Propositional Logic

Outline

I. Semantics and truth table

II. Knowledge base for the Wumpus world

III. Inference by model checking

IV. Inference by rules

* Figures are from the textbook site unless the source is specifically cited.

I. Semantics of Propositional Logic

A model fixes the truth value (*true* or *false*) for every proposition symbols.

$$m_1 = \{P_{1,2} = \textit{false}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$$
$$m_2 = \{P_{1,2} = \textit{true}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$$

Semantics defines the rules for determining the truth of a sentence w.r.t. any model.

I. Semantics of Propositional Logic

A model fixes the truth value (*true* or *false*) for every proposition symbols.

$$m_1 = \{P_{1,2} = \textit{false}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$$
$$m_2 = \{P_{1,2} = \textit{true}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$$

Semantics defines the rules for determining the truth of a sentence w.r.t. any model.

The truth value of any sentence can be computed once we know

- how to evaluate the truth of atomic sentences;
- how to compute the truth of sentences formed with each of the five connectives.

Determining the Truth Value

Atomic sentences:

- *true* is true in every model.
- *false* is false in every model.
- The truth value of every other proposition symbol must be specified in a model m.

Determining the Truth Value

Atomic sentences:

- *true* is true in every model.
- *false* is false in every model.
- The truth value of every other proposition symbol must be specified in a model m.

Complex sentences in the model m:

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

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*

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*

In the model $m_1 = \{P_{1,2} =$ *false*, $P_{2,2} =$ *false*, $P_{3.1} =$ *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*

In the model $m_1 = \{P_{1,2} = \textit{false}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$

No pit in [1,2].

Truth Table

P	Q	$\neg P$	$P \land Q$	$P \lor Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
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*

In the model $m_1 = \{P_{1,2} = \textit{false}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$

No pit in [1,2].

$$\neg P_{1,2} \land (P_{2,2} \lor P_{3,1}) = \neg \textit{false} \land (\textit{false} \lor \textit{true})$$

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*

In the model $m_1 = \{P_{1,2} = \textit{false}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$

No pit in [1,2].

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

$$= \textit{true} \wedge \textit{true}$$

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*

In the model $m_1 = \{P_{1,2} = \textit{false}, P_{2,2} = \textit{false}, P_{3.1} = \textit{true}\}$

No pit in [1,2].

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

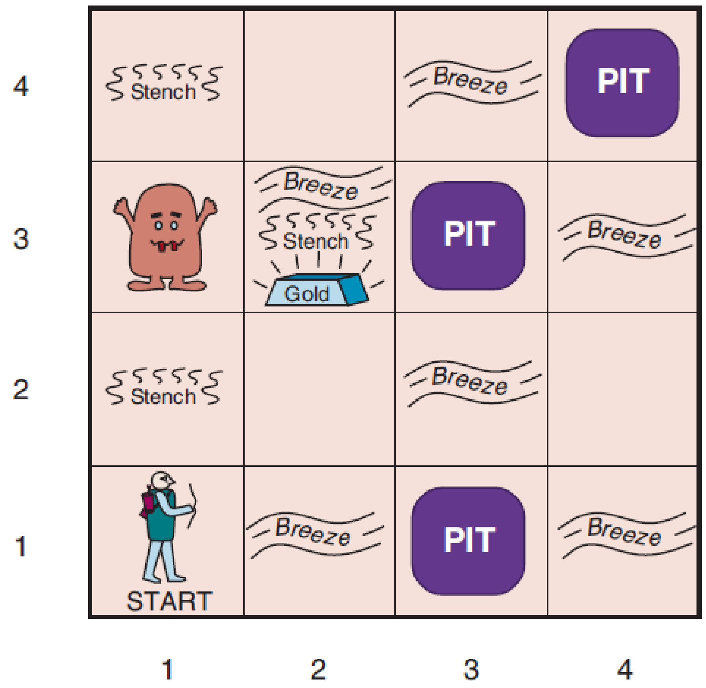
$$= \textit{true} \wedge \textit{true}$$

