UNIT - III
Artificial Intelligence

III / II IT, R 16 - JNTUK

Ms. M. Rajya Lakshmi

VVIT

Contents

- **Logic concepts:**
  - Propositional calculus
  - Propositional logic
  - Natural deduction system
  - Axiomatic system
  - Semantic tableau system in proportional logic
  - Resolution refutation in proportional logic
  - Predicate logic

Propositional Calculus

- A set of rules are used to combine simple propositions to form compound propositions
- Few logical operators are used, which are known as **connectives**
- E.g. not(~), and(˄), or(˅), implies(->), etc.
- A well formed formula is defined as a symbol or a string of symbols generated by the formal grammar of a formal language

Truth table

- A Truth Table is used to provide operational definitions of important logical operators
- The logical constants in PC are true and false, these are represented as T and F
- The truth values of well formed formulae are calculated by using the truth table approach

Table 4.1 Truth Table for Logical Operators

A	B	~A	A ∧ B	A ∨ B	A → B	A ↔ B
T	T	F	T	T	T	T
T	F	F	F	T	F	F
F	T	T	F	T	T	F
F	F	T	F	F	T	T

Equivalence laws

- Equivalence laws are used to reduce or simplify given well formed formula
- Or to derive a new formula from the existing formula
- These laws can be verified using the truth table approach

Table 4.3 Equivalence Laws

Name of Relation	Equivalence Relations
Commutative Law	$A \vee B \equiv B \vee A$ $A \wedge B \equiv B \wedge A$
Associative Law	$A \vee (B \vee C) \equiv (A \vee B) \vee C$ $A \wedge (B \wedge C) \equiv (A \wedge B) \wedge C$
Double Negation	$\sim(\sim A) \equiv A$
Distributive Laws	$A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$ $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
De Morgan's Laws	$\sim(A \vee B) \equiv \sim A \wedge \sim B$ $\sim(A \wedge B) \equiv \sim A \vee \sim B$

Table 4.3 (Contd.)

Name of Relation	Equivalence Relations
Absorption Laws	$A \vee (A \wedge B) \equiv A$ $A \wedge (A \vee B) \equiv A$ $A \vee (\sim A \wedge B) \equiv A \vee B$ $A \wedge (\sim A \vee B) \equiv A \wedge B$
Idempotence	$A \vee A \equiv A$ $A \wedge A \equiv A$
Excluded Middle Law	$A \vee \sim A \equiv$ T (True)
Contradiction Law	$A \wedge \sim A \equiv F$ (False).
Commonly used equivalence relations	$A \vee F \equiv A$ $A \vee T \equiv T$ $A \wedge T \equiv A$ $A \wedge F \equiv F$ $A \rightarrow B \equiv \sim A \vee B$ $A \leftrightarrow B \equiv (A \rightarrow B) \wedge (B \rightarrow A)$ $\equiv (A \wedge B) \vee (\sim A \wedge \sim B)$

778.JPG

Propositional logic

- It deals with the validity, satisfiability (also known as consistency), and unsatisfiability of a formula and the derivation of a new formula using equivalence laws
- A formula α is said to be tautology iff the value of α is true for all its interpretations
- The validity, satisfiability, and unsatisfiability of a formula may be determined as follows:

- A formula α is said to be valid iff it is tautology
- A formula α is said to be satisfiable if there exists at least one interpretation for which α is true
- A formula α is said to be unsatisfiable if the value of α is false under all interpretations
- E.g. if it is humid then it will rain and since it is humid today it will rain
- Solution:
  – A: It is humid
  – B: It will rain

- The formula α corresponding to the given sentence is α: [(A->B)^A]->B

Table 4.5 Truth Table for $[(A \rightarrow B) \wedge A] \rightarrow B$

A	B	$A \rightarrow B = (X)$	$X \wedge A = (Y)$	$Y \rightarrow B$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	T
F	F	T	F	T

- Truth table approach is easy for evaluating consistency, validity, etc. of a formula
- But the size of truth table grows exponentially
- If a formula contains n atoms, then the truth table will contain 2^n entries
- Some other methods are concerned are:
  – Natural deduction system
  – Axiomatic system
  – Semantic tableau method
  – Resolution refutation method

Natural deduction system

- It mimics the pattern of natural reasoning
- Follows forward chaining
- The system is based on a set of deductive inference rules
- Assuming that $A_1, A_2, \ldots A_k$, where $1 <= k <= n$, are set of atoms and α_j, where $1 <= j <= m$, and β are well formed formula
- The inference rules may be stated as in table

