Formal Methods in Software Engineering

Propositional Logic — Spring 2025

Konstantin Chukharev

§1 Propositional Logic

Motivation

- Boolean functions are at the core of logic-based reasoning.
- A Boolean function $F(X_1, ..., X_n)$ describes the output of a system based on its inputs.
- Boolean gates (AND, OR, NOT) form the building blocks of digital circuits.
- Propositional logic formalizes reasoning about Boolean functions and circuits.
- **Applications**:
  - Digital circuit design.
  - Verification and synthesis of hardware and software.
  - Expressing logical constraints in AI and optimization problems.
  - Automated reasoning and theorem proving.

Boolean Circuits and Propositional Logic

Boolean circuit is a directed acyclic graph (DAG) of Boolean gates.

- Inputs: Propositional variables.
- Outputs: Logical expressions describing the circuit's behavior.

"Can the output of a circuit ever be true?"

- Propositional logic provides a formal framework to answer such questions.

Real-world examples:

- Error detection circuits.
- Arithmetic logic units (ALUs) in processors.
- Routing logic in network devices.

What is Logic?

A formal logic is defined by its **syntax** and **semantics**.

□ Syntax

- An **alphabet** Σ is a set of symbols.
- A finite sequence of symbols (from Σ) is called an **expression** or **string** (over Σ).
- A set of rules defines the **well-formed** expressions.

□ Semantics

- Gives meaning to (well-formed) expressions.

Syntax of Propositional Logic

□ Alphabet

1. Logical connectives: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$.
2. Propositional variables: $A_1, A_2, \ldots, A_n$.
3. Parentheses for grouping: (,).

□ Well-Formed Formulas (WFFs)

Valid (**well-formed**) expressions are defined **inductively**:

1. A single propositional symbol (e.g. A) is a WFF.
2. If α and β are WFFs, so are: $\neg\alpha,\ (\alpha \wedge \beta),\ (\alpha \vee \beta),\ (\alpha \rightarrow \beta),\ (\alpha \leftrightarrow \beta)$.
3. No other expressions are WFFs.

Syntax of Propositional Logic [2]

□ Conventions

- Large variety of propositional variables: $A, B, C, \ldots, p, q, r, \ldots$.
- Outer parentheses can be omitted: $A \wedge B$ instead of $(A \wedge B)$.
- Operator precedence: $\neg > \wedge > \vee > \rightarrow > \leftrightarrow$.
- Left-to-right associativity for $\wedge$ and $\vee$: $\quad A \wedge B \wedge C = (A \wedge B) \wedge C$.
- Right-to-left associativity for $\rightarrow$: $\quad A \rightarrow B \rightarrow C = A \rightarrow (B \rightarrow C)$.

Semantics of Propositional Logic

- Each propositional variable is assigned a truth value: T (true) or F (false).
- More formally, *interpretation* $\nu : V \to \{0, 1\}$ assigns truth values to all variables (atoms).
- Truth values of complex formulas are computed (evaluated) recursively:
  1. $[\![p]\!]_\nu \triangleq \nu(p)$, where $p \in V$ is a propositional variable
  2. $[\![\neg\alpha]\!]_\nu \triangleq 1 - [\![\alpha]\!]_\nu$
  3. $[\![\alpha \wedge \beta]\!]_\nu \triangleq \min([\![\alpha]\!]_\nu, [\![\beta]\!]_\nu)$
  4. $[\![\alpha \vee \beta]\!]_\nu \triangleq \max([\![\alpha]\!]_\nu, [\![\beta]\!]_\nu)$
  5. $[\![\alpha \to \beta]\!]_\nu \triangleq ([\![\alpha]\!]_\nu \leq [\![\beta]\!]_\nu) = \max(1 - [\![\alpha]\!]_\nu, [\![\beta]\!]_\nu)$
  6. $[\![\alpha \leftrightarrow \beta]\!]_\nu \triangleq ([\![\alpha]\!]_\nu = [\![\beta]\!]_\nu) = 1 - |[\![\alpha]\!]_\nu - [\![\beta]\!]_\nu|$

§2 Foundations

Truth Tables

α	β	γ	$\alpha \wedge (\beta \vee \neg\gamma)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Normal Forms

- **Conjunctive Normal Form (CNF)**:
  - A formula is in CNF if it is a conjunction of *clauses* (disjunctions of literals).

  Example: $(A \lor B) \land (\neg A \lor C) \land (B \lor \neg C)$ — CNF with 3 clauses.

