_{1,1}$

$R_2: \neg B_{1,1}$

$R_3: B_{2,1}$

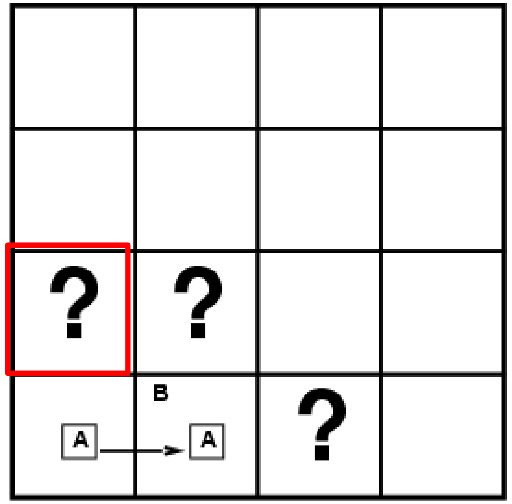
$R_4: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

$R_5: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

证明：KB $\models$ α

KB：$R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5 \wedge R_6 \wedge R_7$

α : $\neg P_{1,2}$



$R_6: (B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

$R_7: (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}$

逆否命题的逻辑等价 of R_7

$R_8: \neg B_{1,1} \Rightarrow \neg (P_{1,2} \vee P_{2,1})$

Wumpus world sentences

R_1: $\neg P_{1,1}$

R_2: $\neg B_{1,1}$

R_3: $B_{2,1}$

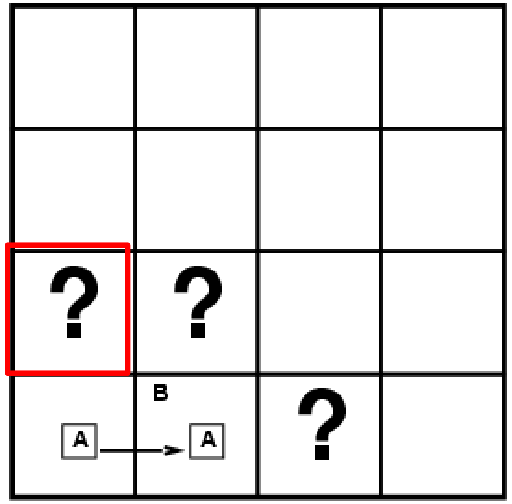
R_4: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

R_5: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

证明：KB $\models$ α

KB：$R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5 \wedge R_6 \wedge R_7 \wedge R_8$

α : $\neg P_{1,2}$



R_6 :$(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

R_7 : $(P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}$

R_8: $\neg B_{1,1} \Rightarrow \neg (P_{1,2} \vee P_{2,1})$

假言推理 with R_8 and R_2 (i.e., $\neg B_{1,1}$)

R_9 : $\neg(P_{1,2} \vee P_{2,1})$

Wumpus world sentences

$R_1: \neg P_{1,1}$

$R_2: \neg B_{1,1}$

$R_3: B_{2,1}$

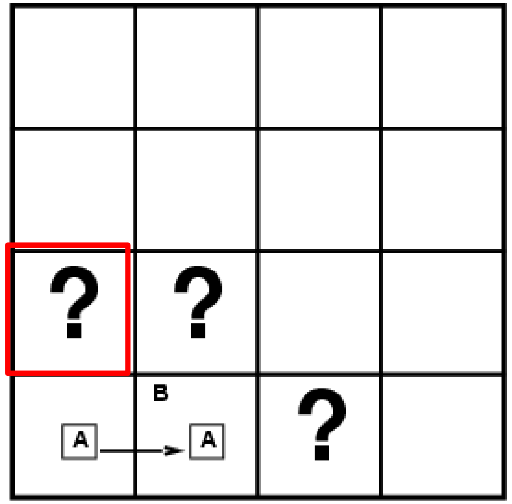
$R_4: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$

$R_5: B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$

证明：KB $\models \alpha$

KB： $R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5 \wedge R_6 \wedge R_7 \wedge R_8 \wedge R_9$