Table 4.6 NDS Rules Table

Rule Name	Symbol	Rule	Description
Introducing $\wedge$	$(I{:}\wedge)$	If $A_1, \ldots, A_n$ then $A_1 \wedge \ldots \wedge A_n$	If $A_1, \ldots, A_n$ are true, then their conjunction $A_1 \wedge \ldots \wedge A_n$ is also *true*.
Eliminating $\wedge$	$(E{:}\wedge)$	If $A_1 \wedge \ldots \wedge A_n$ then $A_i (1 \le i \le n)$	If $A_1 \wedge \ldots \wedge A_n$ is *true*, then any A_i is also *true*.
Introducing $\vee$	$(I{:}\vee)$	If any A_i $(1 \le i \le n)$ then $A_1 \vee \ldots \vee A_n$	If any A_i $(1 \le i \le n)$ is *true*, then $A_1 \vee \ldots \vee A_n$ is also *true*
Eliminating $\vee$	$(E{:}\vee)$	If $A_1 \vee \ldots \vee A_n, A_1 \to A, \ldots, A_n \to A$ then A	If $A_1 \vee \ldots \vee A_n, A_1 \to A, A_2 \to A, \ldots,$ and $A_n \to A$ are *true*, then A is *true*.
Introducing $\to$	$(I:\to)$	If from $\alpha_1, \ldots, \alpha_n$ infer β is proved then $\alpha_1 \wedge \ldots \wedge \alpha_n \to \beta$ is proved	If given that $\alpha_1, \alpha_2, \ldots,$ and α_n are *true* and from these we deduce β then $\alpha_1 \wedge \ldots \wedge \alpha_n \to \beta$ is also *true*.

Table 4.6 (Contd.)

Rule Name	Symbol	Rule	Description
Eliminating $\rightarrow$	$(E: \rightarrow)$	If $A_1 \rightarrow A, A_1$, then A	If $A_1 \rightarrow A$ and A_1 are *true* then A is also *true*. This is called *Modus Ponen* rule.
Introducing $\leftrightarrow$	$(I: \leftrightarrow)$	If $A_1 \rightarrow A_2$, $A_2 \rightarrow A_1$ then $A_1 \leftrightarrow A_2$	If $A_1 \rightarrow A_2$ and $A_2 \rightarrow A_1$ are *true* then $A_1 \leftrightarrow A_2$ is also *true*.
Elimination $\leftrightarrow$	$(E: \leftrightarrow)$	If $A_1 \leftrightarrow A_2$ then $A_1 \rightarrow A_2$, $A_2 \rightarrow A_1$	If $A_1 \leftrightarrow A_2$ is *true* then $A_1 \rightarrow A_2$ and $A_2 \rightarrow A_1$ are *true*
Introducing ~	$(I: \sim)$	*If from A infer* $A_1 \wedge \sim A_1$ is proved then ~A is proved	If from A (which is *true*), a contradiction is proved then truth of ~A is also proved
Eliminating ~	$(E: \sim)$	*If from ~A infer* $A_1 \wedge \sim A_1$ is proved then A is proved	If from ~A, a contradiction is proved then truth of A is also proved

- A theorem in NDS written as, from α1,… α2 infer β leads to the interpretation that β is deduced from a set of hypotheses {α1,… α2} are assumed to be true in a given context
- Therefore the theorem β is also true in the same context
- Thus we can conclude that β is consistent
- A theorem that is written as infer β implies that there are no hypotheses and β is true under all interpretations, i.e. β is a tautology

- E.g. prove that $A \wedge (B \vee C)$ is deduced from $A \wedge B$
- Solution: The theorem in NDS can be written as from $A \wedge B$ infer $A \wedge (B \vee C)$ in NDS

Table 4.7 Proof of the Theorem for Example 4.3

Description	Formula	Comments
Theorem	*from* $A \wedge B$ *infer* $A \wedge (B \vee C)$	*To be proved*
Hypothesis (given)	$A \wedge B$	1
E: $\wedge$ (1)	A	2
E: $\wedge$ (1)	B	3
I: $\vee$ (3)	$B \vee C$	4
I: $\wedge$ (2, 4)	$A \wedge (B \vee C)$	Proved

Axiomatic system

- It is based on a set of three axioms and one rule of deduction
- A forward chaining approach
- It is as powerful as Truth table & NDS approach
- In this, a guess is required in selection of appropriate axiom(s)
- In this, only two logical operators, not(~) and implication(->) are allowed to forma a formula
- Any formula can be written using the ~ and ->

