Chapter 3
Sentential Logic: Semantics

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

- We shall develop formal tests for *truth-functional* versions of logical concepts:

i) Truth-functional truth, falsity, and indeterminacy

ii) Truth-functional equivalence

iii) Truth-functional consistency

iv) Truth-functional entailment

v) Truth-functional validity

- *Recall*: Every sentence of SL can be built up from its atomic components.

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

- The truth-value of a sentence of SL is determined completely by the truth-values of its atomic components in accordance with the truth tables for the connectives:

P	Q	~P	P & Q	P ∨ Q	P ⊃ Q	P ≡ Q
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

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

- The truth-values of atomic sentences are fixed by truth-value assignments, i.e., by an assignment of a truth-value (T, F)
- If ***P*** has *n* atomic components, then there are 2^n different combinations of truth-values for its atomic components.
- A truth-table with *n* sentence letters, will have 2^n rows.

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

- For a truth table with *n* sentence letters, each column consists of $2^{n\text{-}i}$ T' s alternating with $2^{n\text{-}i}$ F' s.
- E.g., '~B ⊃ C': two atomic components, 2^2 rows:

B	C
T	T
T	F
F	T
F	F

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

'$(\sim B \supset C) \& (A \equiv B)$' has three atomic components, so 2^3 rows:

A	B	C	$(\sim B \supset C) \& (A \equiv B)$
T	T	T	
T	T	F	
T	F	T	
T	F	F	
F	T	T	
F	T	F	
F	F	T	
F	F	F	

Rule of halves

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

E.g., ‘~[(U ∨ (W ⊃ ~U)) ≡ W]’

U	W	~	[(U	∨	(W	⊃	~	U))	≡	W]
T	T	F	T	T	T	F	F	T	T	T
T	F	T	T	T	F	T	F	T	F	F
F	T	F	F	T	T	T	T	F	T	T
F	F	T	F	T	F	T	T	F	F	F

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

- Sometimes we are not interested in determining the truth-values of a sentence ***P*** for every truth-value assignment, but only in the truth-value of ***P*** on some particular truth-value assignment.

E.g., '(A & B) ⊃ B', where 'A' is false and 'B' is true:

A	B	(A & B) ⊃ B
F	T	F F T T T

3.1 – Truth-Value Assignments and Truth-Tables for Sentences

- *Defn*: a sentence is *true on a truth-value assignment iff* it has the truth-value T on the truth-value assignment
- *Defn*: A sentence is *false on a truth-value assignment iff* it has the truth-value F on the truth-value assignment.

E.g., ‘(A ∨ B) ⊃ B’, where ‘A’ is true and ‘B’ is false:

A	B	(A ∨ B) ⊃ B
T	F	T T F F F

3.2 – Truth-Functional Truth, Falsity, and Indeterminacy

- *Defn*: a sentence ***P*** of SL is *truth-functionally true iff* ***P*** is true on *every* truth-value assignment.

E.g., ‘C ∨ ~C’

C	C	∨	~	C
T	T	T	F	T
F	F	T	T	F

3.2 – Truth-Functional Truth, Falsity, and Indeterminacy

- *Defn*: a sentence of ***P*** of SL is *truth-functionally false iff* ***P*** is false on *every* truth-value assignment.

E.g., ‘P & ~P’

P	P	&	~	P
T	T	F	F	T
F	F	F	T	F

3.2 – Truth-Functional Truth, Falsity, and Indeterminacy

- Negate a truth-functionally true sentence and the result is truth-functionally false, and vice versa for a truth-functionally false sentence.

E.g., ‘P v ~P’

P	~(P v ~P)
T	F T T F T
F	F F T T F

‘~ ~(P v ~P)’ is truth-functionally true

3.2 – Truth-Functional Truth, Falsity, and Indeterminacy

- *Defn*: a sentence ***P*** is *truth-functionally indeterminate iff* ***P*** is neither truth-functionally true nor truth-functionally false.
- Each atomic sentence of SL is truth-functionally indeterminate.

3.2 – Truth-Functional Truth, Falsity, and Indeterminacy

- '$A \supset (B \supset A)$' is truth-functionally true
- '$(A \supset B) \supset A$' is truth-functionally indeterminate

-

A	B	A ⊃ (B ⊃ A)
T	T	T T T T T
T	F	T T F T T
F	T	F T T F F
F	F	F T F T F

A	B	(A ⊃ B) ⊃ A
T	T	T T T T T
T	F	T F F T T
F	T	F T T F F
F	F	F T F F F

3.2 – Truth-Functional Truth, Falsity, and Indeterminacy

- We can show that a sentence is *not* truth-functionally (t-f) true (or false) by showing that there is at least one row which yields the truth-value false (or true).