$$= \textit{true}$$

II. Knowledge Base for the Wumpus World

Proposition symbols

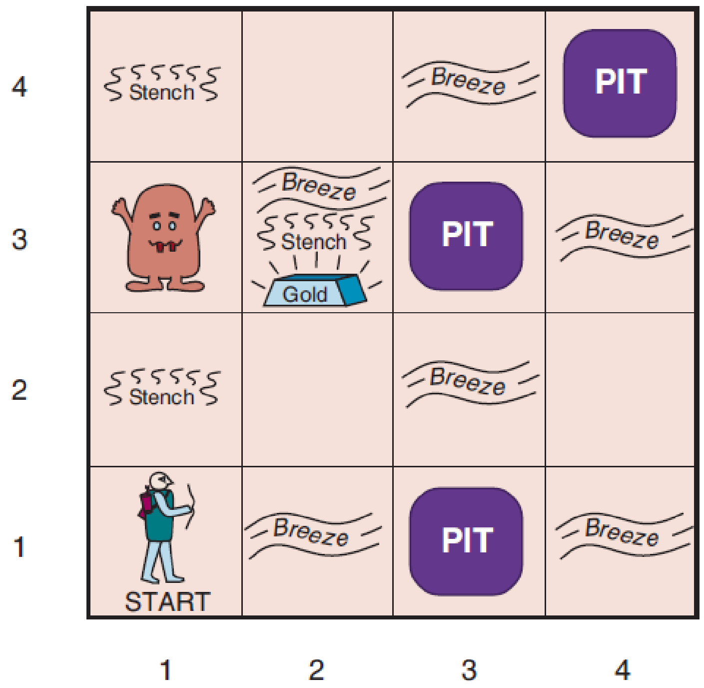
- $P_{x,y}$ is true if there is a pit in $[x, y]$.
- $W_{x,y}$ is true if there is a wumpus in $[x, y]$, dead or alive.
- $B_{x,y}$ is true if the agent perceives a breeze in $[x, y]$.
- $S_{x,y}$ is true if the agent perceives a stench in $[x, y]$.



Knowledge Base (cont'd)

- General knowledge (partial – only for relevant squares):
  - There is no pit in [1,1].

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



Knowledge Base (cont'd)

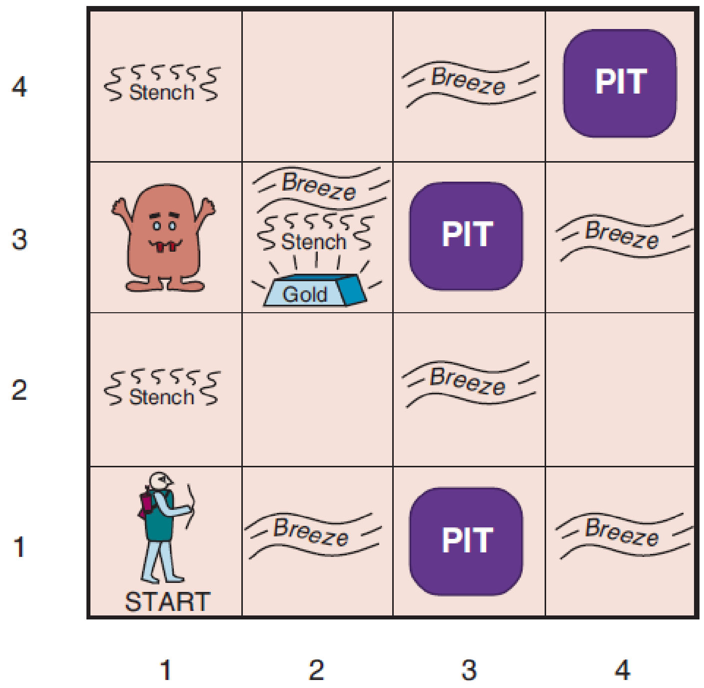
- General knowledge (partial – only for relevant squares):
  - There is no pit in [1,1].

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

  - A square is breezy if and only if a neighboring square has a pit.

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

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



Knowledge Base (cont'd)

- General knowledge (partial – only for relevant squares):
  - There is no pit in [1,1].

