# Propositional Logic

## Outline

I. Syntax

II. Semantics and truth table

III. Knowledge base for the Wumpus world

IV. Inference by model checking

V. Inference by rules

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

# I. Syntax of Propositional Logic

An *atomic sentence* is a single proposition symbol.

standing for a proposition that has to be either true or false but not both.

$P, Q, R, W_{1,3},$ *FacingEast*

# I. Syntax of Propositional Logic

An *atomic sentence* is a single proposition symbol.

standing for a proposition that has to be either true or false but not both.

$P, Q, R, W_{1,3}$, *FacingEast*

The wumpus is in [1,3].

# I. Syntax of Propositional Logic

An *atomic sentence* is a single proposition symbol.

standing for a proposition that has to be either true or false but not both.

$P, Q, R, W_{1,3}$, *FacingEast*

The wumpus is in [1,3].

Two special proposition symbols:

*True* : always-true proposition
*False*: always-false proposition

# Complex Sentences

A *complex sentence* is constructed from simpler sentences, using parentheses and *logical connectives* (5 in total).

- $\neg$ (not).
  - $\neg P$ is the *negation* of $P$.
  - *literal*: either an atomic sentence or a negated atomic sentence.

# Complex Sentences

A *complex sentence* is constructed from simpler sentences, using parentheses and *logical connectives* (5 in total).

- $\neg$ (not).
  - $\neg P$ is the *negation* of $P$.
  - *literal*: either an atomic sentence or a negated atomic sentence.

$$W_{1,3}, \neg Q$$

# Complex Sentences

A *complex sentence* is constructed from simpler sentences, using parentheses and *logical connectives* (5 in total).

- $\neg$ (not).
  - $\neg P$ is the *negation* of $P$.
  - *literal*: either an atomic sentence or a negated atomic sentence.

$$W_{1,3}, \neg Q$$

- $\wedge$ (and).
  - $W_{1,3} \wedge P_{3,1}$ is a *conjunction* whose parts $W_{1,3}$ and $P_{3,1}$ are *conjuncts*.

There is a pit in [3,1].

# Complex Sentences

A *complex sentence* is constructed from simpler sentences, using parentheses and *logical connectives* (5 in total).

- $\neg$ (not).
  - $\neg P$ is the *negation* of $P$.
  - *literal*: either an atomic sentence or a negated atomic sentence.

$$W_{1,3}, \neg Q$$

- $\wedge$ (and).
  - $W_{1,3} \wedge P_{3,1}$ is a *conjunction* whose parts $W_{1,3}$ and $P_{3,1}$ are *conjuncts*.

    There is a pit in $[3,1]$.
- $\vee$ (or).
  - $(W_{1,3} \wedge P_{3,1}) \vee W_{2,2}$ is a *disjunction* whose parts $(W_{1,3} \wedge P_{3,1})$ and $W_{2,2}$ are *disjuncts*.

# More Logical Connectives

- $\Rightarrow$ (implies).
  - $(W_{1,3} \wedge P_{3,1}) \Rightarrow \neg W_{2,2}$ is an *implication*.

# More Logical Connectives

- $\Rightarrow$ (implies).
  - $(W_{1,3} \land P_{3,1}) \Rightarrow \neg W_{2,2}$ is an *implication*.

*premise* or *antecedent*

# More Logical Connectives

- $\Rightarrow$ (implies).
  - $(W_{1,3} \wedge P_{3,1}) \Rightarrow \neg W_{2,2}$ is an *implication*.

*premise* or *antecedent* — $(W_{1,3} \wedge P_{3,1})$

*conclusion* or *consequent* — $\neg W_{2,2}$

# More Logical Connectives

- $\Rightarrow$ (implies).
  - $(W_{1,3} \wedge P_{3,1}) \Rightarrow \neg W_{2,2}$ is an *implication*.

*premise* or *antecedent* *conclusion* or *consequent*

- $\Leftrightarrow$ (if and only if).
  - $W_{1,3} \Leftrightarrow W_{2,2}$ is a *biconditional*.

# Grammar of Propositional Logic

John Backus (IBM)

National Medal of Science (1975)
ACM Turing Award (1977)

Peter Naur (U. Copenhagen)

ACM Turing Award (2005)

Backus-Naur form (BNF):

$$\begin{array}{rcl}
\mathit{Sentence} & \rightarrow & \mathit{AtomicSentence} \mid \mathit{ComplexSentence} \\
\mathit{AtomicSentence} & \rightarrow & \mathit{True} \mid \mathit{False} \mid P \mid Q \mid R \mid \ldots \\
\mathit{ComplexSentence} & \rightarrow & (\ \mathit{Sentence}\ ) \\
& \mid & \neg\ \mathit{Sentence} \\
& \mid & \mathit{Sentence} \land \mathit{Sentence} \\
& \mid & \mathit{Sentence} \lor \mathit{Sentence} \\
& \mid & \mathit{Sentence} \ \Rightarrow\ \mathit{Sentence} \\
& \mid & \mathit{Sentence} \ \Leftrightarrow\ \mathit{Sentence} \\
\\
\textsc{Operator Precedence} & : & \neg, \land, \lor, \Rightarrow, \Leftrightarrow
\end{array}$$

* Photos from https://amturing.acm.org/byyear.cfm.

