Discrete Mathematics in Computer Science

Simplified Notation

Malte Helmert, Gabriele Röger

University of Basel

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

$\rightarrow$ 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? $\rightarrow$ No!

Parentheses

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

$$((\varphi \wedge \psi) \vee \chi) \not\equiv (\varphi \wedge (\psi \vee \chi))$$

Parentheses

Does this mean we can always omit all parentheses and assume an arbitrary placement? $\rightarrow$ 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 $\wedge$
- $\wedge$ binds more strongly than $\vee$
- $\vee$ binds more strongly than $\rightarrow$ or $\leftrightarrow$

$\rightarrow$ cf. PEMDAS/"Punkt vor Strich"

Placement of Parentheses by Convention

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

- $\neg$ binds more strongly than $\wedge$
- $\wedge$ binds more strongly than $\vee$
- $\vee$ binds more strongly than $\rightarrow$ or $\leftrightarrow$

$\rightarrow$ cf. PEMDAS/"Punkt vor Strich"

Example

$A \vee \neg C \wedge B \rightarrow A \vee \neg D$ stands for $A \vee \neg C \wedge B \rightarrow A \vee \neg D$

Placement of Parentheses by Convention

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

- $\neg$ binds more strongly than $\wedge$
- $\wedge$ binds more strongly than $\vee$
- $\vee$ binds more strongly than $\rightarrow$ or $\leftrightarrow$

$\rightarrow$ cf. PEMDAS/"Punkt vor Strich"

Example

$A \vee \neg C \wedge B \rightarrow A \vee \neg D$ stands for $A \vee (\neg C \wedge B) \rightarrow A \vee \neg D$

Placement of Parentheses by Convention

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

- $\neg$ binds more strongly than $\wedge$
- $\wedge$ binds more strongly than $\vee$
- $\vee$ binds more strongly than $\rightarrow$ or $\leftrightarrow$

$\rightarrow$ cf. PEMDAS/"Punkt vor Strich"

Example

$A \vee \neg C \wedge B \rightarrow A \vee \neg D$ stands for $(A \vee (\neg C \wedge B)) \rightarrow (A \vee \neg D)$

Placement of Parentheses by Convention

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

- $\neg$ binds more strongly than $\wedge$
- $\wedge$ binds more strongly than $\vee$
- $\vee$ binds more strongly than $\rightarrow$ or $\leftrightarrow$

$\rightarrow$ cf. PEMDAS/ "Punkt vor Strich"

Example

$A \vee \neg C \wedge B \rightarrow A \vee \neg D$ stands for $((A \vee (\neg C \wedge B)) \rightarrow (A \vee \neg D))$

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 $\leftrightarrow$

$\rightarrow$ 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

$\rightarrow$ not used in this course

Short Notations for Conjunctions and Disjunctions

Short notation for addition:

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

Short Notations for Conjunctions and Disjunctions

Short notation for addition:

$$\sum\nolimits_{i=1}^{n} x_i = x_1 + x_2 + \cdots + x_n$$

Analogously:

$$\bigwedge\nolimits_{i=1}^{n} \varphi_i = (\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n)$$
$$\bigvee\nolimits_{i=1}^{n} \varphi_i = (\varphi_1 \vee \varphi_2 \vee \cdots \vee \varphi_n)$$

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:

$$\bigwedge_{i=1}^{n} \varphi_i = (\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n)$$

$$\bigvee_{i=1}^{n} \varphi_i = (\varphi_1 \vee \varphi_2 \vee \cdots \vee \varphi_n)$$

Short Notations for Conjunctions and Disjunctions

Short notation for addition:

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

Analogously (possible because of commutativity of $\wedge$ and $\vee$):

$$\bigwedge_{i=1}^{n} \varphi_i = (\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n)$$
$$\bigvee_{i=1}^{n} \varphi_i = (\varphi_1 \vee \varphi_2 \vee \cdots \vee \varphi_n)$$
$$\bigwedge_{\varphi \in X} \varphi = (\varphi_1 \wedge \varphi_2 \wedge \cdots \wedge \varphi_n)$$
$$\bigvee_{\varphi \in X} \varphi = (\varphi_1 \vee \varphi_2 \vee \cdots \vee \varphi_n)$$
$$\text{for } X = \{\varphi_1, \dots, \varphi_n\}$$

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 $X = \{\chi\}$?

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.
- $\bigwedge_{\varphi \in \{\chi\}} \varphi = \bigvee_{\varphi \in \{\chi\}} \varphi = \chi$

Discrete Mathematics in Computer Science

Normal Forms

Malte Helmert, Gabriele Röger

University of Basel

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).

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., $(Q \lor \neg P \lor \neg S \lor R)$).

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., $(Q \vee \neg P \vee \neg S \vee R)$).
- A monomial is a conjunction of literals
  (e. g., $(Q \wedge \neg P \wedge \neg S \wedge R)$).

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., $(Q \vee \neg P \vee \neg S \vee R)$).
- A monomial is a conjunction of literals (e. g., $(Q \wedge \neg P \wedge \neg S \wedge R)$).

