Theory of Computer Science

B2. Propositional Logic II

Gabriele Röger

University of Basel

February 24, 2020

Theory of Computer Science

February 24, 2020 — B2. Propositional Logic II

B2.1 Story so far

B2.2 Properties of Propositional Formulas

B2.3 Equivalences

B2.4 Simplified Notation

B2.5 Normal Forms

B2.6 Knowledge Bases

B2.7 Summary

B2.1 Story so far

Reminder: Syntax of Propositional Logic

Definition (Syntax of Propositional Logic)

Let A be a set of atomic propositions. The set of propositional formulas (over A) is inductively defined as follows:

- Every atom $a \in A$ is a propositional formula over A.
- If φ is a propositional formula over A, then so is its negation $\neg\varphi$.
- If φ and ψ are propositional formulas over A, then so is the conjunction $(\varphi \land \psi)$.
- If φ and ψ are propositional formulas over A, then so is the disjunction $(\varphi \lor \psi)$.

The implication $(\varphi \rightarrow \psi)$ is an abbreviation for $(\neg\varphi \lor \psi)$.
The biconditional $(\varphi \leftrightarrow \psi)$ is an abbrev. for $((\varphi \rightarrow \psi) \land (\psi \rightarrow \varphi))$.

Reminder: Semantics of Propositional Logic

Definition (Semantics of Propositional Logic)

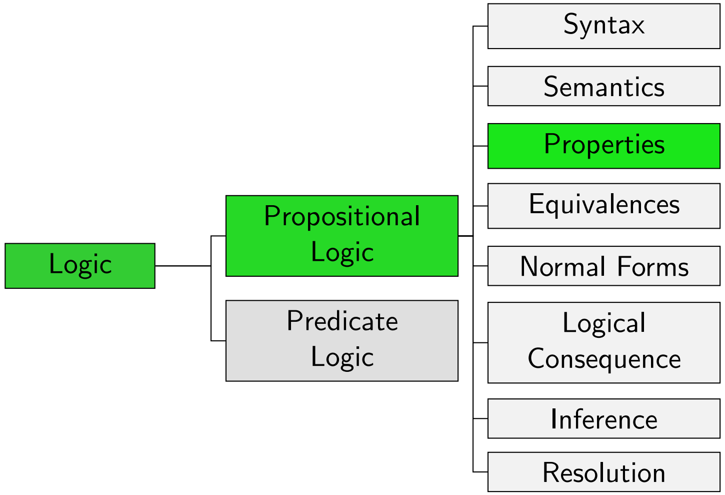
A truth assignment (or interpretation) for a set of atomic propositions A is a function $\mathcal{I} : A \to \{0, 1\}$.

A propositional formula φ (over A) holds under $\mathcal{I}$ (written as $\mathcal{I} \models \varphi$) according to the following definition:

$$\begin{array}{lll} \mathcal{I} \models a & \text{iff} & \mathcal{I}(a) = 1 \qquad\qquad (\text{for } a \in A) \\ \mathcal{I} \models \neg\varphi & \text{iff} & \text{not } \mathcal{I} \models \varphi \\ \mathcal{I} \models (\varphi \land \psi) & \text{iff} & \mathcal{I} \models \varphi \text{ and } \mathcal{I} \models \psi \\ \mathcal{I} \models (\varphi \lor \psi) & \text{iff} & \mathcal{I} \models \varphi \text{ or } \mathcal{I} \models \psi \end{array}$$

B2.2 Properties of Propositional Formulas

Logic: Overview



Properties of Propositional Formulas

A propositional formula φ is

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

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

Exercise

Which properties do the following formulas have?
Satisfiable? Unsatisfiable? Valid? Falsifiable?

- $(A \wedge \neg A)$
- $(A \vee \neg A)$
- $(A \wedge (\neg B \vee C))$
- $((A \wedge \neg B) \vee (\neg A \wedge B))$

Examples

How can we show that a formula has one of these properties?

- Show that $(\mathrm{A} \wedge \mathrm{B})$ is satisfiable.
  $\mathcal{I} = \{\mathrm{A} \mapsto 1, \mathrm{B} \mapsto 1\}$ (+ simple proof that $\mathcal{I} \models (\mathrm{A} \wedge \mathrm{B})$)
- Show that $(\mathrm{A} \wedge \mathrm{B})$ is falsifiable.
  $\mathcal{I} = \{\mathrm{A} \mapsto 0, \mathrm{B} \mapsto 1\}$ (+ simple proof that $\mathcal{I} \not\models (\mathrm{A} \wedge \mathrm{B})$)
- Show that $(\mathrm{A} \wedge \mathrm{B})$ is not valid.
  Follows directly from falsifiability.
- Show that $(\mathrm{A} \wedge \mathrm{B})$ is not unsatisfiable.
  Follows directly from satisfiability.