- E.g.
  - A∧B≡ ~(~A∨~B) ≡ ~(A->~B)
  - A∨B ≡ ~A->B
  - A<->B ≡ (A->B)∧(B->A) ≡ ~[(A->B)->~(B->A)]
- In axiomatic system, there are three axioms, which are always true(or valid), and one rule is called Modus Ponen (MP) as follows:
  - Axiom1: A->(B->A)
  - Axiom2: [A->(B->C)]-> [(A->B)->(A->C)]
  - Axiom3: (~A->~B)->(B->A)
  - Rule: Hypothesis: A->B and A; consequent: B

- Interpretation of Modus Ponen’s rule:
  - Given that A->B and A are hypotheses (assumed to be true), B is inferred (i.e., true) as a consequent
- E.g. Establish A->C is a deductive sequence of {A->B, B->C}, i.e. {A->B,B->C} |- (A->C)

Table 4.9 Proof of the theorem $\{A \rightarrow B, B \rightarrow C\} \vdash (A \rightarrow C)$

Description	Formula	Comments
Theorem	$\{A \rightarrow B, B \rightarrow C\} \vdash (A \rightarrow C)$	*Prove*
Hypothesis 1	$A \rightarrow B$	1
Hypothesis 2	$B \rightarrow C$	2
Instance of Axiom 1	$(B \rightarrow C) \rightarrow [A \rightarrow (B \rightarrow C)]$	3
MP (2, 3)	$[A \rightarrow (B \rightarrow C)]$	4
Instance of Axiom 2	$[A \rightarrow (B \rightarrow C)] \rightarrow [(A \rightarrow B) \rightarrow (A \rightarrow C)]$	5
MP (4, 5)	$(A \rightarrow B) \rightarrow (A \rightarrow C)$	6
MP (1, 6)	$(A \rightarrow C)$	*Proved*

Hence, we can conclude that $A \rightarrow C$ is a deductive consequence of $\{A \rightarrow B, B \rightarrow C\}$.

Semantic tableau system

- A backward chaining approach
- A set of rules are applied systematically on a formula/ a set of formulae in order to establish consistency or inconsistency
- Semantic Tableau is a binary tree constructed by using semantic tableau rules with a formula as a root

Semantic Tableau rules

Table 4.11 Semantic Tableau Rules for α and β

Rule No.	Tableau tree	Explanation
Rule 1	*$\alpha \wedge \beta$ is true if both α and β are true* $\alpha \wedge \beta$ | α | β	A tableau for a formula $(\alpha \wedge \beta)$ is constructed by adding both α and β to the same path (branch)
Rule 2	*$\sim(\alpha \wedge \beta)$ is true if either $\sim\alpha$ or $\sim\beta$ is true* $\sim(\alpha \wedge \beta)$ $\sim\alpha$ $\quad$ $\sim\beta$	A tableau for a formula $\sim(\alpha \wedge \beta)$ is constructed by adding two new paths: one containing $\sim\alpha$ and the other containing $\sim\beta$
Rule 3	*$\alpha \vee \beta$ is true if either α or β is true* $\alpha \vee \beta$ α $\quad$ β	A tableau for a formula $(\alpha \vee \beta)$ is constructed by adding two new paths: one containing α and the other containing β

Rule 4	$\sim(\alpha \vee \beta)$ *is true if both* $\sim\alpha$ *and* $\sim\beta$ *are true* $\sim(\alpha \vee \beta)$ | $\sim\alpha$ | $\sim\beta$	A tableau for a formula $\sim(\alpha \vee \beta)$ is constructed by adding both $\sim\alpha$ and $\sim\beta$ to the same path
Rule 5	$\sim(\sim\alpha)$ *is true then* α *is true* $\sim(\sim\alpha)$ | α	A tableau for $\sim(\sim\alpha)$ is constructed by adding α on the same path
Rule 6	$\alpha \rightarrow \beta$ *is true then* $\sim\alpha \vee \beta$ *is true* $\alpha \rightarrow \beta$ / \\ $\sim\alpha$ β	A tableau for a formula $\alpha \rightarrow \beta$ is constructed by adding two new paths: one containing $\sim\alpha$ and the other containing β
Rule 7	$\sim(\alpha \rightarrow \beta)$ *true then* $\alpha \wedge \sim\beta$ *is true* $\sim(\alpha \rightarrow \beta)$ | α | $\sim\beta$	A tableau for a formula $\sim(\alpha \rightarrow \beta)$ is constructed by adding both α and $\sim\beta$ to the same path

