Foundations of Artificial Intelligence

E2. Propositional Logic: Equivalence, Normal Forms and Reasoning

Malte Helmert

University of Basel

April 20, 2026

Propositional Logic: Overview

Chapter overview: propositional logic

- E1. Syntax and Semantics
- E2. Equivalence, Normal Forms and Reasoning
- E3. Resolution
- E4. DPLL Algorithm
- E5. Local Search and Outlook

Equivalence

Logical Equivalance

Definition (logically equivalent)

Formulas φ and ψ are called logically equivalent ($\varphi \equiv \psi$) if for all interpretations I: $I \models \varphi$ iff $I \models \psi$.

German: logisch äquivalent

Equivalences

Logical Equivalences

Let φ, ψ, and η be formulas.

- $(\varphi \wedge \psi) \equiv (\psi \wedge \varphi)$ and $(\varphi \vee \psi) \equiv (\psi \vee \varphi)$ (commutativity)
- $((\varphi \wedge \psi) \wedge \eta) \equiv (\varphi \wedge (\psi \wedge \eta))$ and $((\varphi \vee \psi) \vee \eta) \equiv (\varphi \vee (\psi \vee \eta))$ (associativity)
- $((\varphi \wedge \psi) \vee \eta) \equiv ((\varphi \vee \eta) \wedge (\psi \vee \eta))$ and $((\varphi \vee \psi) \wedge \eta) \equiv ((\varphi \wedge \eta) \vee (\psi \wedge \eta))$ (distributivity)
- $\neg(\varphi \wedge \psi) \equiv (\neg\varphi \vee \neg\psi)$ and $\neg(\varphi \vee \psi) \equiv (\neg\varphi \wedge \neg\psi)$ (De Morgan)
- $\neg\neg\varphi \equiv \varphi$ (double negation)
- $(\varphi \rightarrow \psi) \equiv (\neg\varphi \vee \psi)$ (($\rightarrow$)-elimination)

Commutativity and associativity are often used implicitly

$\rightsquigarrow$ We write $(X_1 \wedge X_2 \wedge X_3 \wedge X_4)$ instead of $(X_1 \wedge (X_2 \wedge (X_3 \wedge X_4)))$

Normal Forms

Normal Forms: Terminology

Definition (literal)

If $P \in \Sigma$, then the formulas P and $\neg P$ are called literals.

P is called positive literal, $\neg P$ is called negative literal.

The complementary literal to P is $\neg P$ and vice versa.
For a literal ℓ, the complementary literal to ℓ is denoted with $\bar{\ell}$.

German: Literal, positives/negatives/komplementäres Literal

Question: What is the difference between $\bar{\ell}$ and $\neg\ell$?

Normal Forms: Terminology

Definition (clause)

A disjunction of 0 or more literals is called a clause.
The empty clause (with 0 literals) is $\bot$.
Clauses consisting of exactly one literal are called unit clauses.

German: Klausel, leere Klausel, Einheitsklausel

Definition (monomial)

A conjunction of 0 or more literals is called a monomial.

German: Monom

Normal Forms

Definition (normal forms)

A formula φ is in conjunctive normal form (CNF, clause form) if φ is a conjunction of 0 or more clauses:

$$\varphi = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{m_i} \ell_{i,j} \right)$$

A formula φ is in disjunctive normal form (DNF) if φ is a disjunction of 0 or more monomials:

$$\varphi = \bigvee_{i=1}^{n} \left(\bigwedge_{j=1}^{m_i} \ell_{i,j} \right)$$

German: konjunktive Normalform, disjunktive Normalform

Normal Forms

For every propositional formula, there exists
a logically equivalent propositional formula in CNF and in DNF.

Conversion to CNF with equivalences

1. eliminate implications
   $(\varphi \rightarrow \psi) \equiv (\neg\varphi \vee \psi)$ (($\rightarrow$)-elimination)
2. move negations inside
   $\neg(\varphi \wedge \psi) \equiv (\neg\varphi \vee \neg\psi)$ (De Morgan)
   $\neg(\varphi \vee \psi) \equiv (\neg\varphi \wedge \neg\psi)$ (De Morgan)
   $\neg\neg\varphi \equiv \varphi$ (double negation)
3. distribute $\vee$ over $\wedge$
   $((\varphi \wedge \psi) \vee \eta) \equiv ((\varphi \vee \eta) \wedge (\psi \vee \eta))$ (distributivity)
4. simplify constant subformulas $(\top, \bot)$

There are formulas φ for which every logically equivalent formula in CNF and DNF is exponentially longer than φ.

Reasoning

Reasoning: Intuition

Reasoning: Intuition

- Generally, formulas only represent
  an incomplete description of the world.
- In many cases, we want to know
  if a formula logically follows from (a set of) other formulas.
- What does this mean?

Reasoning: Intuition

- example: $\varphi = (P \vee Q) \wedge (R \vee \neg P) \wedge S$
- S holds in every model of φ.
  What about P, Q and R?