α : $\neg P_{1,2}$



$R_6: (B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

$R_7: (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}$

$R_8: \neg B_{1,1} \Rightarrow \neg (P_{1,2} \vee P_{2,1})$

$R_9: \neg(P_{1,2} \vee P_{2,1})$

上述过程是手工给出的，如何利用搜索算法来找出证明序列？

De Morgan 定律 to R_9

$R_{10}: \neg P_{1,2} \wedge \neg P_{2,1}$

结论： KB $\models \alpha$

消去合取词 to R_{10}

$R_{11}: \neg P_{1,2}$

推理: 判断逻辑蕴涵

- Method 2: Application of inference rules（推理规则）
  - 在知识库的语句上，直接应用推理规则，构建目标语句的证明
  - 找出证明序列的搜索算法:
    - 问题描述:
      - 初始状态：初始知识库
      - 行动：行动集合由应用于语句的所有推理规则组成，要匹配推理规则的上半部分
      - 结果：行动的结果是将推理规则的下半部分的语句实例加入知识库
      - 目标：包含要证明的语句的状态
    - 解序列：找出一条证明序列（动作序列），每个动作是在*语句上应用*规则

推理: 判断逻辑蕴涵

- Method 1: ***model-checking*（模型检验）**
  - 枚举所有的模型(可能世界), 并验证语句在所有模型中为真
- Method 2: Application of inference rules **（推理规则）**
  - 在知识库的语句上，直接应用推理规则，构建目标语句的证明
- Method 3: ***theorem-proving*（定理证明）**
  - 归结证明
  - 当它和任何一个完备的搜索算法相结合时，可以得到完备的推理算法

7.5.2 归结证明

- **归结 推理规则**
  - 单元归结规则：选取一个子句（文字的析取式）和一个文字，生成一个新的子句

$$\frac{l_1 \vee \ldots \vee l_k, \qquad m}{l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee \ldots \vee l_k}$$

其中, 每个l都是一个文字， l_i和 m是互补文字(一个是另一个的否定式)

E.g., $\frac{P_{1,1} \vee P_{3,1}, \quad \neg P_{1,1}}{P_{3,1}}$

文字是指原子语句(正文字)或原子语句的否定式 (负文字)

7.5.2 归结Resolution

- **归结 推理规则**
  - **全归结**: 选取两个子句（<u>文字的析取式</u>），生成一个新的子句（包含除了两个互补文字之外的原始子句中的所有文字）

$$\frac{l_1 \vee \ldots \vee l_k, \qquad m_1 \vee \ldots \vee m_n}{l_1 \vee \ldots \vee l_{i-1} \vee l_{i+1} \vee \ldots \vee l_k \vee m_1 \vee \ldots \vee m_{j-1} \vee m_{j+1} \vee \ldots \vee m_n}$$

其中 l_i 和 m_j 是互补文字(一个是另一个的否定式)

注意：结果子句中每个文字只出现一次，多余副本将被归并。

如， $A \vee B$ 与 $A \vee \neg B$ 归结，得到 $A \vee A$，简化为A

7.5.2 Resolution 归结

Resolution inference rule 只应用于子句（文字的析取式），那么对于所有的命题逻辑，如何实现完备推理呢？

1）语句转换为合取范式

2）归结算法

合取范式CNF

以子句的合取式表达的语句被称为**合取范式**

Conjunctive Normal Form (CNF)

conjunction of disjunctions of literals clauses

E.g., **$(A \vee \neg B) \wedge (B \vee \neg C \vee \neg D)$**

定理：命题逻辑的每个语句逻辑上都等价于某个子句的合取式

语句转换成CNF的过程：

1. **消去等价词⇔**
2. **消去蕴含词⇒**
3. **否定词¬内移**
4. **使用分配率，将∨对∧进行分配**

转换成CNF

Conjunctive Normal Form (CNF)

conjunction of disjunctions of literals clauses

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

1. 消除 $\Leftrightarrow$, 用 $(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$代替$\alpha \Leftrightarrow \beta$:
   $(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$

2. 消除 $\Rightarrow$, 用 $\neg\alpha \vee \beta$代替$\alpha \Rightarrow \beta$:
   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg(P_{1,2} \vee P_{2,1}) \vee B_{1,1})$

3. 将 $\neg$ 移进去，利用 de Morgan‘s 规则 and 双重否定:
   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge ((\neg P_{1,2} \wedge \neg P_{2,1}) \vee B_{1,1})$

4. 使用分配率 ($\wedge$ over $\vee$) 和结合律:
   $(\neg B_{1,1} \vee P_{1,2} \vee P_{2,1}) \wedge (\neg P_{1,2} \vee B_{1,1}) \wedge (\neg P_{2,1} \vee B_{1,1})$

归结算法 Resolution algorithm

- 证明KB |= α
- 归结证明步骤：

1）反证法，为了证明 KB |= α，需要证明KB ∧ ¬ α 是不可满足的

2）$KB \wedge \neg\alpha$转化为合取范式

3）利用归结规则证明$KB \wedge \neg\alpha$不可满足

For example: KB |= α

$$KB = (B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})) \wedge \neg B_{1,1}$$
$$\alpha = \neg P_{1,2}$$

