Part 1: Propositional Logic

Literature (also for first-order logic)

Schöning: Logik für Informatiker, Spektrum

Fitting: First-Order Logic and Automated Theorem Proving, Springer

Part 1: Propositional Logic

Propositional logic

- logic of truth values
- decidable (but NP-complete)
- can be used to describe functions over a finite domain
- important for hardware applications (e.g., model checking)

1.1 Syntax

- propositional variables
- logical symbols
  $\Rightarrow$ Boolean combinations

Propositional Variables

Let Π be a set of propositional variables.

We use letters P, Q, R, S, to denote propositional variables.

Propositional Formulas

F_Π is the set of propositional formulas over Π defined as follows:

$$
\begin{array}{llll}
F, G, H & ::= & \bot & \text{(falsum)} \\
 & \mid & \top & \text{(verum)} \\
 & \mid & P, \quad P \in \Pi & \text{(atomic formula)} \\
 & \mid & \neg F & \text{(negation)} \\
 & \mid & (F \land G) & \text{(conjunction)} \\
 & \mid & (F \lor G) & \text{(disjunction)} \\
 & \mid & (F \rightarrow G) & \text{(implication)} \\
 & \mid & (F \leftrightarrow G) & \text{(equivalence)}
\end{array}
$$

Notational Conventions

- We omit brackets according to the following rules:
  - $\neg \quad >_p \quad \wedge \quad >_p \quad \vee \quad >_p \quad \rightarrow \quad >_p \quad \leftrightarrow$ (binding precedences
  - $\vee$ and $\wedge$ are associative and commutative

1.2 Semantics

In classical logic (dating back to Aristoteles) there are “only” two truth values “true” and “false” which we shall denote, respectively, by 1 and 0.

There are multi-valued logics having more than two truth values.

Valuations

A propositional variable has no intrinsic meaning. The meaning of a propositional variable has to be defined by a valuation.

A Π-valuation is a map

$$\mathcal{A} : \Pi \to \{0, 1\}.$$

where $\{0, 1\}$ is the set of truth values.

Truth Value of a Formula in $\mathcal{A}$

Given a Π-valuation $\mathcal{A}$, the function $\mathcal{A}^*$: Σ-formulas $\rightarrow \{0, 1\}$ is defined inductively over the structure of F as follows:

$$\begin{aligned}
\mathcal{A}^*(\bot) &= 0 \\
\mathcal{A}^*(\top) &= 1 \\
\mathcal{A}^*(P) &= \mathcal{A}(P) \\
\mathcal{A}^*(\neg F) &= 1 - \mathcal{A}^*(F) \\
\mathcal{A}^*(F \rho G) &= \mathrm{B}_\rho(\mathcal{A}^*(F), \mathcal{A}^*(G)) \\
&\quad \text{with } \mathrm{B}_\rho \text{ the Boolean function associated with } \rho
\end{aligned}$$

For simplicity, we write $\mathcal{A}$ instead of $\mathcal{A}^*$.

Truth Value of a Formula in $\mathcal{A}$

Example: Let's evaluate the formula

$$(P \to Q) \wedge (P \wedge Q \to R) \to (P \to R)$$

w.r.t. the valuation $\mathcal{A}$ with

$$\mathcal{A}(P) = 1, \mathcal{A}(Q) = 0, \mathcal{A}(R) = 1$$

(On the blackboard)

1.3 Models, Validity, and Satisfiability

F is valid in $\mathcal{A}$ ($\mathcal{A}$ is a model of F; F holds under $\mathcal{A}$):

$$\mathcal{A} \models F :\Leftrightarrow \mathcal{A}(F) = 1$$

F is valid (or is a tautology):

$$\models F :\Leftrightarrow \mathcal{A} \models F \text{ for all } \Pi\text{-valuations } \mathcal{A}$$

F is called satisfiable iff there exists an $\mathcal{A}$ such that $\mathcal{A} \models F$.
Otherwise F is called unsatisfiable (or contradictory).

A set N of formulae is satisfiable iff there exists an $\mathcal{A}$ such that $\mathcal{A} \models F$ for all $F \in N$.
Otherwise N is called unsatisfiable (or contradictory).

Example

$F = (A \vee C) \wedge (B \vee \neg C)$

A	B	C	$(A \vee C)$	$\neg C$	$(B \vee \neg C)$	$(A \vee C) \wedge (B \vee \neg C)$
0	0	0	0	1	1	0
0	0	1	1	0	0	0
0	1	0	0	1	1	0
0	1	1	1	0	1	1
1	0	0	1	1	1	1
1	0	1	1	0	0	0
1	1	0	1	1	1	1
1	1	1	1	0	1	1

Let $\mathcal{A} : \{A, B, C\} \to \{0, 1\}$ with $\mathcal{A}(A) = 0, \mathcal{A}(B) = 1, \mathcal{A}(C) = 1$.

