Discrete Mathematics in Computer Science

D3. Normal Forms and Logical Consequence

Malte Helmert, Gabriele Röger

University of Basel

December 4/6, 2023

Discrete Mathematics in Computer Science

December 4/6, 2023 — D3. Normal Forms and Logical Consequence

D3.1 Simplified Notation

Parentheses

Associativity:

$$((\varphi \land \psi) \land \chi) \equiv (\varphi \land (\psi \land \chi))$$
$$((\varphi \lor \psi) \lor \chi) \equiv (\varphi \lor (\psi \lor \chi))$$

- Placement of parentheses for a conjunction of conjunctions does not influence whether an interpretation is a model.
- ditto for disjunctions of disjunctions

→ can omit parentheses and treat this as if parentheses placed arbitrarily

- Example: $(A_1 \land A_2 \land A_3 \land A_4)$ instead of $((A_1 \land (A_2 \land A_3)) \land A_4)$
- Example: $(\neg A \lor (B \land C) \lor D)$ instead of $((\neg A \lor (B \land C)) \lor D)$

Parentheses

Does this mean we can always omit all parentheses and assume an arbitrary placement? → No!

$$((\varphi \land \psi) \lor \chi) \not\equiv (\varphi \land (\psi \lor \chi))$$

What should $\varphi \land \psi \lor \chi$ mean?

Placement of Parentheses by Convention

Often parentheses can be dropped in specific cases and an implicit placement is assumed:

- $\neg$ binds more strongly than $\land$
- $\land$ binds more strongly than $\lor$
- $\lor$ binds more strongly than $\rightarrow$ or $\leftrightarrow$

→ cf. PEMDAS/"Punkt vor Strich"

Example

$A \lor \neg C \land B \rightarrow A \lor \neg D$ stands for $((A \lor (\neg C \land B)) \rightarrow (A \lor \neg D))$

- often harder to read
- error-prone

→ not used in this course

Short Notations for Conjunctions and Disjunctions

Short notation for addition:

$$\sum_{i=1}^{n} x_i = x_1 + x_2 + \cdots + x_n$$
$$\sum_{x \in \{x_1,\ldots,x_n\}} x = x_1 + x_2 + \cdots + x_n$$

Analogously:

$$\bigwedge_{i=1}^{n} \varphi_i = (\varphi_1 \land \varphi_2 \land \cdots \land \varphi_n)$$
$$\bigvee_{i=1}^{n} \varphi_i = (\varphi_1 \lor \varphi_2 \lor \cdots \lor \varphi_n)$$
$$\bigwedge_{\varphi \in X} \varphi = (\varphi_1 \land \varphi_2 \land \cdots \land \varphi_n)$$
$$\bigvee_{\varphi \in X} \varphi = (\varphi_1 \lor \varphi_2 \lor \cdots \lor \varphi_n)$$

for $X = \{\varphi_1, \ldots, \varphi_n\}$

Short Notation: Corner Cases

Is $\mathcal{I} \models \psi$ true for

$$\psi = \bigwedge_{\varphi \in X} \varphi \text{ and } \psi = \bigvee_{\varphi \in X} \varphi$$

if $X = \emptyset$ or $X = \{\chi\}$?

convention:

- $\bigwedge_{\varphi \in \emptyset} \varphi$ is a tautology.
- $\bigvee_{\varphi \in \emptyset} \varphi$ is unsatisfiable.
- $\bigwedge_{\varphi \in \{\chi\}} \varphi = \bigvee_{\varphi \in \{\chi\}} \varphi = \chi$

⇝ Why?

Exercise

Express $\bigwedge_{i=1}^{2} \bigvee_{j=1}^{3} \varphi_{ij}$ without $\bigwedge$ and $\bigvee$.

D3.2 Normal Forms

Why Normal Forms?

- A normal form is a representation with certain syntactic restrictions.
- condition for reasonable normal form: every formula must have a logically equivalent formula in normal form
- advantages:
  - can restrict proofs to formulas in normal form
  - can define algorithms only for formulas in normal form

German: Normalform

Literals, Clauses and Monomials

- A literal is an atomic proposition or the negation of an atomic proposition (e. g., A and $\neg A$).
- A clause is a disjunction of literals (e. g., $(Q \lor \neg P \lor \neg S \lor R)$).
- A monomial is a conjunction of literals (e. g., $(Q \land \neg P \land \neg S \land R)$).