- **Disjunctive Normal Form (DNF)**:
  - A formula is in DNF if it is a disjunction of *cubes* (conjunctions of literals).

  Example: $(\neg A \land B) \lor (B \land C) \lor (\neg A \land B \land \neg C)$ — DNF with 3 cubes.

- **Algebraic Normal Form (ANF)**:
  - A formula is in ANF if it is a sum of *products* of variables (or a constant 1).

  Example: $B \oplus AB \oplus ABC$ — ANF with 3 terms.

Logical Laws and Tautologies

- **Associative** and **Commutative** laws for $\wedge$, $\vee$, $\leftrightarrow$:
  - $A \circ (B \circ C) \equiv (A \circ B) \circ C$
  - $A \circ B \equiv B \circ A$
- **Distributive laws**:
  - $A \wedge (B \vee C) \equiv (A \wedge B) \vee (A \wedge C)$
  - $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$
- **Negation**:
  - $\neg\neg A \equiv A$
- **De Morgan's laws**:
  - $\neg(A \wedge B) \equiv \neg A \vee \neg B$
  - $\neg(A \vee B) \equiv \neg A \wedge \neg B$

Logical Laws and Tautologies [2]

- **Implication**:
  - $(A \to B) \equiv (\neg A \lor B)$
- **Contraposition**:
  - $(A \to B) \equiv (\neg B \to \neg A)$
- **Law of Excluded Middle**:
  - $(A \lor \neg A) \equiv \top$
- **Contradiction**:
  - $(A \land \neg A) \equiv \bot$
- **Exportation**:
  - $((A \land B) \to C) \equiv (A \to (B \to C))$

Completeness of Connectives

- All Boolean functions can be expressed using $\{\neg, \wedge, \vee\}$ (so called *"standard Boolean basis"*).
- Even smaller sets are sufficient:
  - $\{\neg, \wedge\}$ — AIG (And-Inverter Graph), see also: AIGER format.
  - $\{\neg, \vee\}$
  - $\{\overline{\wedge}\}$ — NAND
  - $\{\overline{\vee}\}$ — NOR

Incompleteness of Connectives

To prove that a set of connectives is incomplete, we find a property that is true for all WFFs expressed using those connectives, but that is not true for some Boolean function.

Example: $\{\land, \to\}$ is not complete.

Proof: Let α be a WFF which uses only these connectives. Let ν be an interpretation such that $\nu(A_i) = 1$ for all propositional variables A_i. Next, we prove by induction that $[\![\alpha]\!]_\nu = 1$.

- Base case:
  - $[\![A_i]\!]_\nu = \nu(A_i) = 1$
- Inductive step:
  - $[\![\beta \land \gamma]\!]_\nu = \min([\![\beta]\!]_\nu, [\![\gamma]\!]_\nu) = 1$
  - $[\![\beta \to \gamma]\!]_\nu = \max(1 - [\![\beta]\!]_\nu, [\![\gamma]\!]_\nu) = 1$

Thus, $[\![\alpha]\!]_\nu = 1$ for all WFFs α built from $\{\land, \to\}$. However, $[\![\neg A_1]\!]_\nu = 0$, so there is no such formula α tautologically equivalent to $\neg A_1$. $\square$

§3 Semantical Aspects

Validity, Satisfiability, Entailment

□ Validity

- α is a **tautology** if α is true under all truth assignments.
  Formally, α is **valid**, denoted "$\vDash \alpha$", iff $[\![\alpha]\!]_\nu = 1$ for all interpretations $\nu \in \{0,1\}^V$.
- α is a **contradiction** if α is false under all truth assignments.
  Formally, α is **unsatisfiable** if $[\![\alpha]\!]_\nu = 0$ for all interpretations $\nu \in \{0,1\}^V$.

□ Satisfiability

- α is **satisfiable** (**consistent**) if there exists an interpretation $\nu \in \{0,1\}^V$ where $[\![\alpha]\!]_\nu = 1$.
  When α is satisfiable by ν, denoted $\nu \vDash \alpha$, this interpretation is called a **model** of α.
- α is **falsifiable** (**invalid**) if there exists an interpretation $\nu \in \{0,1\}^V$ where $[\![\alpha]\!]_\nu = 0$.

