

**CS 2710 Foundations of AI**  
**Lecture 9**

**Propositional logic**

**Milos Hauskrecht**

[milos@cs.pitt.edu](mailto:milos@cs.pitt.edu)

5329 Sennott Square

---

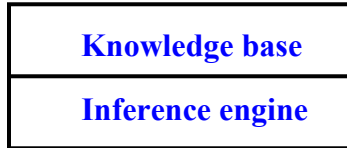
CS 2710 Foundations of AI

**Knowledge representation**

---

CS 2710 Foundations of AI

## Knowledge-based agent



- **Knowledge base (KB):**
  - A set of sentences that describe facts about the world in some formal (representational) language
  - **Domain specific**
- **Inference engine:**
  - A set of procedures that use the representational language to infer new facts from known ones or answer a variety of KB queries. Inferences typically require search.
  - **Domain independent**

---

CS 2710 Foundations of AI

## Example: MYCIN

- MYCIN: an expert system for diagnosis of bacterial infections
- **Knowledge base** represents
  - Facts about a specific patient case
  - Rules describing relations between entities in the bacterial infection domain

<b>If</b>	1. The stain of the organism is gram-positive, and 2. The morphology of the organism is coccus, and 3. The growth conformation of the organism is chains
<b>Then</b>	the identity of the organism is streptococcus

- **Inference engine:**
  - manipulates the facts and known relations to answer diagnostic queries (consistent with findings and rules)

---

CS 2710 Foundations of AI

## Knowledge representation

- The objective of knowledge representation is to express the knowledge about the world in a computer-tractable form
- Key aspects of knowledge representation languages:
  - **Syntax**: describes how sentences are formed in the language
  - **Semantics**: describes the meaning of sentences, what is it the sentence refers to in the real world
  - **Computational aspect**: describes how sentences and objects are manipulated in concordance with semantical conventions

**Many KB systems rely on some variant of logic**

---

CS 2710 Foundations of AI

## Logic

A formal language for expressing knowledge and ways of reasoning.

**Logic** is defined by:

- **A set of sentences**
  - A sentence is constructed from a set of primitives according to syntax rules.
- **A set of interpretations**
  - An interpretation gives a semantic to primitives. It associates primitives with values.
- **The valuation (meaning) function  $V$** 
  - Assigns a value (typically the truth value) to a given sentence under some interpretation

$V : \text{sentence} \times \text{interpretation} \rightarrow \{True, False\}$

---

CS 2710 Foundations of AI

## Example of logic

### Language of numerical constraints:

- **A sentence:**

$$x + 3 \leq z$$

$x, z$  - variable symbols (primitives in the language)

- **An interpretation:**

$$I: x = 5, z = 2$$

Variables mapped to specific real numbers

- **Valuation (meaning) function  $V$ :**

$V(x + 3 \leq z, I)$  is **False** for  $I: x = 5, z = 2$

is **True** for  $I: x = 5, z = 10$

## Types of logic

- Different types of logics possible:
  - Propositional logic
  - First-order logic
  - Temporal logic
  - Numerical constraints logic
  - Map-coloring logic

In the following:

- **Propositional logic.**
  - Formal language for making logical inferences
  - Foundations of **propositional logic: George Boole (1854)**

## Propositional logic. Syntax

- **Propositional logic P:**
  - defines a language for symbolic reasoning
- **Proposition:** a statement that is either true or false
- Examples of propositions:
  - *Pitt is located in the Oakland section of Pittsburgh.*
  - *France is in Europe.*
  - *It rains outside.*
  - *2 is a prime number and 6 is a prime*
  
  - *How are you?* Not a proposition.