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

  - A square is breezy if and only if a neighboring square has a pit.

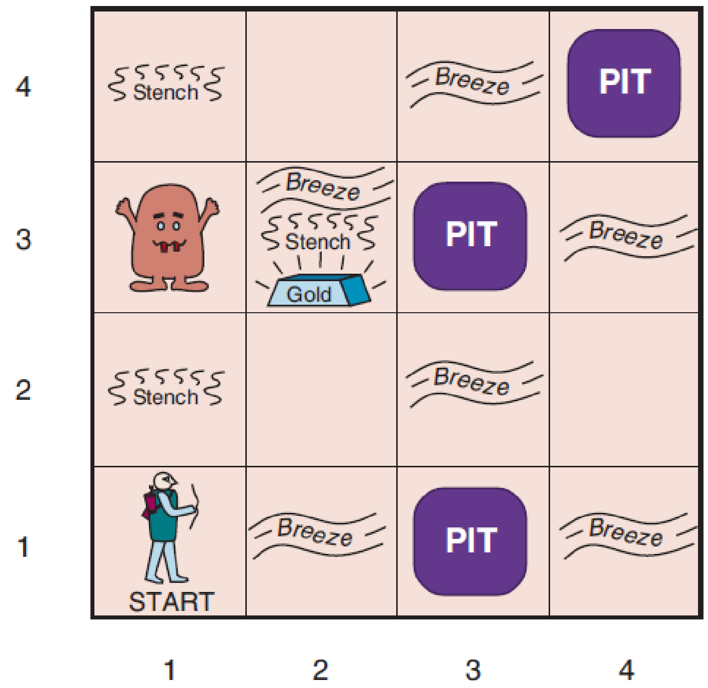
    $R_2: B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$

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

- Percepts for the first two squares:

  $R_4: \neg B_{1,1}$

  $R_5: B_{2,1}$



Knowledge Base (cont'd)

- General knowledge (partial – only for relevant squares):
  - There is no pit in [1,1].

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

  - A square is breezy if and only if a neighboring square has a pit.

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

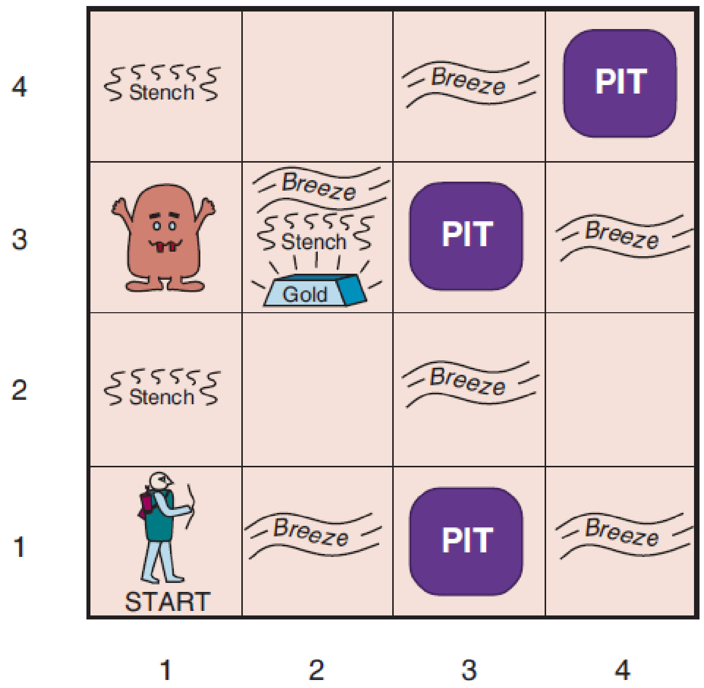
$$R_3:\ B_{2,1} \Leftrightarrow (P_{1,1} \lor P_{2,2} \lor P_{3,1})$$

- Percepts for the first two squares:

$$R_4:\ \neg B_{1,1}$$

$$R_5:\ B_{2,1}$$

$$KB = \{R_1, R_2, R_3, R_4, R_5\}$$



III. Inference by Model Checking

$KB = \{R_1, R_2, R_3, R_4, R_5\}$

Q. Does $KB \models \neg P_{1,2}$?

III. Inference by Model Checking

$KB = \{R_1, R_2, R_3, R_4, R_5\}$

Q. Does $KB \models \neg P_{1,2}$?

Enumerate the models of KB and check if $\neg P_{1,2}$ is true in every model.