The terms clause and monomial are also used for the corner case with only one literal.

German: Literal, Klausel, Monom

Terminology: Examples

Examples
- (¬Q ∧ R) is a monomial
- (P ∨ ¬Q) is a clause
- ((P ∨ ¬Q) ∧ P) is neither literal nor clause nor monomial
- ¬P is a literal, a clause and a monomial
- (P → Q) is neither literal nor clause nor monomial (but (¬P ∨ Q) is a clause!)
- (P ∨ P) is a clause, but not a literal or monomial
- ¬¬P is neither literal nor clause nor monomial

Conjunctive Normal Form

Definition (Conjunctive Normal Form)
A formula is in conjunctive normal form (CNF) if it is a conjunction of clauses, i. e., if it has the form

$$\bigwedge_{i=1}^{n} \bigvee_{j=1}^{m_i} L_{ij}$$

with $n, m_i > 0$ (for $1 \leq i \leq n$), where the L_{ij} are literals.

German: konjunktive Normalform (KNF)

Example
((¬P ∨ Q) ∧ R ∧ (P ∨ ¬S)) is in CNF.

Disjunctive Normal Form

Definition (Disjunctive Normal Form)
A formula is in disjunctive normal form (DNF) if it is a disjunction of monomials, i. e., if it has the form

$$\bigvee_{i=1}^{n} \bigwedge_{j=1}^{m_i} L_{ij}$$

with $n, m_i > 0$ (for $1 \leq i \leq n$), where the L_{ij} are literals.

German: disjunktive Normalform (DNF)

Example
((¬P ∧ Q) ∨ R ∨ (P ∧ ¬S)) is in DNF.

CNF and DNF: Examples

Which of the following formulas are in CNF? Which are in DNF?

- ((P ∨ ¬Q) ∧ P)
- ((R ∨ Q) ∧ P ∧ (R ∨ S))
- (P ∨ (¬Q ∧ R))
- ((P ∨ ¬Q) → P)
- P

Construction of CNF (and DNF)

Algorithm to Construct CNF

1. Replace abbreviations $\rightarrow$ and $\leftrightarrow$ by their definitions (($\rightarrow$)-elimination and ($\leftrightarrow$)-elimination).
   $\rightsquigarrow$ formula structure: only $\lor$, $\land$, $\neg$
2. Move negations inside using De Morgan and double negation.
   $\rightsquigarrow$ formula structure: only $\lor$, $\land$, literals
3. Distribute $\lor$ over $\land$ with distributivity (strictly speaking also with commutativity).
   $\rightsquigarrow$ formula structure: CNF
4. optionally: Simplify the formula at the end or at intermediate steps (e. g., with idempotence).

Note: For DNF, distribute $\land$ over $\lor$ instead.

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \land \neg Q) \lor R) \rightarrow (P \lor \neg(S \lor T)))$

$$\begin{aligned}
\varphi &\equiv (\neg((P \land \neg Q) \lor R) \lor P \lor \neg(S \lor T)) && \text{[Step 1]}\\
&\equiv ((\neg(P \land \neg Q) \land \neg R) \lor P \lor \neg(S \lor T)) && \text{[Step 2]}\\
&\equiv (((\neg P \lor \neg\neg Q) \land \neg R) \lor P \lor \neg(S \lor T)) && \text{[Step 2]}\\
&\equiv (((\neg P \lor Q) \land \neg R) \lor P \lor \neg(S \lor T)) && \text{[Step 2]}\\
&\equiv (((\neg P \lor Q) \land \neg R) \lor P \lor (\neg S \land \neg T)) && \text{[Step 2]}\\
&\equiv ((\neg P \lor Q \lor P \lor (\neg S \land \neg T)) \land \\
&\qquad (\neg R \lor P \lor (\neg S \land \neg T))) && \text{[Step 3]}\\
&\equiv (\neg R \lor P \lor (\neg S \land \neg T)) && \text{[Step 4]}\\
&\equiv ((\neg R \lor P \lor \neg S) \land (\neg R \lor P \lor \neg T)) && \text{[Step 3]}
\end{aligned}$$