---

CS 2710 Foundations of AI

## Propositional logic. Syntax

- **Formally propositional logic P:**
  - Is defined by **Syntax+interpretation+semantics of P**

### Syntax:

- **Symbols (alphabet)** in P:
  - **Constants:** *True, False*
  - **Propositional symbols**
- Examples:
  - *P*
  - *Pitt is located in the Oakland section of Pittsburgh.,*
  - *It rains outside,* etc.
- **A set of connectives:**

$\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$

---

CS 2710 Foundations of AI

## Propositional logic. Syntax

### Sentences in the propositional logic:

- **Atomic sentences:**
  - **Constructed from constants and propositional symbols**
  - True, False are (atomic) sentences
  - $P, Q$  or *Light in the room is on, It rains outside* are (atomic) sentences
- **Composite sentences:**
  - **Constructed from valid sentences via connectives**
  - If  $A, B$  are sentences then
    - $\neg A$   $(A \wedge B)$   $(A \vee B)$   $(A \Rightarrow B)$   $(A \Leftrightarrow B)$
    - or  $(A \vee B) \wedge (A \vee \neg B)$are sentences

## Propositional logic. Semantics.

**The semantic gives the meaning to sentences.**

the semantics in the propositional logic is defined by:

- 1. Interpretation of propositional symbols and constants**
  - Semantics of atomic sentences
- 2. Through the meaning of connectives**
  - Meaning (semantics) of composite sentences

## Semantic: propositional symbols

A **propositional symbol**

- a statement about the world that is either true or false

Examples:

- *Pitt is located in the Oakland section of Pittsburgh*
- *It rains outside*
- *Light in the room is on*

- An **interpretation** maps symbols to one of the two values: **True (T)**, or **False (F)**, depending on whether the symbol is satisfied in the world

**I**: *Light in the room is on* -> **True**, *It rains outside* -> **False**

**I'**: *Light in the room is on* -> **False**, *It rains outside* -> **False**

## Semantic: propositional symbols

The **meaning (value)** of the propositional symbol for a specific interpretation is given by its interpretation

**I**: *Light in the room is on* -> **True**, *It rains outside* -> **False**

$V(\textit{Light in the room is on}, \mathbf{I}) = \textit{True}$

$V(\textit{It rains outside}, \mathbf{I}) = \textit{False}$

**I'**: *Light in the room is on* -> **False**, *It rains outside* -> **False**

$V(\textit{Light in the room is on}, \mathbf{I}') = \textit{False}$

## Semantics: constants

- **The meaning (truth) of constants:**
  - True and False constants are always (under any interpretation) assigned the corresponding *True, False* value

$$\left. \begin{array}{l} V(\mathit{True}, \mathbf{I}) = \mathit{True} \\ V(\mathit{False}, \mathbf{I}) = \mathit{False} \end{array} \right\} \text{For any interpretation } \mathbf{I}$$

## Semantics: composite sentences.

- **The meaning (truth value) of complex propositional sentences.**
  - Determined using the standard rules of logic:

$P$	$Q$	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>

## Translation

### Assume the following sentences:

- It is not sunny this afternoon and it is colder than yesterday.
- We will go swimming only if it is sunny.
- If we do not go swimming then we will take a canoe trip.
- If we take a canoe trip, then we will be home by sunset.

### Denote:

- $p$  = It is sunny this afternoon
- $q$  = it is colder than yesterday
- $r$  = We will go swimming
- $s$  = we will take a canoe trip
- $t$  = We will be home by sunset

## Translation

### Assume the following sentences:

- It is not sunny this afternoon and it is colder than yesterday.  $\neg p \wedge q$
- We will go swimming only if it is sunny.
- If we do not go swimming then we will take a canoe trip.
- If we take a canoe trip, then we will be home by sunset.

### Denote:

- $p$  = It is sunny this afternoon
- $q$  = it is colder than yesterday
- $r$  = We will go swimming
- $s$  = we will take a canoe trip
- $t$  = We will be home by sunset

## Translation

### Assume the following sentences:

- It is not sunny this afternoon and it is colder than yesterday.  $\neg p \wedge q$
- We will go swimming only if it is sunny.  $r \rightarrow p$
- If we do not go swimming then we will take a canoe trip.
- If we take a canoe trip, then we will be home by sunset.

### Denote:

- $p$  = It is sunny this afternoon
- $q$  = it is colder than yesterday
- $r$  = We will go swimming
- $s$  = we will take a canoe trip
- $t$  = We will be home by sunset

## Translation

### Assume the following sentences:

- It is not sunny this afternoon and it is colder than yesterday.  $\neg p \wedge q$
- We will go swimming only if it is sunny.  $r \rightarrow p$
- If we do not go swimming then we will take a canoe trip.  $\neg r \rightarrow s$
- If we take a canoe trip, then we will be home by sunset.