III. Inference by Model Checking

$KB = \{R_1, R_2, R_3, R_4, R_5\}$

Q. Does $KB \vDash \neg P_{1,2}$?

Enumerate the models of KB and check if $\neg P_{1,2}$ is true in every model.

Relevant propositions: $B_{1,1}$, $B_{2,1}$, $P_{1,1}$, $P_{1,2}$ $P_{2,1}$ $P_{2,2}$ $P_{3,1}$

III. Inference by Model Checking

$KB = \{R_1, R_2, R_3, R_4, R_5\}$

Q. Does $KB \vDash \neg P_{1,2}$?

Enumerate the models of KB and check if $\neg P_{1,2}$ is true in every model.

Relevant propositions: $B_{1,1}$, $B_{2,1}$, $P_{1,1}$, $P_{1,2}$ $P_{2,1}$ $P_{2,2}$ $P_{3,1}$

⇓

$2^7 = 128$ possible models!

Truth Table Enumeration

128 rows

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

Truth Table Enumeration

$KB = \{R_1, R_2, R_3, R_4, R_5\}$ is true in only 3 models.

128 rows

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

Truth Table Enumeration

$KB = \{R_1, R_2, R_3, R_4, R_5\}$ is true in only 3 models.

$\neg P_{1,2}$ is true in all 3.

128 rows

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

Truth Table Enumeration

$KB = \{R_1, R_2, R_3, R_4, R_5\}$ is true in only 3 models.

$\neg P_{1,2}$ is true in all 3.

$\Rightarrow$ $KB \vDash \neg P_{1,2}$

128 rows

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

Truth Table Enumeration

$KB = \{R_1, R_2, R_3, R_4, R_5\}$ is true in only 3 models.

$\neg P_{1,2}$ is true in all 3.

$\Rightarrow KB \vDash \neg P_{1,2}$

$P_{2,2}$ is true in only 2 of 3.

128 rows

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

Truth Table Enumeration

$KB = \{R_1, R_2, R_3, R_4, R_5\}$ is true in only 3 models.

$\neg P_{1,2}$ is true in all 3.

⇨ $KB \vDash \neg P_{1,2}$

$P_{2,2}$ is true in only 2 of 3. ⇨ No inference of $KB \vDash P_{2,2}$

128 rows

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

Bad News

Suppose KB and α have n symbols. $\Longrightarrow$ 2^n models!

The propositional entailment problem of showing $KB \vDash \alpha$ by truth table enumeration requires

$\Theta(2^n n)$ time

$O(n)$ space (not bad)

The problem is co NP-complete (likely not easier than NP-complete).

IV. Logical Equivalence

Theorem proving: Apply rules of inference directly to the sentences in KB to construct a proof of a sentence without consulting models.

IV. Logical Equivalence

Theorem proving: Apply rules of inference directly to the sentences in KB to construct a proof of a sentence without consulting models.

Two sentences α and β are *logically equivalent* if $M(\alpha) = M(\beta)$.

$$(\alpha \equiv \beta)$$

IV. Logical Equivalence

Theorem proving: Apply rules of inference directly to the sentences in KB to construct a proof of a sentence without consulting models.

Two sentences α and β are *logically equivalent* if $M(\alpha) = M(\beta)$.

$(\alpha \equiv \beta)$

set of models for α

IV. Logical Equivalence

Theorem proving: Apply rules of inference directly to the sentences in KB to construct a proof of a sentence without consulting models.

Two sentences α and β are *logically equivalent* if $M(\alpha) = M(\beta)$.

$(\alpha \equiv \beta)$

set of models for α

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

IV. Logical Equivalence

Theorem proving: Apply rules of inference directly to the sentences in KB to construct a proof of a sentence without consulting models.