E.g.,

A	(A & ~A) ∨ ~A
T	T F F T F F T

This truth table shows that the sentence is not t-f true.

E.g.,

A	(A & ~A) ∨ ~A
F	F F T F T T F

This truth table shows that the sentence is not t-f false.

3.3 – Truth-Functional Equivalence

- *Defn*: Sentences ***P*** and ***Q*** are *truth-functionally equivalent iff* there is *no* truth-value assignment on which ***P*** and ***Q*** have different truth-values, i.e., *iff* the columns under the main connective are identical.

E.g., ‘(W & Y) ⊃ H’ and ‘W ⊃ (Y ⊃ H)’

3.3 – Truth-Functional Equivalence

W	Y	H	(W & Y) ⊃ H	W ⊃ (Y ⊃ H)
T	T	T	T	T
T	T	F	F	F
T	F	T	T	T
T	F	F	T	T
F	T	T	T	T
F	T	F	T	T
F	F	T	T	T
F	F	F	T	T

3.3 – Truth-Functional Equivalence

- Any two truth-functionally true (or false) sentences are truth-functionally equivalent.

E.g., ‘(P ∨ ~P)’ and ‘~(B & ~B)’

P	B	P ∨ ~P	~(B & ~B)
T	T	T	T
T	F	T	T
F	T	T	T
F	F	T	T

3.4 – Truth-Functional Consistency

- We denote a set of sentences Γ as $\{S_1 \ldots S_k\}$, where S_i are the members, e.g., $\Gamma = \{A, B \supset H, C \vee A\}$
- *Defn*: a set of sentences of SL is *truth-functionally consistent iff* there is *at least one* truth-value assignment on which all the members of the set are true.
- *Defn*: A set of sentences of SL is *truth-functionally inconsistent iff* it is not truth-functionally consistent
  - i.e., iff there is *no* truth-value assignment on which all the members of the set are true.

3.4 – Truth-Functional Consistency

E.g., {A, B ⊃ H, B}

A	B	H	A	B ⊃ H	B	
T	T	T	T	T	T	← can stop here

E.g., {L, L ⊃ J, ~J}

L	J	L	L ⊃ J	~J
T	T	T	T	F
T	F	T	F	T
F	T	F	T	F
F	F	F	T	T

This set is truth-functionally inconsistent.

3.5 – Truth-Functional Entailment and Truth-Functional Validity

- *Defn*: a set Γ of sentences of SL *truth-functionally entails* a sentence ***P*** *iff* there is *no* truth-value assignment on which every member of Γ is true and ***P*** is false, i.e., ***P*** is true on every truth-value assignment where every member of Γ is true.

- We denote Γ truth-functionally entails ***P*** by $\Gamma \models P$.

3.5 – Truth-Functional Entailment and Truth-Functional Validity

E.g., {A, A & B} $\models$ B:

A	B	A	A & B	B
T	T	T	T	T
T	F	T	F	F
F	T	F	F	T
F	F	F	F	F

3.5 – Truth-Functional Entailment and Truth-Functional Validity

E.g., ‘{A, A ∨ B}’ does not entail ‘B’:

A	B	A	A ∨ B	B	
T	T	T	T	T	
T	F	T	T	F	←
F	T	F	T	T	
F	F	F	F	F	

3.5 – Truth-Functional Entailment and Truth-Functional Validity

- If there is *one* truth-value assignment on which every member of Γ is true but P is false, then Γ does not entail P, even if there is an assignment on which every member of Γ is true and P is true.

- The expression '$\models$ ***P***' is an abbreviation for '$\varnothing \models$ ***P***'

- The empty set entails all and only truth-functionally true sentences. (Since there is no assignment to ∅, there is no assignment on which Γ is true and ***P*** is false.)

3.5 – Truth-Functional Entailment and Truth-Functional Validity

- *Defn*: An argument of SL is *truth-functionally valid* iff there is *no* truth value assignment on which the premises are true and the conclusion is false.
- *Defn*: An argument of SL is truth-functionally invalid iff it is not truth-functionally valid.
- An *argument* is truth-functionally valid iff the *set* composed of the premises truth-functionally entails the conclusion.