□ Entailment

- Let Γ be a set of WFFs. Then Γ **tautologically implies** (**semantically entails**) α, denoted $\Gamma \vDash \alpha$, if every truth assignment that satisfies all formulas in Γ also satisfies α.
- Formally, $\Gamma \vDash \alpha$ iff for all interpretations $\nu \in \{0,1\}^V$ and formulas $\beta \in \Gamma$, if $\nu \vDash \beta$, then $\nu \vDash \alpha$.
- Note: $\alpha \vDash \beta$, where α and β are WFFs, is just a shorthand for $\{\alpha\} \vDash \beta$.

Implication vs Entailment

The **implication** operator ($\rightarrow$) is a syntactic construct, while **entailment** ($\vDash$) is a semantical relation.

They are related as follows: $\alpha \rightarrow \beta$ is valid iff $\alpha \vDash \beta$.

Example: $A \rightarrow (A \lor B)$ is valid (a tautology), and $A \vDash A \lor B$

A	B	$A \lor B$	$A \rightarrow (A \lor B)$	$A \vDash A \lor B$
0	0	0	1	—
0	1	1	1	—
1	0	1	1	OK
1	1	1	1	OK

Examples

- $A \lor B \land (\neg A \land \neg B)$ is satisfiable, but not valid.
- $A \lor B \land (\neg A \land \neg B) \land (A \leftrightarrow B)$ is unsatisfiable.
- $\{A \to B, A\} \vDash B$
- $\{A, \neg A\} \vDash A \land \neg A$
- $\neg(A \land B)$ is tautologically equivalent to $\neg A \lor \neg B$.

Duality of SAT vs VALID

- **SAT**: Given a formula α, determine if it is satisfiable.

$$\exists \nu . [\![\alpha]\!]_\nu$$

- **VALID**: Given a formula α, determine if it is valid.

$$\forall \nu . [\![\alpha]\!]_\nu$$

- **Duality**: α is valid iff $\neg\alpha$ is unsatisfiable.
- Note: SAT is NP, but VALID is co-NP.

Solving SAT using Truth Tables

Algorithm for satisfiability:

To check whether α is satisfiable, construct a truth table for α. If there is a row where α evaluates to true, then α is satisfiable. Otherwise, α is unsatisfiable.

Algorithm for semantical entailment (tautological implication):

The check whether $\{\alpha_1, \dots, \alpha_k\} \vDash \beta$, check the satisfiability of $(\alpha_1 \land \dots \land \alpha_k) \land (\neg\beta)$. If it is unsatisfiable, then $\{\alpha_1, \dots, \alpha_k\} \vDash \beta$. Otherwise, $\{\alpha_1, \dots, \alpha_k\} \nvDash \beta$.

Compactness

Recall:

- A WFF α is **satisfiable** if there exists an interpretation ν such that $\nu \vDash \alpha$.
- Hereinafter, let Γ denote a *finite* set of WFFs, and Σ denote a *possibly infinite* set of WFFs.
- A set of WFFs Σ is **satisfiable** if there exists an interpretation ν that satisfies all formulas in Σ.
- A set of WFFs Σ is **finitely satisfiable** if every finite subset of Σ is satisfiable.

Theorem 1 (Compactness Theorem): A set of WFFs Σ is satisfiable iff it is finitely satisfiable.

Proof ($\Rightarrow$): Suppose Σ is satisfiable, i.e. there exists an interpretation ν that satisfies all formulas in Σ.

This direction is trivial: any subset of a satisfiable set is clearly satisfiable.

- For each finite subset $\Sigma' \subseteq \Sigma$, ν also satisfies all formulas in Σ'.
- Thus, every finite subset of Σ is satisfiable.

$\square$

Compactness [2]

Proof (⇐): Suppose Σ is finitely satisfiable, i.e. every finite subset of Σ is satisfiable.

Construct a *maximal* finitely satisfiable set Δ as follows:

- Let $\alpha_1, \ldots, \alpha_n, \ldots$ be a fixed enumeration of all WFFs.
  - *This is possible since the set of all sequences of a countable set is countable.*
- Then, let:

$$\Delta_0 = \Sigma,$$

$$\Delta_{n+1} = \begin{cases} \Delta_n \cup \{\alpha_{n+1}\} & \text{if this is finitely satisfiable,} \\ \Delta_n \cup \{\neg\alpha_{n+1}\} & \text{otherwise.} \end{cases}$$

  - *Note that each Δ_n is finitely satisfiable by construction.*

Compactness [3]

- Let $\Delta = \bigcup_{n \in \mathbb{N}} \Delta_n$. Note:
  1. $\Sigma \subseteq \Delta$
  2. $\alpha \in \Delta$ or $\neg\alpha \in \Delta$ for any WFF α
  3. Δ is finitely satisfiable by construction.