### Denote:

- $p$  = It is sunny this afternoon
- $q$  = it is colder than yesterday
- $r$  = We will go swimming
- $s$  = we will take a canoe trip
- $t$  = We will be home by sunset

## Translation

### Assume the following sentences:

- It is not sunny this afternoon and it is colder than yesterday.  $\neg p \wedge q$
- We will go swimming only if it is sunny.  $r \rightarrow p$
- If we do not go swimming then we will take a canoe trip.  $\neg r \rightarrow s$
- If we take a canoe trip, then we will be home by sunset.  $s \rightarrow t$

### Denote:

- $p$  = It is sunny this afternoon
- $q$  = it is colder than yesterday
- $r$  = We will go swimming
- $s$  = we will take a canoe trip
- $t$  = We will be home by sunset

## Contradiction and Tautology

Some composite sentences may always (under any interpretation) evaluate to a single truth value:

- **Contradiction** (always *False*)

$$P \wedge \neg P$$

- **Tautology** (always *True*)

$$P \vee \neg P$$

$$\left. \begin{array}{l} \neg(P \vee Q) \Leftrightarrow (\neg P \wedge \neg Q) \\ \neg(P \wedge Q) \Leftrightarrow (\neg P \vee \neg Q) \end{array} \right\} \text{DeMorgan's Laws}$$

## Model, validity and satisfiability

- A **model (in logic)**: An interpretation is a model for a set of sentences if it assigns true to each sentence in the set.
- A sentence is **satisfiable** if it has a model;
  - There is at least one interpretation under which the sentence can evaluate to True.
- A sentence is **valid** if it is **True** in all interpretations
  - i.e., if its negation is **not satisfiable** (leads to contradiction)

$P$	$Q$	$P \vee Q$	$(P \vee Q) \wedge \neg Q$	$((P \vee Q) \wedge \neg Q) \Rightarrow P$
<i>True</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>

## Model, validity and satisfiability

- A **model (in logic)**: An interpretation is a model for a set of sentences if it assigns true to each sentence in the set.
- A sentence is **satisfiable** if it has a model;
  - There is at least one interpretation under which the sentence can evaluate to True.
- A sentence is **valid** if it is **True** in all interpretations
  - i.e., if its negation is **not satisfiable** (leads to contradiction)

$P$	$Q$	$P \vee Q$	$(P \vee Q) \wedge \neg Q$	$((P \vee Q) \wedge \neg Q) \Rightarrow P$
<i>True</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>

## Model, validity and satisfiability

- A **model (in logic)**: An interpretation is a model for a set of sentences if it assigns true to each sentence in the set.
- A sentence is **satisfiable** if it has a model;
  - There is at least one interpretation under which the sentence can evaluate to True.
- A sentence is **valid** if it is **True** in all interpretations
  - i.e., if its negation is **not satisfiable** (leads to contradiction)

### Satisfiable sentence

$P$	$Q$	$P \vee Q$	$(P \vee Q) \wedge \neg Q$	$((P \vee Q) \wedge \neg Q) \Rightarrow P$
<i>True</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>

CS 2710 Foundations of AI

## Model, validity and satisfiability

- A **model (in logic)**: An interpretation is a model for a set of sentences if it assigns true to each sentence in the set.
- A sentence is **satisfiable** if it has a model;
  - There is at least one interpretation under which the sentence can evaluate to True.
- A sentence is **valid** if it is **True** in all interpretations
  - i.e., if its negation is **not satisfiable** (leads to contradiction)

### Satisfiable sentence

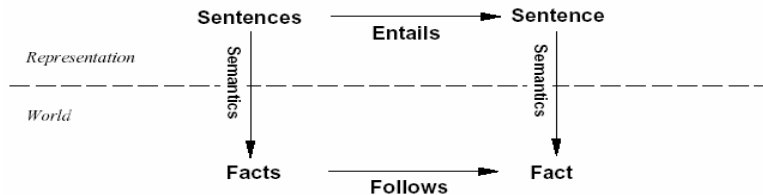
### Valid sentence

$P$	$Q$	$P \vee Q$	$(P \vee Q) \wedge \neg Q$	$((P \vee Q) \wedge \neg Q) \Rightarrow P$
<i>True</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>

CS 2710 Foundations of AI

## Entailment

- **Entailment** reflects the relation of one fact in the world following from the others



- Entailment  $KB \models \alpha$
- Knowledge base KB entails sentence  $\alpha$  if and only if  $\alpha$  is true in all worlds where KB is true

## Sound and complete inference.

**Inference** is a process by which conclusions are reached.

- We want to implement the inference process on a computer !!