3.5 – Truth-Functional Entailment and Truth-Functional Validity

- For any argument of SL that has a finite number of premises, we may look for a sentence called the corresponding *material conditional*:

P_1

P_2

$\underline{P_3}$

C

$$((P_1 \& P_2) \& P_3) \supset C$$

- The argument is truth-functionally valid *iff* the corresponding material conditional is truth-functionally true.

3.5 – Truth-Functional Entailment and Truth-Functional Validity

A

A ⊃ B

B

is truth-functionally valid iff '[A & (A ⊃ B)] ⊃ B' is truth-functionally true

A	B	[A & (A ⊃ B)] ⊃ B
T	T	T T T T T T T
T	F	T F T F F T F
F	T	F F F T T T T
F	F	F F F T F T F

3.6 – Truth-Functional Properties and Truth-Functional Consistency

- Recasting the truth-functional notions in terms of *truth-functional consistency*:

  i) A sentence ***P*** is *truth-functionally false iff* {***P***} is truth-functionally inconsistent.

  ii) A sentence ***P*** is *truth-functionally true iff* {~***P***} is truth-functionally inconsistent.

  iii) A sentence ***P*** is *truth-functionally indeterminate iff* both {~***P***} and {***P***} are truth-functionally consistent.

3.6 – Truth-Functional Properties and Truth-Functional Consistency

- *Proof*: A sentence ***P*** is *truth-functionally false* iff {***P***} is truth-functionally inconsistent.

  Suppose ***P*** is truth-functionally false.

  No truth-value assignment on which ***P*** is true (*defn*).

  No truth-value assignment on which every member of {***P***} is true. (***P*** is the only member of {***P***}.)

  Hence, {***P***} is truth-functionally inconsistent.

3.6 – Truth-Functional Properties and Truth-Functional Consistency

- *Proof*: A sentence ***P*** is *truth-functionally true* iff {~***P***} is truth-functionally inconsistent.

Suppose ***P*** is truth-functionally true.

P is true on every truth-value assignment (*defn*).

~***P*** is true on no truth-value assignment. (~***P*** is false on every truth-value assignment.)

No truth-value assignment on which every member of {~***P***} is true. (~***P*** is the only member of {~***P***}.)

Hence, {~***P***} is truth-functionally inconsistent.

3.6 – Truth-Functional Properties and Truth-Functional Consistency

- ***P*** and ***Q*** are *truth-functionally equivalent iff* $\boldsymbol{P} \equiv \boldsymbol{Q}$ is truth-functionally true
- $\boldsymbol{P} \equiv \boldsymbol{Q}$ is *truth-functionally true iff* $\{{\sim}(\boldsymbol{P} \equiv \boldsymbol{Q})\}$ is truth-functionally inconsistent
- ***P*** and ***Q*** are *truth-functionally equivalent iff* $\{{\sim}(\boldsymbol{P} \equiv \boldsymbol{Q})\}$ is truth-functionally inconsistent.

3.6 – Truth-Functional Properties and Truth-Functional Consistency

A	B	$\sim[(A \vee B) \equiv (\sim A \supset B)]$	
T	T	F	T
T	F	F	T
F	T	F	T
F	F	F	T

$\{\sim[(A \vee B) \equiv (\sim A \supset B)]\}$ is truth-functionally inconsistent.

Hence, ‘$(A \vee B)$’ and ‘$(\sim A \supset B)$’ are truth-functionally equivalent.

3.6 – Truth-Functional Properties and Truth-Functional Consistency

- If $\Gamma \vDash$ ***P***, for some sentence ***P*** and set of sentences Γ, then $\Gamma \cup$ {~***P***} is truth-functionally inconsistent.

- $\Gamma \vDash$ ***P***, that is, Γ *truth-functionally entails* ***P*** *iff* $\Gamma \cup$ {~***P***} is truth-functionally inconsistent.

3.6 – Truth-Functional Properties and Truth-Functional Consistency

An argument P_1

P_2

$\underline{P_k}$

C

is *truth-functionally valid iff* $\{P_1 \ldots P_k\} \cup \{\sim C\}$ is truth-functionally inconsistent.

E.g., $(A \supset D) \,\&\, H$

$\underline{F \vee H}$

D

is valid *iff* $\{(A \supset D) \,\&\, H, F \vee H, \sim D\}$ is t-f inconsistent.