# Grammar of Propositional Logic

John Backus (IBM)

National Medal of Science (1975)
ACM Turing Award (1977)

Peter Naur (U. Copenhagen)

ACM Turing Award (2005)

Backus-Naur form (BNF):

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

$\neg A \vee B \wedge C \Rightarrow D$ is equivalent to

$$((\neg A) \vee (B \wedge C)) \Rightarrow D$$

* Photos from https://amturing.acm.org/byyear.cfm.

# II. 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.

# II. 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

| $P$ | $Q$ | $\neg P$ | $P \wedge Q$ | $P \vee Q$ | $P \Rightarrow Q$ | $P \Leftrightarrow Q$ |
|---|---|---|---|---|---|---|
| *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].

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

# 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].

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

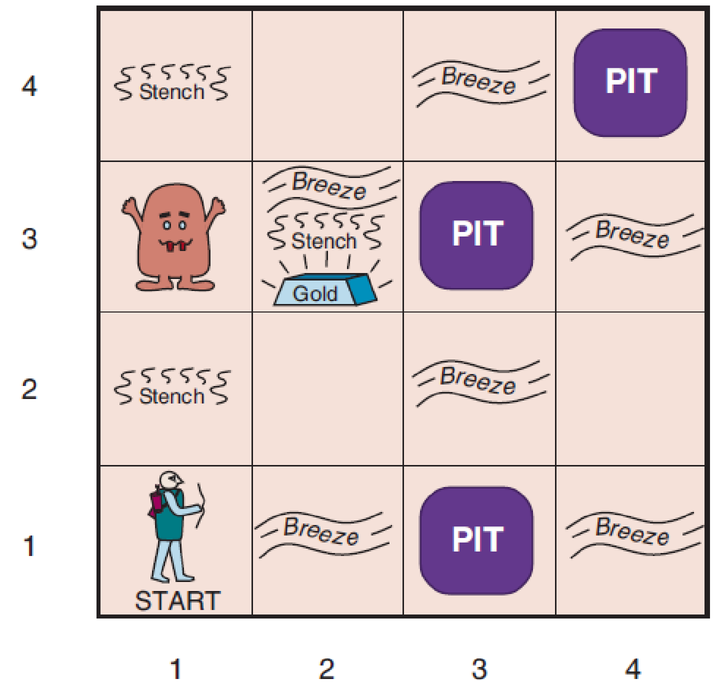
$$= \textit{true} \wedge \textit{true}$$

$$= \textit{true}$$

# III. Knowledge Base for the Wumpus World

Proposition symbols

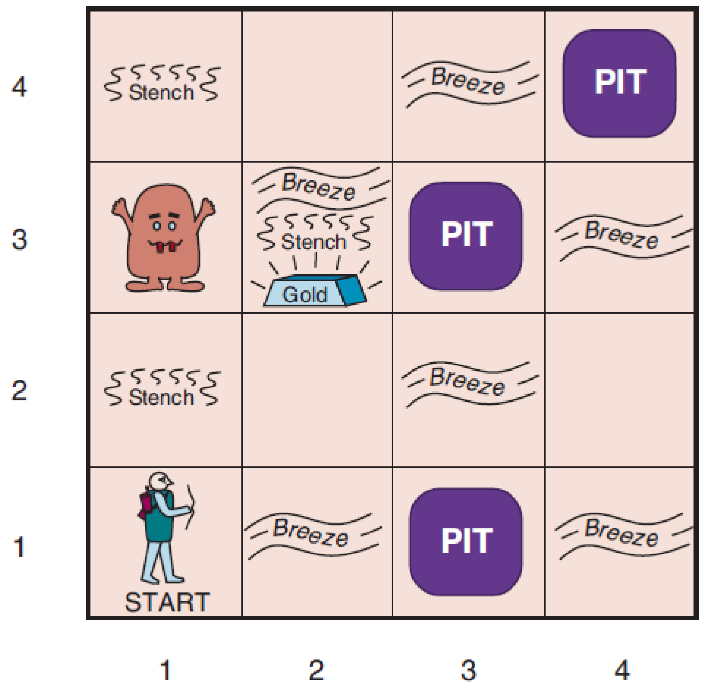
- $P_{x,y}$ is true if there is a pit in $[x, y]$.
- $W_{x,y}$ is true if there is a wumpus in $[x, y]$, dead or alive.
- $B_{x,y}$ is true if the agent perceives a breeze in $[x, y]$.
- $S_{x,y}$ is true if the agent perceives a stench in $[x, y]$.