Two sentences α and β are *logically equivalent* if $M(\alpha) = M(\beta)$.

$(\alpha \equiv \beta)$

set of models for α

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

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

Validity

A sentence is *valid* if it is true in all models.

$$P \vee \neg P$$

Valid sentences are *tautologies*.

Validity

A sentence is *valid* if it is true in all models.

$$P \lor \neg P$$

Valid sentences are *tautologies*.

$\alpha\alpha\Rightarrow\beta\beta$) *is valid.*
$\alpha\alpha\Rightarrow\beta\beta$) *is valid.*
if and only if the sentence $(\alpha \Rightarrow \beta)$ is valid.
if and only if the sentence $(\alpha \Rightarrow \beta)$ is valid.

Validity

A sentence is *valid* if it is true in all models.

$$P \lor \neg P$$

Valid sentences are *tautologies*.

αα⇒ββ) is valid.
αα⇒ββ) is valid.
if and only if the sentence $(\alpha \Rightarrow \beta)$ is valid.
if and only if the sentence $(\alpha \Rightarrow \beta)$ is valid.
We can decide if $\alpha \vDash \beta$ by checking that $\alpha \Rightarrow \beta$ is a tautology.

Satisfiability

A sentence is *satisfiable* if it is true in, or satisfied by, some model.

R_1	R_2	R_3	R_4	R_5	KB
true	*true*	*true*	*true*	*false*	*false*
true	*true*	*false*	*true*	*false*	*false*
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
true	*true*	*false*	*true*	*true*	*false*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*false*	*false*	*true*	*true*	*false*
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
false	*true*	*true*	*false*	*true*	*false*

…

$(R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5)$ is satisfiable.

Satisfiability

A sentence is *satisfiable* if it is true in, or satisfied by, some model.

R_1	R_2	R_3	R_4	R_5	KB
true	*true*	*true*	*true*	*false*	*false*
true	*true*	*false*	*true*	*false*	*false*
⋮	⋮	⋮	⋮	⋮	⋮
true	*true*	*false*	*true*	*true*	*false*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*false*	*false*	*true*	*true*	*false*
⋮	⋮	⋮	⋮	⋮	⋮
false	*true*	*true*	*false*	*true*	*false*

...

$(R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5)$ is satisfiable.

The SAT problem: Determining the satisfiability of sentences in propositional logic.

Satisfiability

A sentence is *satisfiable* if it is true in, or satisfied by, some model.

...

R_1	R_2	R_3	R_4	R_5	KB
true	*true*	*true*	*true*	*false*	*false*
true	*true*	*false*	*true*	*false*	*false*
⋮	⋮	⋮	⋮	⋮	⋮
true	*true*	*false*	*true*	*true*	*false*
true	*true*	*true*	*true*	*true*	<u>*true*</u>
true	*true*	*true*	*true*	*true*	<u>*true*</u>
true	*true*	*true*	*true*	*true*	<u>*true*</u>
true	*false*	*false*	*true*	*true*	*false*
⋮	⋮	⋮	⋮	⋮	⋮
false	*true*	*true*	*false*	*true*	*false*

$(R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5)$ is satisfiable.

The SAT problem: Determining the satisfiability of sentences in propositional logic.

first NP-complete problem

Satisfiability

A sentence is *satisfiable* if it is true in, or satisfied by, some model.

R_1	R_2	R_3	R_4	R_5	KB
true	*true*	*true*	*true*	*false*	*false*
true	*true*	*false*	*true*	*false*	*false*
⋮	⋮	⋮	⋮	⋮	⋮
true	*true*	*false*	*true*	*true*	*false*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*false*	*false*	*true*	*true*	*false*
⋮	⋮	⋮	⋮	⋮	⋮
false	*true*	*true*	*false*	*true*	*false*

$(R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5)$ is satisfiable.

The SAT problem: Determining the satisfiability of sentences in propositional logic.

first NP-complete problem

$\alpha \vDash \beta$ if and only if the sentence $(\alpha \wedge \neg\beta)$ is unsatisfiable.

Satisfiability

A sentence is *satisfiable* if it is true in, or satisfied by, some model.