$\mathcal{A} \models (A \vee C), \quad \mathcal{A} \models (B \vee \neg C)$
$\mathcal{A} \models (A \vee C) \wedge (B \vee \neg C)$
$\mathcal{A} \models \{(A \vee C), (B \vee \neg C)\}$

1.3 Models, Validity, and Satisfiability

Examples:

$F \rightarrow F$ and $F \vee \neg F$ are valid for all formulae F.

Obviously, every valid formula is also satisfiable

$F \wedge \neg F$ is unsatisfiable

The formula P is satisfiable, but not valid

Example

$F = (A \vee C) \wedge (B \vee \neg C)$

A	B	C	$(A \vee C)$	$\neg C$	$(B \vee \neg C)$	$(A \vee C) \wedge (B \vee \neg C)$
0	0	0	0	1	1	0
0	0	1	1	0	0	0
0	1	0	0	1	1	0
0	1	1	1	0	1	1
1	0	0	1	1	1	1
1	0	1	1	0	0	0
1	1	0	1	1	1	1
1	1	1	1	0	1	1

F is not valid:

$\mathcal{A}_1(F) = 0$ für $\mathcal{A}_1 : \{A, B, C\} \to \{0, 1\}$ mit $\mathcal{A}(A) = \mathcal{A}(B) = \mathcal{A}(C) = 0$.

F is satisfiable:

$\mathcal{A}_2(F) = 1$ für $\mathcal{A} : \{A, B, C\} \to \{0, 1\}$ mit $\mathcal{A}(A) = 0, \mathcal{A}(B) = 1, \mathcal{A}(C) = 1$.

Entailment and Equivalence

F entails (implies) G (or G is a consequence of F), written $F \models G$, if for all Π-valuations $\mathcal{A}$, whenever $\mathcal{A} \models F$ then $\mathcal{A} \models G$.

F and G are called equivalent if for all Π-valuations $\mathcal{A}$ we have $\mathcal{A} \models F \Leftrightarrow \mathcal{A} \models G$.

Example

$F = (A \vee C) \wedge (B \vee \neg C) \qquad G = (A \vee B)$

Check if $F \models G$

A	B	C	$(A \vee C)$	$(B \vee \neg C)$	$(A \vee C) \wedge (B \vee \neg C)$	$(A \vee B)$
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

Example

$F = (A \lor C) \land (B \lor \neg C) \qquad G = (A \lor B)$

Check if $F \models G$

A	B	C	$(A \lor C)$	$(B \lor \neg C)$	$(A \lor C) \land (B \lor \neg C)$	$(A \lor B)$
0	0	0	0	1	0	0
0	0	1	1	0	0	0
0	1	0	0	1	0	1
0	1	1	1	1	1	1
1	0	0	1	1	1	1
1	0	1	1	0	0	1
1	1	0	1	1	1	1
1	1	1	1	1	1	1

Example

$F = (A \lor C) \land (B \lor \neg C) \qquad G = (A \lor B)$

Check if $F \models G$ Yes, $F \models G$

A	B	C	$(A \lor C)$	$(B \lor \neg C)$	$(A \lor C) \land (B \lor \neg C)$	$(A \lor B)$
0	0	1	1	0	0	0
0	0	0	0	1	0	0
0	1	1	1	1	1	1
0	1	0	0	1	0	1
1	0	1	1	0	0	1
1	0	0	1	1	1	1
1	1	1	1	1	1	1
1	1	0	1	1	1	1

Example

$F = (A \vee C) \wedge (B \vee \neg C)$ $\qquad$ $G = (A \vee B)$

Check if $F \models G$ Yes, $F \models G$

... But it is not true that $G \models F$ (Notation: $G \not\models F$)

A	B	C	$(A \vee C)$	$(B \vee \neg C)$	$(A \vee C) \wedge (B \vee \neg C)$	$(A \vee B)$
0	0	1	1	0	0	0
0	0	0	0	1	0	0
0	1	1	1	1	1	1
0	1	0	0	1	0	1
1	0	1	1	0	0	1
1	0	0	1	1	1	1
1	1	1	1	1	1	1
1	1	0	1	1	1	1

Entailment and Equivalence

F entails (implies) G (or G is a consequence of F), written $F \models G$, if for all Π-valuations $\mathcal{A}$, whenever $\mathcal{A} \models F$ then $\mathcal{A} \models G$.

F and G are called equivalent if for all Π-valuations $\mathcal{A}$ we have $\mathcal{A} \models F \Leftrightarrow \mathcal{A} \models G$.

Proposition 1.1:
F entails G iff $(F \rightarrow G)$ is valid

Proposition 1.2:

F and G are equivalent iff $(F \leftrightarrow G)$ is valid.

Entailment and Equivalence

Extension to sets of formulas N in the "natural way", e.g., $N \models F$ if for all Π-valuations $\mathcal{A}$: if $\mathcal{A} \models G$ for all $G \in N$, then $\mathcal{A} \models F$.