# 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)

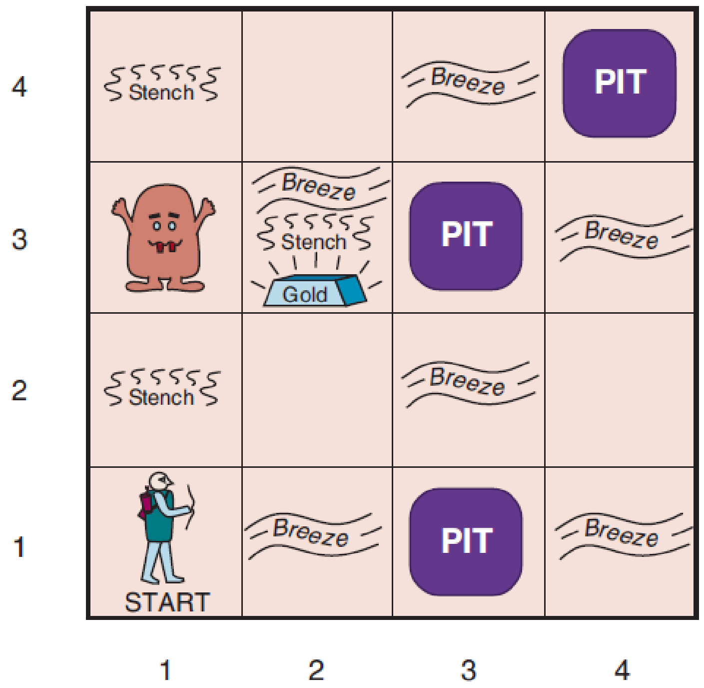
- General knowledge (partial – only for relevant squares):
  - There is no pit in [1,1].

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

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

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

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



# Knowledge Base (cont'd)

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

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

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

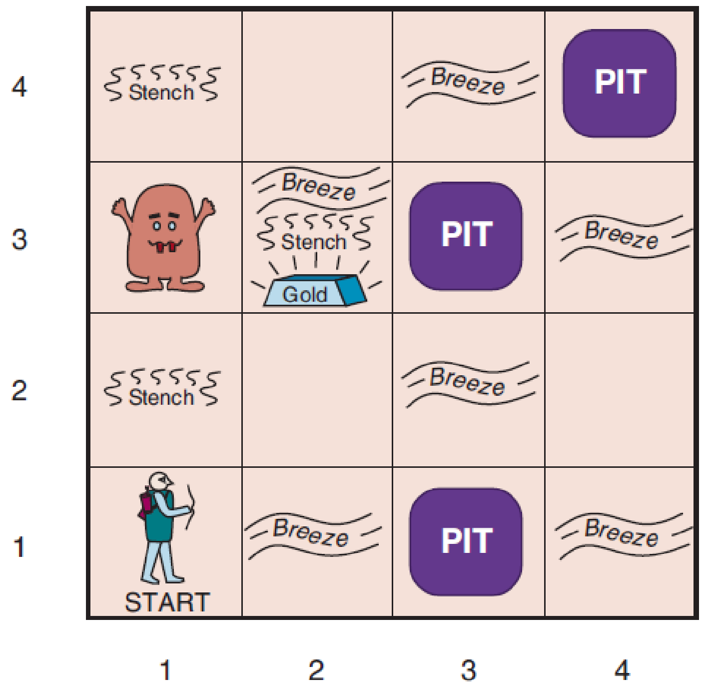
    $R_2\colon\ B_{1,1} \Leftrightarrow (P_{1,2} \lor P_{2,1})$

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

- Percepts for the first two squares:

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

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



# Knowledge Base (cont'd)

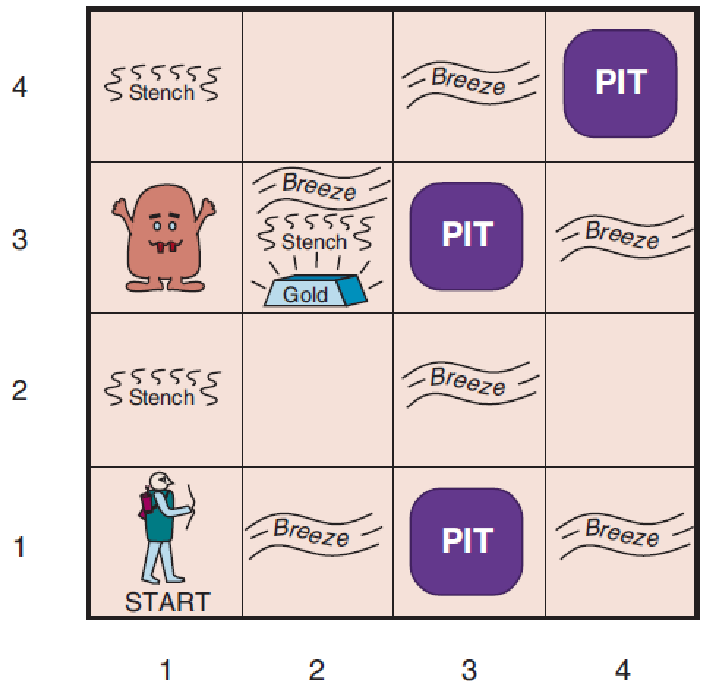
- General knowledge (partial – only for relevant squares):
  - There is no pit in [1,1].

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

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

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

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




- Percepts for the first two squares:

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

  $R_5\colon B_{2,1}$

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

# IV. Inference by Model Checking

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

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

# IV. 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.

# IV. 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.

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

# IV. 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* | <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* |

# 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.

} ⇨ $KB \vDash \neg P_{1,2}$

$P_{2,2}$ 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 $\alpha$ 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).

# V. Logical Equivalence

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

# V. 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 $\alpha$ and $\beta$ are *logically equivalent* if $M(\alpha) = M(\beta)$.

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

# V. 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 $\alpha$ and $\beta$ are *logically equivalent* if $M(\alpha) = M(\beta)$.

$(\alpha \equiv \beta)$

set of models for $\alpha$

# V. 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 $\alpha$ and $\beta$ are *logically equivalent* if $M(\alpha) = M(\beta)$.

$(\alpha \equiv \beta)$

set of models for $\alpha$

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

# V. 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 $\alpha$ and $\beta$ are *logically equivalent* if $M(\alpha) = M(\beta)$.

$(\alpha \equiv \beta)$

set of models for $\alpha$

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

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

# Validity

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

$$P \lor \neg P$$

Valid sentences are *tautologies*.

# Validity

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

$$P \vee \neg P$$

Valid sentences are *tautologies*.

**Deduction theorem** For any sentences $\alpha$ and $\beta$, $\alpha \vDash \beta$ if and only if the sentence $(\alpha \Rightarrow \beta)$ is valid.

# Validity

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

$$P \vee \neg P$$

Valid sentences are *tautologies*.

**Deduction theorem** For any sentences $\alpha$ and $\beta$, $\alpha \vDash \beta$ 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* |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| *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.

# 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* |

$(R_1 \land R_2 \land R_3 \land R_4 \land R_5)$ 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* |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| *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

Proof by contradiction

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

# Inference Rules

## Modus Ponens

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

$\alpha \quad \Rightarrow \quad \beta$

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

**Today is Tuesday**.

---

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

# Inference Rules

## Modus Ponens

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

$\alpha \quad \Rightarrow \quad \beta$

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

**Today is Tuesday**.

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

## And-elimination

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

$\alpha \quad \wedge \quad \beta$

**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

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

# Other Inference Rules

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

$$\neg(\alpha \lor \beta)$$

# 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

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

$$\frac{\neg(\alpha \lor \beta)}{\neg\alpha \land \neg\beta} \quad \text{De Morgan}$$

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

# 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:

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

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

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

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

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

**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

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

and-elimination

# Applying Rules to the Wumpus World

KB:

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

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

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

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

$R_5: B_{2,1}$

**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

and-elimination

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

logical equivalence

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

# 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

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

and-elimination

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

logical equivalence

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

# 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

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

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

# 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

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

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

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

# Applying Rules to the Wumpus World

KB:

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

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

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

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

$R_5: B_{2,1}$

**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

# 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.

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

# Monotonicity

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

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

$\alpha$: There is no pit in $[1, 2]$.

$\beta$: 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.

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

$\alpha$: There is no pit in $[1, 2]$.

$\beta$: There are exactly eight pits in the wumpus world

Knowing $\beta$ might help draw additional conclusions, but it cannot invalidate already inferred $\alpha$.

# Monotonicity

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

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

$\alpha$: There is no pit in $[1, 2]$.

$\beta$: There are exactly eight pits in the wumpus world

Knowing $\beta$ might help draw additional conclusions, but it cannot invalidate already inferred $\alpha$.

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.