R_1	R_2	R_3	R_4	R_5	KB
true	*true*	*true*	*true*	*false*	*false*
true	*true*	*false*	*true*	*false*	*false*
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
true	*true*	*false*	*true*	*true*	*false*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*true*	*true*	*true*	*true*	*true*
true	*false*	*false*	*true*	*true*	*false*
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
false	*true*	*true*	*false*	*true*	*false*

...

$(R_1 \wedge R_2 \wedge R_3 \wedge R_4 \wedge R_5)$ is satisfiable.

The SAT problem: Determining the satisfiability of sentences in propositional logic.

first NP-complete problem

Proof by contradiction

$\alpha \vDash \beta$ if and only if the sentence $(\alpha \wedge \neg\beta)$ is unsatisfiable.

Inference Rules

Modus Ponens

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

$\alpha \qquad \Rightarrow \qquad \beta$

If **today is Tuesday**, then **John will go to campus**.

Today is Tuesday.

Therefore, **John will go to campus**.

Inference Rules

Modus Ponens

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

$\alpha \qquad \Rightarrow \qquad \beta$

If **today is Tuesday**, then **John will go to campus**.

Today is Tuesday.

Therefore, **John will go to campus**.

And-elimination

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

$\alpha \qquad \wedge \qquad \beta$

A star is a sphere of gas, and **it is held together by its own gravity.**

A star is a sphere of gas.

Other Inference Rules

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

Other Inference Rules

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

$$\neg(\alpha \vee \beta)$$

Other Inference Rules

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

$$
\frac{\neg(\alpha \vee \beta)}{\neg\alpha \wedge \neg\beta} \quad \text{De Morgan}
$$

Other Inference Rules

$(\alpha \wedge \beta)$	$\equiv$	$(\beta \wedge \alpha)$	commutativity of $\wedge$
$(\alpha \vee \beta)$	$\equiv$	$(\beta \vee \alpha)$	commutativity of $\vee$
$((\alpha \wedge \beta) \wedge \gamma)$	$\equiv$	$(\alpha \wedge (\beta \wedge \gamma))$	associativity of $\wedge$
$((\alpha \vee \beta) \vee \gamma)$	$\equiv$	$(\alpha \vee (\beta \vee \gamma))$	associativity of $\vee$
$\neg(\neg\alpha)$	$\equiv$	α	double-negation elimination
$(\alpha \Rightarrow \beta)$	$\equiv$	$(\neg\beta \Rightarrow \neg\alpha)$	contraposition
$(\alpha \Rightarrow \beta)$	$\equiv$	$(\neg\alpha \vee \beta)$	implication elimination
$(\alpha \Leftrightarrow \beta)$	$\equiv$	$((\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha))$	biconditional elimination
$\neg(\alpha \wedge \beta)$	$\equiv$	$(\neg\alpha \vee \neg\beta)$	De Morgan
$\neg(\alpha \vee \beta)$	$\equiv$	$(\neg\alpha \wedge \neg\beta)$	De Morgan
$(\alpha \wedge (\beta \vee \gamma))$	$\equiv$	$((\alpha \wedge \beta) \vee (\alpha \wedge \gamma))$	distributivity of $\wedge$ over $\vee$
$(\alpha \vee (\beta \wedge \gamma))$	$\equiv$	$((\alpha \vee \beta) \wedge (\alpha \vee \gamma))$	distributivity of $\vee$ over $\wedge$

$$\frac{\neg(\alpha \vee \beta)}{\neg\alpha \wedge \neg\beta} \quad \text{De Morgan}$$

$$\frac{\neg\alpha \wedge \neg\beta}{\neg\beta} \quad \text{and-elimination}$$

Applying Rules to the Wumpus World

KB:

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

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

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

$R_4: \neg B_{1,1}$

$R_5: B_{2,1}$

Applying Rules to the Wumpus World

KB:

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

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

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

$R_4: \neg B_{1,1}$

$R_5: B_{2,1}$

Proof for $\neg P_{1,2}$

Applying Rules to the Wumpus World

KB:

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

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

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

$R_4: \neg B_{1,1}$

$R_5: B_{2,1}$

Proof for $\neg P_{1,2}$

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

Applying Rules to the Wumpus World

KB:

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

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

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

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2:\ B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})}{R_6:\ \left(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})\right) \wedge \left((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}\right)}$$ biconditional elimination

Applying Rules to the Wumpus World

KB:

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

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

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

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2:\ B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})}{R_6:\ \left(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})\right) \wedge \left((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}\right)}$$

biconditional elimination

$$\frac{R_6}{R_7:\ (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}}$$

and-elimination

Applying Rules to the Wumpus World

KB:

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

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

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

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2\text{: } B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})}{R_6\text{: } \left(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})\right) \wedge \left((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}\right)}$$

biconditional elimination

and-elimination

$$\frac{R_7\text{: } (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}}{R_8\text{: } \neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})}$$

logical equivalence

Applying Rules to the Wumpus World

KB:

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

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

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

$R_4: \neg B_{1,1}$

$R_5: B_{2,1}$

Proof for $\neg P_{1,2}$

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

biconditional elimination

$R_6: \left(B_{1,1} \Rightarrow (P_{1,2} \lor P_{2,1})\right) \land \left((P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1}\right)$

and-elimination

$R_7: \left(P_{1,2} \lor P_{2,1}\right) \Rightarrow B_{1,1}$

logical equivalence

$R_4: \neg B_{1,1}$

$R_8: \neg B_{1,1} \Rightarrow \neg\left(P_{1,2} \lor P_{2,1}\right)$

Applying Rules to the Wumpus World

KB:

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

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

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

$R_4: \neg B_{1,1}$

$R_5: B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2: B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})}{R_6: \left(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})\right) \wedge \left((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}\right)}$$

biconditional elimination

$$\frac{R_6}{R_7: \left(P_{1,2} \vee P_{2,1}\right) \Rightarrow B_{1,1}}$$

and-elimination

$$\frac{R_7}{R_8: \neg B_{1,1} \Rightarrow \neg\left(P_{1,2} \vee P_{2,1}\right)}$$

logical equivalence

$$\frac{R_4: \neg B_{1,1} \qquad R_8: \neg B_{1,1} \Rightarrow \neg\left(P_{1,2} \vee P_{2,1}\right)}{R_9: \neg\left(P_{1,2} \vee P_{2,1}\right)}$$

modus ponens

Applying Rules to the Wumpus World

KB:

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

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

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

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2:\ B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})}{R_6:\ \left(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})\right) \wedge \left((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}\right)}$$

biconditional elimination

$$\frac{R_6}{R_7:\ (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}}$$

and-elimination

$$\frac{R_7}{R_8:\ \neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})}$$

logical equivalence

$$\frac{R_4:\ \neg B_{1,1} \qquad R_8:\ \neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})}{R_9:\ \neg(P_{1,2} \vee P_{2,1})}$$

modus ponens

$$\frac{R_9}{R_{10}:\ \neg P_{1,2} \wedge \neg P_{2,1}}$$

De Morgan’s rule

Applying Rules to the Wumpus World

KB:

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

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

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

R_4: $\neg B_{1,1}$

R_5: $B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2:\ B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})}{R_6:\ \left(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})\right) \wedge \left((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}\right)}$$ biconditional elimination

$$\frac{R_6}{R_7:\ (P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}}$$ and-elimination

$$\frac{R_7}{R_8:\ \neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})}$$ logical equivalence

$$\frac{R_4:\ \neg B_{1,1} \qquad R_8:\ \neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})}{R_9:\ \neg(P_{1,2} \vee P_{2,1})}$$ modus ponens

$$\frac{R_9}{R_{10}:\ \neg P_{1,2} \wedge \neg P_{2,1}}$$ De Morgan's rule

$$\frac{R_{10}}{R_{11}:\ \neg P_{1,2}}$$ and-elimination

Applying Rules to the Wumpus World

KB:

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

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

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

$R_4: \neg B_{1,1}$

$R_5: B_{2,1}$

Proof for $\neg P_{1,2}$

$$\frac{R_2:\ B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})}{R_6:\ \left(B_{1,1} \Rightarrow (P_{1,2} \lor P_{2,1})\right) \land \left((P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1}\right)}$$ biconditional elimination

$$\frac{\quad}{R_7:\ (P_{1,2} \lor P_{2,1}) \Rightarrow B_{1,1}}$$ and-elimination

$$\frac{\quad}{R_8:\ \neg B_{1,1} \Rightarrow \neg(P_{1,2} \lor P_{2,1})}$$ logical equivalence

$$\frac{R_4:\ \neg B_{1,1} \qquad R_8:\ \neg B_{1,1} \Rightarrow \neg(P_{1,2} \lor P_{2,1})}{R_9:\ \neg(P_{1,2} \lor P_{2,1})}$$ modus ponens

$$\frac{\quad}{R_{10}:\ \neg P_{1,2} \land \neg P_{2,1}}$$ De Morgan's rule

$$\frac{\quad}{R_{11}:\ \neg P_{1,2}}$$ and-elimination

Searching for Proofs

- INITIAL STATE: the initial KB.
- ACTIONS: Apply any inference rule to any sentence that matches its top half.
- RESULT: Add the sentence in the bottom half of the inference rules.
- GOAL: A state containing the sentence to be proved.

Searching for Proofs

- INITIAL STATE: the initial KB.
- ACTIONS: Apply any inference rule to any sentence that matches its top half.
- RESULT: Add the sentence in the bottom half of the inference rules.
- GOAL: A state containing the sentence to be proved.

- You can apply any of the search algorithms.
- Searching for a proof is an alternative to enumerating models.
- It is more efficient because of ignoring irrelevant propositions.

Searching for Proofs

- INITIAL STATE: the initial KB.
- ACTIONS: Apply any inference rule to any sentence that matches its top half.
- RESULT: Add the sentence in the bottom half of the inference rules.
- GOAL: A state containing the sentence to be proved.

♦ You can apply any of the search algorithms.

♦ Searching for a proof is an alternative to enumerating models.

♦ It is more efficient because of ignoring irrelevant propositions.

♠ The truth-table algorithm runs in time $\Theta(2^n n)$ – always the worst case.

Searching for Proofs

- INITIAL STATE: the initial KB.
- ACTIONS: Apply any inference rule to any sentence that matches its top half.
- RESULT: Add the sentence in the bottom half of the inference rules.
- GOAL: A state containing the sentence to be proved.

♦ You can apply any of the search algorithms.

♦ Searching for a proof is an alternative to enumerating models.

♦ It is more efficient because of ignoring irrelevant propositions.

- ♠ The truth-table algorithm runs in time $\Theta(2^n n)$ – always the worst case.
- ♣ Search takes time $O(2^n n)$ – often not the worst case.

Monotonicity

The set of entailed sentences can only *increase* as new information is added to the KB.

$$\text{if } KB \models \alpha \quad \text{then} \quad KB \wedge \beta \models \alpha$$

Monotonicity

The set of entailed sentences can only *increase* as new information is added to the KB.

$$\text{if } KB \models \alpha \quad \text{then} \quad KB \land \beta \models \alpha$$

α: There is no pit in $[1, 2]$.

β: There are exactly eight pits in the wumpus world

Monotonicity

The set of entailed sentences can only *increase* as new information is added to the KB.

$$\text{if } KB \models \alpha \quad \text{then} \quad KB \wedge \beta \models \alpha$$

α: There is no pit in $[1, 2]$.

β: There are exactly eight pits in the wumpus world

Knowing β might help draw additional conclusions, but it cannot invalidate already inferred α.

Monotonicity

The set of entailed sentences can only *increase* as new information is added to the KB.

$$\text{if } KB \vDash \alpha \quad \text{then} \quad KB \wedge \beta \vDash \alpha$$

α: There is no pit in $[1, 2]$.

β: There are exactly eight pits in the wumpus world

Knowing β might help draw additional conclusions, but it cannot invalidate already inferred α.

An inference rule can be applied whenever its premise is found in the KB, regardless of what else is in the KB, and the conclusion must follow.