Construct DNF: Example

Construction of Disjunctive Normal Form

Given: $\varphi = (((P \land \neg Q) \lor R) \rightarrow (P \lor \neg(S \lor T)))$

$$\begin{aligned}
\varphi &\equiv (\neg((P \land \neg Q) \lor R) \lor P \lor \neg(S \lor T)) && \text{[Step 1]}\\
&\equiv ((\neg(P \land \neg Q) \land \neg R) \lor P \lor \neg(S \lor T)) && \text{[Step 2]}\\
&\equiv (((\neg P \lor \neg\neg Q) \land \neg R) \lor P \lor \neg(S \lor T)) && \text{[Step 2]}\\
&\equiv (((\neg P \lor Q) \land \neg R) \lor P \lor \neg(S \lor T)) && \text{[Step 2]}\\
&\equiv (((\neg P \lor Q) \land \neg R) \lor P \lor (\neg S \land \neg T)) && \text{[Step 2]}\\
&\equiv ((\neg P \land \neg R) \lor (Q \land \neg R) \lor P \lor (\neg S \land \neg T)) && \text{[Step 3]}
\end{aligned}$$

Existence of an Equivalent Formula in Normal Form

Theorem

For every formula φ there is a logically equivalent formula in CNF and a logically equivalent formula in DNF.

- "There is a" always means "there is at least one". Otherwise we would write "there is exactly one".
- Intuition: algorithm to construct normal form works with any given formula and only uses equivalence rewriting.
- actual proof would use induction over structure of formula

Size of Normal Forms

- In the worst case, a logically equivalent formula in CNF or DNF can be exponentially larger than the original formula.
- Example: for $(x_1 \lor y_1) \land \dots \land (x_n \lor y_n)$ there is no smaller logically equivalent formula in DNF than:

$$\bigvee_{S \in \mathcal{P}(\{1,\dots,n\})} \left(\bigwedge_{i \in S} x_i \land \bigwedge_{i \in \{1,\dots,n\} \setminus S} y_i \right)$$

- As a consequence, the construction of the CNF/DNF formula can take exponential time.

More Theorems

Theorem

A formula in CNF is a tautology iff every clause is a tautology.

Theorem

A formula in DNF is satisfiable iff at least one of its monomials is satisfiable.

$\rightsquigarrow$ both proved easily with semantics of propositional logic

D3.3 Knowledge Bases

Knowledge Bases: Example

If not DrinkBeer, then EatFish.
If EatFish and DrinkBeer,
then not EatIceCream.
If EatIceCream or not DrinkBeer,
then not EatFish.

$$\begin{aligned} \text{KB} = \{&(\neg \text{DrinkBeer} \rightarrow \text{EatFish}), \\ &((\text{EatFish} \land \text{DrinkBeer}) \rightarrow \neg \text{EatIceCream}), \\ &((\text{EatIceCream} \lor \neg \text{DrinkBeer}) \rightarrow \neg \text{EatFish})\} \end{aligned}$$

Exercise from U. Schöning: Logik für Informatiker
Picture courtesy of graur razvan ionut / FreeDigitalPhotos.net

Models for Sets of Formulas

Definition (Model for Knowledge Base)
Let KB be a knowledge base over A,
i. e., a set of propositional formulas over A.

A truth assignment $\mathcal{I}$ for A is a model for KB (written: $\mathcal{I} \models \text{KB}$)
if $\mathcal{I}$ is a model for every formula $\varphi \in \text{KB}$.

German: Wissensbasis, Modell

Properties of Sets of Formulas

A knowledge base KB is

- satisfiable if KB has at least one model
- unsatisfiable if KB is not satisfiable
- valid (or a tautology) if every interpretation is a model for KB
- falsifiable if KB is no tautology

German: erfüllbar, unerfüllbar, gültig, gültig/eine Tautologie, falsifizierbar

Example I

Which of the properties does $\text{KB} = \{(\text{A} \wedge \neg \text{B}), \neg(\text{B} \vee \text{A})\}$ have?

KB is unsatisfiable:
For every model $\mathcal{I}$ with $\mathcal{I} \models (\text{A} \wedge \neg \text{B})$ we have $\mathcal{I}(\text{A}) = 1$.
This means $\mathcal{I} \models (\text{B} \vee \text{A})$ and thus $\mathcal{I} \not\models \neg(\text{B} \vee \text{A})$.

This directly implies that KB is falsifiable, not satisfiable
and no tautology.