Rule No.	Tableau tree	Explanation
Rule 8	$\alpha \leftrightarrow \beta$ *is true then* $(\alpha \wedge \beta) \vee (\sim\alpha \wedge \sim\beta)$ *is true* $\alpha \leftrightarrow \beta$ $\alpha \wedge \beta$ $\quad$ $\sim\alpha \wedge \sim\beta$	A tableau for a formula $\alpha \leftrightarrow \beta$ is constructed by adding two new paths. one containing $\alpha \wedge \beta$ and other $\sim\alpha \wedge \sim\beta$ which are further expanded
Rule 9	$\sim(\alpha \leftrightarrow \beta)$ is *true* then $(\alpha \wedge \sim\beta) \vee (\sim\alpha \wedge \beta)$ *is true* $\sim(\alpha \leftrightarrow \beta)$ $\alpha \wedge \sim\beta$ $\quad$ $\sim\alpha \wedge \beta$	A tableau for a formula $\sim(\alpha \leftrightarrow \beta)$ is constructed by adding two new paths: one containing $\alpha \wedge \sim\beta$ and the other $\sim\alpha \wedge \beta$ which are further expanded

E.g. construct a semantic tableau for a formula (A∧~B)∧ (~B->C)

Table 4.12 Semantic Tableau for Example 4.7

Description	Formula	Line number
Tableau root	$(A \wedge \sim B) \wedge (\sim B \rightarrow C)$ B V C	1
Rule 1 (1)	$A \wedge \sim B$	2
	$\sim B \rightarrow C$	3
Rule 1 (2)	A	4
	$\sim B$	5
Rule 6 (3)	$\sim(\sim B)$ C	6
Rule 3 (6)	B √(open)	
	× (closed) $\{B, \sim B\}$	

IMG_0786.JF

Table 4.14 Tableau Method For Example 4.10

Description	Formula	Line number
Tableau root	$\sim(A \vee B) \wedge (C \rightarrow B) \wedge (A \vee C)$	1
Rule 1 (1)	$\sim(A \vee B)$	2
	$(C \rightarrow B)$	3
	$(A \vee C)$	4
Rule 4 (2)	$\sim A$	
	$\sim B$	
Rule 3 (4)	A ; C	
	$\times \{A, \sim A\}$	
Rule 6 (3)	$\sim C$; B	
	$\times \{C, \sim C\}$; $\times \{B, \sim B\}$	

Table 4.15 Tableau Method for Example 4.11

Description	Formula	Line number
Tableau root	$\sim(A \vee B) \wedge (B \rightarrow C) \wedge (A \vee C)$	1
Rule-1 (1)	$\sim(A \vee B)$	2
	$(B \rightarrow C)$	3
	$(A \vee C)$	4
Rule 4 (2)	$\sim A$	
	$\sim B$	
Rule 3 (4)	A C	
	$\times \{A, \sim A\}$	
Rule 6 (3)	$\sim B$ C	
	$\surd$ $\surd$	

Resolution refutation

- This method is used to derive a goal from a given set of clauses by contradiction
- Clause denotes a special formula containing the Boolean operators ~ and ٧
- This is used to develop computer-based-systems that can be used to prove theorems automatically
- Uses a single inference rule, ‘resolution based on modus ponen inference rule’

Table 4.17 Tableau Method for Example 4.13

Description	Formula	Line number
Tableau root	$\sim(B \vee \sim(A \rightarrow B) \vee \sim A)$	1
Rule 4 (1)	$\sim B$	2
	$\sim[\sim(A \rightarrow B) \vee \sim A]$	3
Rule 4 (3)	$\sim[\sim(A \rightarrow B)]$	4
	$\sim(\sim A)$	5
Rule 5 (5)	A	6
Rule 5 (4)	$A \rightarrow B$	
Rule 6 (7)	$\sim A \qquad B$	7
	$\times\{A, \sim A\} \qquad \times\{B, \sim B\}$	

Predicate logic

- Propositional logic has many limitations
- E.g. John is a boy, Paul is a boy, Peter is a boy
  - It can be symbolized by A,B, & C respectively in propositional logic
  - But no conclusions can be drawn similarities among boys, i.e. A, B, C cannot represent boys
- So, a general statement boy(X), where X is bound with John, Paul, Peter
- These facts are called instances of a boy(X)
- Here, boy(X) is called a **Predicate statement** or **expression,** boy is **predicate symbol,** X is **argument**

- When a variable X gets bound to its actual value then the predicate statement boy(X) becomes either true or false
- E.g. boy(Peter) = true, boy(Marry) = false
- Predicate logic is a logical extension of Propositional logic
- Predicate calculus is the study of predicate systems
- When inference rules are added to predicate calculus, it becomes predicate logic