1）反证法

$$KB \wedge \neg\alpha \equiv (B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})) \wedge \neg B_{1,1} \wedge P_{1,2}$$

Resolution example

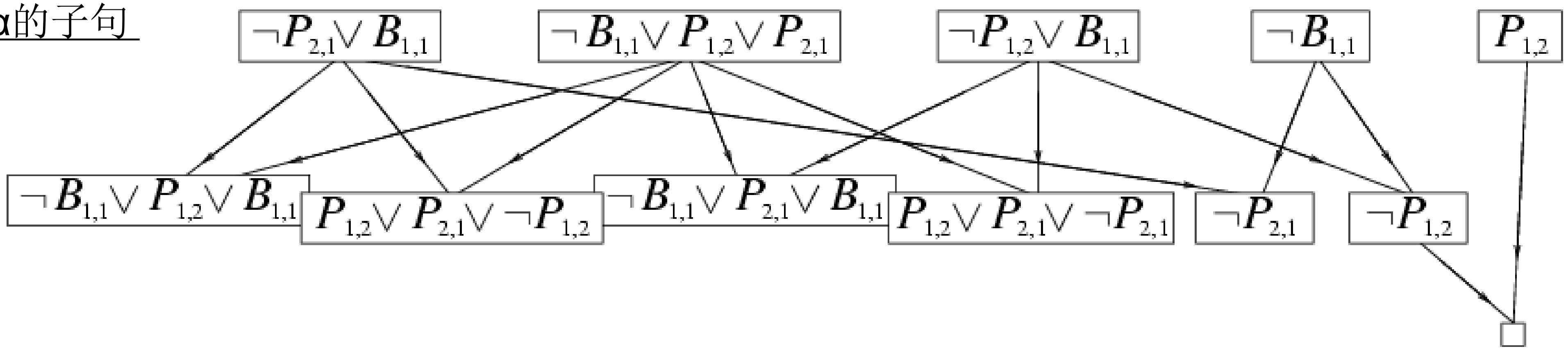
1） $KB \wedge \neg \alpha \equiv (B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})) \wedge \neg B_{1,1} \wedge P_{1,2}$

2）转为合取范式

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

3）两子句运用归结规则

$KB \wedge \neg \alpha$的子句



If a set of clauses is unsatisfiable, then the resolution closure of those clauses contains the empty clause.

空子句不包含任何文字的子句，是永假的

归结算法

- 用反证法证明
- 为了证明 KB |= α，需要证明KB ∧ ¬ α 是不可满足的

function PL-RESOLUTION(KB, α) **returns** *true* or *false*
  inputs: KB, the knowledge base, a sentence in propositional logic
    α, the query, a sentence in propositional logic

  $clauses \leftarrow$ the set of clauses in the CNF representation of $KB \wedge \neg\alpha$ 转化为CNF
  $new \leftarrow \{\ \}$
  loop do
    for each pair of clauses C_i, C_j **in** $clauses$ **do**   任意两个子句，运用归结规则
      $resolvents \leftarrow$ PL-RESOLVE(C_i, C_j)
      if $resolvents$ contains the empty clause **then return** *true*   归结出空子句
      $new \leftarrow new \cup resolvents$
    if $new \subseteq clauses$ **then return** *false*   没有新的语句
    $clauses \leftarrow clauses \cup new$

函数PL-RESOLVE返回对两个输入子句进行归结得到的所有结果子句的集合

练习题：合取范式与归结证明

1、将语句转换为合取范式（CNF）

7.20 Convert the following set of sentences to clausal form.

S1: $A \Leftrightarrow (B \vee E)$.
S2: $E \Rightarrow D$.
S3: $C \wedge F \Rightarrow \neg B$.
S4: $E \Rightarrow B$.
S5: $B \Rightarrow F$.
S6: $B \Rightarrow C$

2、归结证明

7.12 Use resolution to prove the sentence $\neg A \wedge \neg B$ from the clauses in Exercise 7.20.

总结

- 逻辑智能体在知识库中应用推理来得到新的信息并做决策
- 逻辑的基本概念：
  - 语法：语句的形式化结构
  - 语义：模型中为真的语句
  - 蕴含：给定一个语句的真假时另一个语句必然的真假
  - 推理：从其他语句中推导出的语句
- 归结对于命题逻辑是完备的

  对于Horn子句，前向链接和后向链接是线性的，完备的