Assume an **inference procedure**  $i$  that

- derives a sentence  $\alpha$  from the KB :  $KB \vdash_i \alpha$

### Properties of the inference procedure in terms of entailment

- **Soundness:** An inference procedure is **sound**

If  $KB \vdash_i \alpha$  then it is true that  $KB \models \alpha$

- **Completeness:** An inference procedure is **complete**

If  $KB \models \alpha$  then it is true that  $KB \vdash_i \alpha$

## Logical inference problem

### Logical inference problem:

- **Given:**
  - a knowledge base KB (a set of sentences) and
  - a sentence  $\alpha$  (called **a theorem**),
- **Does a KB semantically entail  $\alpha$ ?**  $KB \models \alpha$  ?

In other words: In all interpretations in which sentences in the KB are true, is also  $\alpha$  true?

**Question:** Is there a procedure (program) that can decide this problem in a finite number of steps?

**Answer:** Yes. Logical inference problem for the propositional logic is **decidable**.

## Solving logical inference problem

In the following:

**How to design the procedure that answers:**

$$KB \models \alpha ?$$

**Three approaches:**

- **Truth-table approach**
- **Inference rules**
- **Conversion to the inverse SAT problem**
  - **Resolution-refutation**

## Truth-table approach

**Problem:**  $KB \models \alpha$  ?

- We need to check all possible interpretations for which the KB is true (models of KB) whether  $\alpha$  is true for each of them

**Truth table:**

- enumerates truth values of sentences for all possible interpretations (assignments of True/False values to propositional symbols)

**Example:**

		KB		$\alpha$
$P$	$Q$	$P \vee Q$	$P \Leftrightarrow Q$	$(P \vee \neg Q) \wedge Q$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>

## Truth-table approach

**Problem:**  $KB \models \alpha$  ?

- We need to check all possible interpretations for which the KB is true (models of KB) whether  $\alpha$  is true for each of them

**Truth table:**

- enumerates truth values of sentences for all possible interpretations (assignments of True/False to propositional symbols)

**Example:**

		KB		$\alpha$
$P$	$Q$	$P \vee Q$	$P \Leftrightarrow Q$	$(P \vee \neg Q) \wedge Q$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>

## Truth-table approach

**Problem:**  $KB \models \alpha$  ?

- We need to check all possible interpretations for which the KB is true (models of KB) whether  $\alpha$  is true for each of them

**Truth table:**

- enumerates truth values of sentences for all possible interpretations (assignments of True/False to propositional symbols)

**Example:**

		KB		$\alpha$
$P$	$Q$	$P \vee Q$	$P \Leftrightarrow Q$	$(P \vee \neg Q) \wedge Q$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>



## Truth-table approach

**A two steps procedure:**

- Generate table for all possible interpretations**
- Check whether the sentence  $\alpha$  evaluates to true whenever  $KB$  evaluates to true

**Example:**  $KB = (A \vee C) \wedge (B \vee \neg C)$       $\alpha = (A \vee B)$

$A$	$B$	$C$	$A \vee C$	$(B \vee \neg C)$	KB	$\alpha$
<i>True</i>	<i>True</i>	<i>True</i>				
<i>True</i>	<i>True</i>	<i>False</i>				
<i>True</i>	<i>False</i>	<i>True</i>				
<i>True</i>	<i>False</i>	<i>False</i>				
<i>False</i>	<i>True</i>	<i>True</i>				
<i>False</i>	<i>True</i>	<i>False</i>				
<i>False</i>	<i>False</i>	<i>True</i>				
<i>False</i>	<i>False</i>	<i>False</i>				

## Truth-table approach

A two steps procedure:

1. Generate table for all possible interpretations
2. Check whether the sentence  $\alpha$  evaluates to true whenever  $KB$  evaluates to true

**Example:**  $KB = (A \vee C) \wedge (B \vee \neg C)$      $\alpha = (A \vee B)$

<i>A</i>	<i>B</i>	<i>C</i>	<i>A</i> $\vee$ <i>C</i>	<i>B</i> $\vee$ $\neg$ <i>C</i>	<i>KB</i>	$\alpha$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>

CS 2710 Foundations of AI

## Truth-table approach

A two steps procedure:

1. Generate table for all possible interpretations
2. Check whether the sentence  $\alpha$  evaluates to true whenever  $KB$  evaluates to true

**Example:**  $KB = (A \vee C) \wedge (B \vee \neg C)$      $\alpha = (A \vee B)$

<i>A</i>	<i>B</i>	<i>C</i>	<i>A</i> $\vee$ <i>C</i>	<i>B</i> $\vee$ $\neg$ <i>C</i>	<i>KB</i>	$\alpha$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>



CS 2710 Foundations of AI

## Truth-table approach

$$KB = (A \vee C) \wedge (B \vee \neg C) \quad \alpha = (A \vee B)$$

<i>A</i>	<i>B</i>	<i>C</i>	$A \vee C$	$(B \vee \neg C)$	<i>KB</i>	$\alpha$
<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>False</i>	<i>True</i>	<i>True</i>	<i>False</i>	<i>False</i>	<i>False</i>
<i>False</i>	<i>False</i>	<i>False</i>	<i>False</i>	<i>True</i>	<i>False</i>	<i>False</i>

KB entails  $\alpha$

- The **truth-table approach** is **sound and complete** for the propositional logic!!

## Limitations of the truth table approach.

$$KB \models \alpha ?$$

What is the computational complexity of the truth table approach?

- ?

## Limitations of the truth table approach.

$$KB \models \alpha ?$$

**What is the computational complexity of the truth table approach?**

**Exponential in the number of the proposition symbols**

$2^n$  Rows in the table has to be filled

## Limitations of the truth table approach.

$$KB \models \alpha ?$$

**What is the computational complexity of the truth table approach?**

**Exponential in the number of the proposition symbols**

$2^n$  Rows in the table has to be filled

**But typically only for a small subset of rows the KB is true**

## Limitations of the truth table approach.

$$KB \models \alpha ?$$

### Problem with the truth table approach:

- the truth table is **exponential** in the number of propositional symbols (we checked all assignments)
- KB is true on only a smaller subset

## Limitation of the truth table approach.

$$KB \models \alpha ?$$

### Problem with the truth table approach:

- the truth table is **exponential** in the number of propositional symbols (we checked all assignments)
- KB is true only on a small subset interpretations

### How to make the process more efficient?

## Inference rules approach.

$$KB \models \alpha ?$$

### Problem with the truth table approach:

- the truth table is **exponential** in the number of propositional symbols (we checked all assignments)
- KB is true on only a smaller subset

### How to make the process more efficient?

**Solution: check only entries for which KB is *True*.**

This is the idea behind the inference rules approach

### Inference rules:

- Represent sound inference patterns repeated in inferences
- Can be used to generate new (sound) sentences from the existing ones

## Inference rules for logic

### • Modus ponens

$$\frac{A \Rightarrow B, A}{B} \quad \begin{array}{l} \leftarrow \text{premise} \\ \leftarrow \text{conclusion} \end{array}$$

- If both sentences in the premise are true then conclusion is true.
- The modus ponens inference rule is **sound**.
  - We can prove this through the truth table.