Now we need to show that Δ is satisfiable (and thus $\Sigma \subseteq \Delta$ is also satisfiable).

Define an interpretation ν as follows: for each propositional variable p, let $\nu(p) = 1$ iff $p \in \Delta$.

We claim that $\nu \vDash \alpha$ iff $\alpha \in \Delta$. The proof is by induction on well-formed formulas.

- Base case:
  - Suppose $\alpha \equiv p$ for some propositional variable p.
  - By definition, $[\![p]\!]_\nu = \nu(p) = 1$.
- Inductive step:
  - *(Note: we consider only two cases: $\neg$ and $\wedge$, since they form a complete set of connectives.)*
  - Suppose $\alpha \equiv \neg\beta$.
    - $[\![\alpha]\!]_\nu = 1$ iff $[\![\beta]\!]_\nu = 0$ iff $\beta \notin \Delta$ iff $\neg\beta \in \Delta$ iff $\alpha \in \Delta$.

Compactness [4]

- Suppose $\alpha \equiv \beta \wedge \gamma$.
  - $[\![\alpha]\!]_\nu = 1$ iff both $[\![\beta]\!]_\nu = 1$ and $[\![\gamma]\!]_\nu = 1$ iff both $\beta \in \Delta$ and $\gamma \in \Delta$.
  - If both β and γ are in Δ, then $\beta \wedge \gamma$ is in Δ, thus $\alpha \in \Delta$.
    - Why? Because if $\beta \wedge \gamma \notin \Delta$, then $\neg(\beta \wedge \gamma) \in \Delta$. But then $\{\beta, \gamma, \neg(\beta \wedge \gamma)\}$ is a finite subset of Δ that is not satisfiable, which is a contradiction of Δ being finitely satisfiable.
  - Similarly, if either $\beta \notin \Delta$ or $\gamma \notin \Delta$, then $\beta \wedge \gamma \notin \Delta$, thus $\alpha \notin \Delta$.
    - Why? Again, suppose $\beta \wedge \gamma \in \Delta$. Since $\beta \notin \Delta$ or $\gamma \notin \Delta$, at least one of $\neg\beta$ or $\neg\gamma$ is in Δ. Wlog, assume $\neg\beta \in \Delta$. Then, $\{\neg\beta, \beta \wedge \gamma\}$ is a finite subset of Δ that is not satisfiable, which is a contradiction of Δ being finitely satisfiable.
  - Thus, $[\![\alpha]\!]_\nu = 1$ iff $\alpha \in \Delta$.

This shows that $[\![\alpha]\!]_\nu = 1$ iff $\alpha \in \Delta$, thus Δ is satisfiable by ν. $\square$

Compactness [5]

Corollary 1.1: If $\Sigma \vDash \alpha$, then there is a finite $\Sigma_0 \subseteq \Sigma$ such that $\Sigma_0 \vDash \alpha$.

Proof: Suppose that $\Sigma_0 \nvDash \alpha$ for every finite $\Sigma_0 \subseteq \Sigma$.

Then, $\Sigma_0 \cup \{\neg\alpha\}$ is satisfiable for every finite $\Sigma_0 \subseteq \Sigma$, that is, $\Sigma \cup \{\neg\alpha\}$ is finitely satisfiable.

Then, by the compactness theorem, $\Sigma \cup \{\neg\alpha\}$ is satisfiable, thus $\Sigma \nvDash \alpha$, which contradicts the theorem assumption that $\Sigma \vDash \alpha$. □

§4 Proof Systems

Natural Deduction

- **Natural deduction** is a proof system for propositional logic.
- **Axioms**:
  - **No axioms**.
- **Rules**:
  - **Introduction**: $\land$-introduction, $\lor$-introduction, $\to$-introduction, $\neg$-introduction.
  - **Elimination**: $\land$-elimination, $\lor$-elimination, $\to$-elimination, $\neg$-elimination.
  - **Reduction ad Absurdum**
  - **Law of Excluded Middle** (note: forbidden in *intuitionistic* logic)
- **Proofs** are constructed by applying rules to assumptions and previously derived formulas.

$$\underbrace{A_1, \ldots, A_n \vdash A}_{\text{sequent}} \qquad\qquad \frac{\Gamma_1 \vdash (\textit{premise 1}) \quad \Gamma_2 \vdash (\textit{premise 2}) \quad \ldots}{\Gamma \vdash (\textit{conclusion})}\ \text{rule name}$$