Example II

Which of the properties does

$$\begin{aligned}\text{KB} = \{&(\neg \text{DrinkBeer} \rightarrow \text{EatFish}),\\ &((\text{EatFish} \wedge \text{DrinkBeer}) \rightarrow \neg \text{EatIceCream}),\\ &((\text{EatIceCream} \vee \neg \text{DrinkBeer}) \rightarrow \neg \text{EatFish})\}\end{aligned}$$

have?

- satisfiable, e. g. with
  $\mathcal{I} = \{\text{EatFish} \mapsto 1, \text{DrinkBeer} \mapsto 1, \text{EatIceCream} \mapsto 0\}$
- thus not unsatisfiable
- falsifiable, e. g. with
  $\mathcal{I} = \{\text{EatFish} \mapsto 0, \text{DrinkBeer} \mapsto 0, \text{EatIceCream} \mapsto 1\}$
- thus not valid

D3.4 Logical Consequences

Logical Consequences: Motivation

What's the secret of your long life?

I am on a strict diet: If I don't drink beer to a meal, then I always eat fish. Whenever I have fish and beer with the same meal, I abstain from ice cream. When I eat ice cream or don't drink beer, then I never touch fish.

Claim: the woman drinks beer to every meal.

How can we prove this?

Exercise from U. Schöning: Logik für Informatiker
Picture courtesy of graur razvan ionut/FreeDigitalPhotos.net

Logical Consequences

Definition (Logical Consequence)
Let KB be a set of formulas and φ a formula.

We say that KB logically implies φ (written as $\mathrm{KB} \models \varphi$) if all models of KB are also models of φ.

also: KB logically entails φ, φ logically follows from KB, φ is a logical consequence of KB

German: KB impliziert φ logisch, φ folgt logisch aus KB, φ ist logische Konsequenz von KB

Attention: the symbol $\models$ is "overloaded": $\mathrm{KB} \models \varphi$ vs. $\mathcal{I} \models \varphi$.

What if KB is unsatisfiable or the empty set?

Logical Consequences: Example

Let $\varphi = \mathrm{DrinkBeer}$ and

$$\begin{aligned}\mathrm{KB} = \{&(\neg\mathrm{DrinkBeer} \rightarrow \mathrm{EatFish}),\\ &((\mathrm{EatFish} \land \mathrm{DrinkBeer}) \rightarrow \neg\mathrm{EatIceCream}),\\ &((\mathrm{EatIceCream} \lor \neg\mathrm{DrinkBeer}) \rightarrow \neg\mathrm{EatFish})\}.\end{aligned}$$

Show: $\mathrm{KB} \models \varphi$

Proof sketch.
Proof by contradiction: assume $\mathcal{I} \models \mathrm{KB}$, but $\mathcal{I} \not\models \mathrm{DrinkBeer}$.
Then it follows that $\mathcal{I} \models \neg\mathrm{DrinkBeer}$.
Because $\mathcal{I}$ is a model of KB, we also have $\mathcal{I} \models (\neg\mathrm{DrinkBeer} \rightarrow \mathrm{EatFish})$ and thus $\mathcal{I} \models \mathrm{EatFish}$. (Why?)
With an analogous argumentation starting from $\mathcal{I} \models ((\mathrm{EatIceCream} \lor \neg\mathrm{DrinkBeer}) \rightarrow \neg\mathrm{EatFish})$ we get $\mathcal{I} \models \neg\mathrm{EatFish}$ and thus $\mathcal{I} \not\models \mathrm{EatFish}$. $\rightsquigarrow$ Contradiction!

Important Theorems about Logical Consequences

Theorem (Deduction Theorem)
$\mathrm{KB} \cup \{\varphi\} \models \psi$ *iff* $\mathrm{KB} \models (\varphi \rightarrow \psi)$

German: Deduktionssatz

Theorem (Contraposition Theorem)
$\mathrm{KB} \cup \{\varphi\} \models \neg\psi$ *iff* $\mathrm{KB} \cup \{\psi\} \models \neg\varphi$

German: Kontrapositionssatz

Theorem (Contradiction Theorem)
$\mathrm{KB} \cup \{\varphi\}$ *is unsatisfiable iff* $\mathrm{KB} \models \neg\varphi$

German: Widerlegungssatz

(without proof)