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

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

- $(\neg Q \wedge R)$
- $(P \vee \neg Q)$
- $((P \vee \neg Q) \wedge P)$
- $\neg P$
- $(P \rightarrow Q)$

- $(P \vee P)$
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$
- $((P \vee \neg Q) \wedge P)$
- $\neg P$
- $(P \rightarrow Q)$
- $(P \vee P)$
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$ is a clause
- $((P \vee \neg Q) \wedge P)$
- $\neg P$
- $(P \rightarrow Q)$
- $(P \vee P)$
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$ is a clause
- $((P \vee \neg Q) \wedge P)$ is neither literal nor clause nor monomial
- $\neg P$
- $(P \rightarrow Q)$
- $(P \vee P)$
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$ is a clause
- $((P \vee \neg Q) \wedge P)$ is neither literal nor clause nor monomial
- $\neg P$ is a literal, a clause and a monomial
- $(P \rightarrow Q)$
- $(P \vee P)$
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$ is a clause
- $((P \vee \neg Q) \wedge P)$ is neither literal nor clause nor monomial
- $\neg P$ is a literal, a clause and a monomial
- $(P \rightarrow Q)$ is neither literal nor clause nor monomial
  (but $(\neg P \vee Q)$ is a clause!)
- $(P \vee P)$
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$ is a clause
- $((P \vee \neg Q) \wedge P)$ is neither literal nor clause nor monomial
- $\neg P$ is a literal, a clause and a monomial
- $(P \rightarrow Q)$ is neither literal nor clause nor monomial
  (but $(\neg P \vee Q)$ is a clause!)
- $(P \vee P)$ is a clause, but not a literal or monomial
- $\neg\neg P$

Terminology: Examples

Examples

- $(\neg Q \wedge R)$ is a monomial
- $(P \vee \neg Q)$ is a clause
- $((P \vee \neg Q) \wedge P)$ is neither literal nor clause nor monomial
- $\neg P$ is a literal, a clause and a monomial
- $(P \rightarrow Q)$ is neither literal nor clause nor monomial
  (but $(\neg P \vee Q)$ is a clause!)
- $(P \vee P)$ is a clause, but not a literal or monomial
- $\neg\neg P$ 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

$((\neg P \lor Q) \land R \land (P \lor \neg S))$ 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

$((\neg P \land Q) \lor R \lor (P \land \neg S))$ is in DNF.

CNF and DNF: Examples

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

- $((P \lor \neg Q) \land P)$
- $((R \lor Q) \land P \land (R \lor S))$
- $(P \lor (\neg Q \land R))$
- $((P \lor \neg Q) \rightarrow P)$
- 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 $\vee$, $\wedge$, $\neg$
2. Move negations inside using De Morgan and double negation.
   $\rightsquigarrow$ formula structure: only $\vee$, $\wedge$, literals
3. Distribute $\vee$ over $\wedge$ 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 $\wedge$ over $\vee$ instead.

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

Constructing CNF: Example

Construction of Conjunctive Normal Form

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

$$\varphi \equiv (\neg((P \land \neg Q) \lor R) \lor P \lor \neg(S \lor T)) \qquad \text{[Step 1]}$$

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

Constructing CNF: Example

Construction of Conjunctive Normal Form

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

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

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

Constructing CNF: Example

Construction of Conjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

Construct DNF: Example

Construction of Disjunctive Normal Form

Given: $\varphi = (((P \wedge \neg Q) \vee R) \rightarrow (P \vee \neg(S \vee T)))$

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

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.

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".

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.

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 $(x_1 \vee y_1) \wedge \cdots \wedge (x_n \vee y_n)$ there is no smaller logically equivalent formula in DNF than:

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

- 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

Discrete Mathematics in Computer Science

Knowledge Bases

Malte Helmert, Gabriele Röger

University of Basel

Knowledge Bases: Example

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

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

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: $\mathcal{I} \models \mathrm{KB}$)
if $\mathcal{I}$ is a model for every formula $\varphi \in \mathrm{KB}$.

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 $\mathsf{KB} = \{(\mathsf{A} \wedge \neg \mathsf{B}), \neg(\mathsf{B} \vee \mathsf{A})\}$ have?

Example I

Which of the properties does $\mathsf{KB} = \{(\mathsf{A} \land \neg \mathsf{B}), \neg(\mathsf{B} \lor \mathsf{A})\}$ have?

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

Example I

Which of the properties does $\mathsf{KB} = \{(\mathsf{A} \land \neg \mathsf{B}), \neg(\mathsf{B} \lor \mathsf{A})\}$ have?

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

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

Example II

Which of the properties does

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

Example II

Which of the properties does

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

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

Discrete Mathematics in Computer Science

Logical Consequences

Malte Helmert, Gabriele Röger

University of Basel

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

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. $\mathcal{I} \models \varphi$.

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 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": KB $\models \varphi$ vs. $\mathcal{I} \models \varphi$.

What if KB is unsatisfiable or the empty set?

Logical Consequences: Example

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

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

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

Logical Consequences: Example

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

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

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

Proof sketch.

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

Important Theorems about Logical Consequences

Theorem (Deduction Theorem)

$\mathsf{KB} \cup \{\varphi\} \models \psi$ *iff* $\mathsf{KB} \models (\varphi \rightarrow \psi)$

German: Deduktionssatz

Theorem (Contraposition Theorem)

$\mathsf{KB} \cup \{\varphi\} \models \neg\psi$ *iff* $\mathsf{KB} \cup \{\psi\} \models \neg\varphi$

German: Kontrapositionssatz

Theorem (Contradiction Theorem)

$\mathsf{KB} \cup \{\varphi\}$ *is unsatisfiable iff* $\mathsf{KB} \models \neg\varphi$

German: Widerlegungssatz

(without proof)