Inference Rules

$$\frac{}{\Gamma \vdash \varphi \lor \neg\varphi}\text{ law of excluded middle}$$

$$\frac{}{\Gamma, \varphi \vdash \varphi}\text{ assumption}$$

$$\frac{\Gamma \vdash \alpha \qquad \Gamma \vdash \neg\alpha}{\Gamma \vdash \beta}\text{ reduction ad absurdum}$$

$$\frac{\Gamma \vdash \alpha \land \beta}{\Gamma \vdash \alpha}\land\text{-elimination}$$

$$\frac{\Gamma \vdash \alpha \land \beta}{\Gamma \vdash \beta}\land\text{-elimination}$$

$$\frac{\Gamma \vdash \alpha \qquad \Gamma \vdash \beta}{\Gamma \vdash \alpha \land \beta}\land\text{-introduction}$$

$$\frac{\Gamma \vdash \alpha_1 \lor \alpha_2 \qquad \Gamma, \alpha_1 \vdash \beta \qquad \Gamma, \alpha_2 \vdash \beta}{\Gamma \vdash \beta}\lor\text{-elim}$$

$$\frac{\Gamma \vdash \alpha}{\Gamma \vdash \alpha \lor \beta}\lor\text{-intro}$$

$$\frac{\Gamma \vdash \beta}{\Gamma \vdash \alpha \lor \beta}\lor\text{-intro}$$

$$\frac{\Gamma \vdash \alpha \qquad \Gamma \vdash \alpha \to \beta}{\Gamma \vdash \beta}\to\text{-elimination}$$

$$\frac{\Gamma, \alpha \vdash \beta}{\Gamma \vdash \alpha \to \beta}\to\text{-introduction}$$

Example Derivation

Example: $\underbrace{p \land q, r}_{\text{premises}} \vdash \underbrace{q \land r}_{\text{conclusion}}$

Proof tree:

$$\dfrac{\dfrac{\overline{\;p \land q\;}}{q}\,{\land}\mathrm{e} \qquad \overline{\;r\;}}{q \land r}\,{\land}\mathrm{i}$$

Linear proof (Fitch notation):

1.	$p \land q$	**premise**
2.	r	**premise**
3.	q	**∧e 1**
4.	$q \land r$	**∧i 2,3**

Exercises

1. $\vdash (b \to c) \to ((\neg b \to \neg a) \to (a \to c))$
2. $a \lor b \vdash b \lor a$
3. $a \to c, b \to c, a \lor b \vdash c$
4. $\neg a \lor b \vdash a \to b$
5. $a \to b \vdash \neg a \lor b$
6. $a \to b, a \to \neg b \vdash \neg a$
7. $\neg p \to \bot \vdash p$ (with allowed $\neg\neg$E)
8. $\vdash p \lor \neg p$
9. $a \lor b, b \lor c, \neg b \vdash a \land c$
10. $a \lor (b \to a) \vdash \neg a \to \neg b$
11. $p \to \neg p \vdash \neg p$
12. $a \to b, \neg b \vdash \neg a$
13. $((a \to b) \to a) \to a$
14. $\neg a \to \neg b \vdash b \to a$
15. $\vdash (a \to b) \lor (b \to a)$

Soundness and Completeness

- A formal system is **sound** if every provable formula is true in all models.
  - **Weak soundness**: "every provable formula is a tautology".

    $\boxed{\text{If } \vdash \alpha, \text{ then } \vDash \alpha.}$

  - **Strong soundness**: "every derivable (from Γ) formula is a logical consequence (of Γ)".

    $\boxed{\text{If } \Gamma \vdash \alpha, \text{ then } \Gamma \vDash \alpha.}$

- A formal system is **complete** if every formula true in all models is provable.
  - **Weak completeness**: "every tautology is provable".

    $\boxed{\text{If } \vDash \alpha, \text{ then } \vdash \alpha.}$

  - **Strong completeness**: "every logical consequence (of Γ) is derivable (from Γ)".

    $\boxed{\text{If } \Gamma \vDash \alpha, \text{ then } \Gamma \vdash \alpha.}$

TODO

- [ ] Normal forms
- [ ] Canonical normal forms
- [ ] BDDs
- [x] Natural deduction
- [ ] Sequent calculus
- [ ] Fitch notation
- [ ] Proof checkers
- [ ] Proof assistants
- [ ] Automatic theorem provers
- [ ] Abstract proof systems
- [ ] Intuitionistic logic
- [x] Soundnsess and completeness
- [ ] Proof of soundness
- [ ] Proof of completeness