<i>A</i>	<i>B</i>	<i>A</i> $\Rightarrow$ <i>B</i>
<i>False</i>	<i>False</i>	<i>True</i>
<i>False</i>	<i>True</i>	<i>True</i>
<i>True</i>	<i>False</i>	<i>False</i>
<i>True</i>	<i>True</i>	<i>True</i>

## Inference rules for logic

- **And-elimination**

$$\frac{A_1 \wedge A_2 \wedge \dots \wedge A_n}{A_i}$$

- **And-introduction**

$$\frac{A_1, A_2, \dots, A_n}{A_1 \wedge A_2 \wedge \dots \wedge A_n}$$

- **Or-introduction**

$$\frac{A_i}{A_1 \vee A_2 \vee \dots \vee A_i \vee \dots \vee A_n}$$

---

CS 2710 Foundations of AI

## Inference rules for logic

- **Elimination of double negation**

$$\frac{\neg\neg A}{A}$$

- **Unit resolution**

$$\frac{A \vee B, \neg A}{B}$$

- **Resolution**

$$\frac{A \vee B, \neg B \vee C}{A \vee C}$$

A special  
case of

- All of the above inference rules **are sound**. We can prove this through the truth table, similarly to the **modus ponens** case.

---

CS 2710 Foundations of AI

## Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$

## Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$
4.  $P$

**From 1 and And-elim**

$$\frac{A_1 \wedge A_2 \wedge \dots \wedge A_n}{A_i}$$

## Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$
4.  $P$
5.  $R$

From 2,4 and Modus ponens

$$\frac{A \Rightarrow B, \quad A}{B}$$

## Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$
4.  $P$
5.  $R$
6.  $Q$

From 1 and And-elim

$$\frac{A_1 \wedge A_2 \wedge \dots \wedge A_n}{A_i}$$

### Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$
4.  $P$
5.  $R$
6.  $Q$
7.  $(Q \wedge R)$

From 5,6 and And-introduction

$$\frac{A_1, A_2, A_n}{A_1 \wedge A_2 \wedge A_n}$$

### Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$
4.  $P$
5.  $R$
6.  $Q$
7.  $(Q \wedge R)$
8.  $S$

$$\frac{A \Rightarrow B, A}{B}$$

From 7,3 and Modus ponens

**Proved:  $S$**

## Example. Inference rules approach.

**KB:**  $P \wedge Q$     $P \Rightarrow R$     $(Q \wedge R) \Rightarrow S$    **Theorem:**  $S$

1.  $P \wedge Q$
2.  $P \Rightarrow R$
3.  $(Q \wedge R) \Rightarrow S$
4.  $P$  From 1 and And-elim
5.  $R$  From 2,4 and Modus ponens
6.  $Q$  From 1 and And-elim
7.  $(Q \wedge R)$  From 5,6 and And-introduction
8.  $S$  From 7,3 and Modus ponens

**Proved:**  $S$

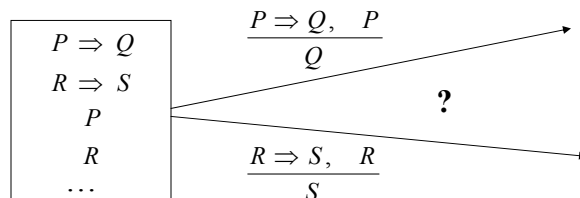
CS 2710 Foundations of AI

## Inference rules

- To show that theorem  $\alpha$  holds for a KB
  - we may need to apply a number of sound inference rules

**Problem:** many possible inference rules to be applied next

**Looks familiar?**



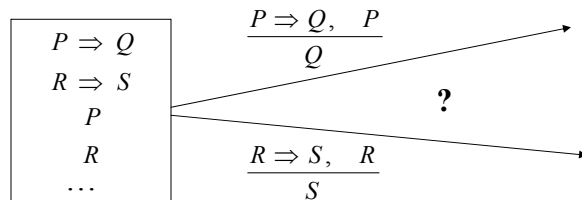
CS 2710 Foundations of AI

## Logic inferences and search

- To show that theorem  $\alpha$  holds for a KB
  - we may need to apply a number of sound inference rules

**Problem:** many possible rules to can be applied next

**Looks familiar?**



**This is an instance of a search problem:**

**Truth table method (from the search perspective):**

- blind enumeration and checking

## Logic inferences and search

**Inference rule method as a search problem:**

- **State:** a set of sentences that are known to be true
- **Initial state:** a set of sentences in the KB
- **Operators:** applications of inference rules
  - Allow us to add new sound sentences to old ones
- **Goal state:** a theorem  $\alpha$  is derived from KB

**Logic inference:**

- **Proof:** A sequence of sentences that are immediate consequences of applied inference rules
- **Theorem proving:** process of finding a proof of theorem