⇝ consider all models of φ:

- $I_1 = \{P \mapsto \mathbf{F}, Q \mapsto \mathbf{T}, R \mapsto \mathbf{F}, S \mapsto \mathbf{T}\}$
- $I_2 = \{P \mapsto \mathbf{F}, Q \mapsto \mathbf{T}, R \mapsto \mathbf{T}, S \mapsto \mathbf{T}\}$
- $I_3 = \{P \mapsto \mathbf{T}, Q \mapsto \mathbf{F}, R \mapsto \mathbf{T}, S \mapsto \mathbf{T}\}$
- $I_4 = \{P \mapsto \mathbf{T}, Q \mapsto \mathbf{T}, R \mapsto \mathbf{T}, S \mapsto \mathbf{T}\}$

Observation

- In all models of φ, the formula $Q \vee R$ holds as well.
- We say: "$Q \vee R$ logically follows from φ."

Reasoning: Formally

Definition (logical consequence)

Let Φ be a set of formulas. A formula ψ logically follows from Φ (in symbols: $\Phi \models \psi$) if all models of Φ are also models of ψ.

German: logische Konsequenz, folgt logisch

In other words: for each interpretation I,
if $I \models \varphi$ for all $\varphi \in \Phi$, then also $I \models \psi$.

Question

How can we automatically compute if $\Phi \models \psi$?

- One possibility: Build a truth table. (How?)
- Are there "better" possibilities that (potentially) avoid generating the whole truth table?

Reasoning: Deduction Theorem

Proposition (deduction theorem)

Let Φ be a finite set of formulas and let ψ be a formula. Then

$$\Phi \models \psi \quad \textit{iff} \quad \left(\bigwedge_{\varphi \in \Phi} \varphi\right) \rightarrow \psi \textit{ is a tautology.}$$

German: Deduktionssatz

Reasoning: Deduction Theorem

Proposition (deduction theorem)

Let Φ be a finite set of formulas and let ψ be a formula. Then

$$\Phi \models \psi \quad \textit{iff} \quad (\bigwedge_{\varphi \in \Phi} \varphi) \to \psi \textit{ is a tautology.}$$

German: Deduktionssatz

Proof.

$\quad \Phi \models \psi$

iff for each interpretation I: if $I \models \varphi$ for all $\varphi \in \Phi$, then $I \models \psi$

iff for each interpretation I: if $I \models \bigwedge_{\varphi \in \Phi} \varphi$, then $I \models \psi$

iff for each interpretation I: $I \not\models \bigwedge_{\varphi \in \Phi} \varphi$ or $I \models \psi$

iff for each interpretation I: $I \models (\bigwedge_{\varphi \in \Phi} \varphi) \to \psi$

iff $(\bigwedge_{\varphi \in \Phi} \varphi) \to \psi$ is tautology $\square$

Reasoning by Unsatisfiability Testing

Consequence of Deduction Theorem

Reasoning can be reduced to testing unsatisfiability.

Question: Does $\Phi \models \psi$ hold?

Idea:

- Let $\chi = (\bigwedge_{\varphi \in \Phi} \varphi) \to \psi$.
- We know that $\Phi \models \psi$ iff χ is a tautology.
- A formula is a tautology iff its negation is unsatisfiable.
- Hence, $\Phi \models \psi$ iff $\neg\chi$ is unsatisfiable.
- Use equivalences:
  $$\neg\chi = \neg((\bigwedge_{\varphi \in \Phi} \varphi) \to \psi) \equiv \neg(\neg(\bigwedge_{\varphi \in \Phi} \varphi) \lor \psi)$$
  $$\equiv (\neg\neg(\bigwedge_{\varphi \in \Phi} \varphi) \land \neg\psi) \equiv \bigwedge_{\varphi \in \Phi} \varphi \land \neg\psi$$
- We have that $\Phi \models \psi$ iff $\bigwedge_{\varphi \in \Phi} \varphi \land \neg\psi$ is unsatisfiable.

Algorithm for Reasoning

Question: Does $\Phi \models \psi$ hold?

Algorithm (given an algorithm for testing unsatisfiability):

1. Let $\eta = \bigwedge_{\varphi \in \Phi} \varphi \land \neg\psi$.
2. Test if η is unsatisfiable.
3. If yes, return "$\Phi \models \psi$".
4. Otherwise, return "$\Phi \not\models \psi$".

Can we test unsatisfiability in a more efficient way than by computing the whole truth table?
$\rightsquigarrow$ next chapters

Summary

Summary

- two formulas are logically equivalent if they have the same models
- different kinds of formulas:
  - atomic formulas and literals
  - clauses and monomials
  - conjunctive normal form (CNF) and disjunctive normal form (DNF)
- for every formula, there is a logically equivalent formula in CNF and a logically equivalent formula in DNF
- Reasoning: the formula ψ follows from the set of formulas Φ if all models of Φ are also models of ψ.
- Reasoning can be reduced to testing validity (with the deduction theorem).
- Testing validity can be reduced to testing unsatisfiability.