So far all proofs by specifying one interpretation.

How to prove that a given formula is valid/unsatisfiable/
not satisfiable/not falsifiable?

$\rightsquigarrow$ must consider all possible interpretations

Truth Tables

Evaluate for all possible interpretations
if they are models of the considered formula.

$\mathcal{I}(\mathrm{A})$	$\mathcal{I} \models \neg\mathrm{A}$
0	Yes
1	No

$\mathcal{I}(\mathrm{A})$	$\mathcal{I}(\mathrm{B})$	$\mathcal{I} \models (\mathrm{A} \wedge \mathrm{B})$
0	0	No
0	1	No
1	0	No
1	1	Yes

$\mathcal{I}(\mathrm{A})$	$\mathcal{I}(\mathrm{B})$	$\mathcal{I} \models (\mathrm{A} \vee \mathrm{B})$
0	0	No
0	1	Yes
1	0	Yes
1	1	Yes

Truth Tables in General

Similarly in the case where we consider a formula whose building blocks are themselves arbitrary unspecified formulas:

$\mathcal{I} \models \varphi$	$\mathcal{I} \models \psi$	$\mathcal{I} \models (\varphi \wedge \psi)$
No	No	No
No	Yes	No
Yes	No	No
Yes	Yes	Yes

Exercise

Specify the truth table for $(\varphi \rightarrow \psi)$.

Truth Tables for Properties of Formulas

Is $\varphi = ((A \rightarrow B) \vee (\neg B \rightarrow A))$ valid, unsatisfiable, . . . ?

$\mathcal{I}(A)$	$\mathcal{I}(B)$	$\mathcal{I} \models \neg B$	$\mathcal{I} \models (A \rightarrow B)$	$\mathcal{I} \models (\neg B \rightarrow A)$	$\mathcal{I} \models \varphi$
0	0	Yes	Yes	No	Yes
0	1	No	Yes	Yes	Yes
1	0	Yes	No	Yes	Yes
1	1	No	Yes	Yes	Yes

Connection Between Formula Properties and Truth Tables

A propositional formula φ is

- satisfiable if φ has at least one model
  $\rightsquigarrow$ result in at least one row is "Yes"
- unsatisfiable if φ is not satisfiable
  $\rightsquigarrow$ result in all rows is "No"
- valid (or a tautology) if φ is true under every interpretation
  $\rightsquigarrow$ result in all rows is "Yes"
- falsifiable if φ is no tautology
  $\rightsquigarrow$ result in at least one row is "No"

Main Disadvantage of Truth Tables

How big is a truth table with n atomic propositions?

1	2 interpretations (rows)
2	4 interpretations (rows)
3	8 interpretations (rows)
n	??? interpretations

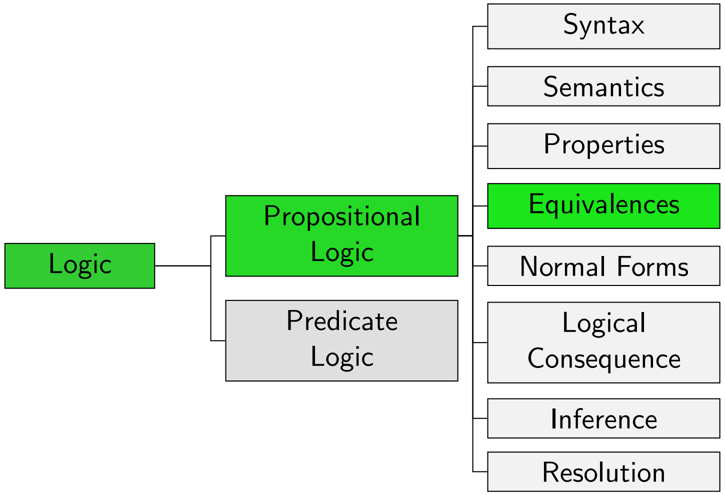
Some examples: $2^{10} = 1024$, $2^{20} = 1048576$, $2^{30} = 1073741824$

$\rightsquigarrow$ not viable for larger formulas; we need a different solution

- more on difficulty of satisfiability etc.: Part E of this course
- practical algorithms: Foundations of AI course

B2.3 Equivalences

Logic: Overview


Logic

Propositional Logic

Predicate Logic

Syntax

Semantics

Properties

Equivalences

Normal Forms

Logical Consequence

Inference

Resolution

Equivalent Formulas

Definition (Equivalence of Propositional Formulas)

Two propositional formulas φ and ψ over A are (logically) equivalent ($\varphi \equiv \psi$) if for all interpretations $\mathcal{I}$ for A it is true that $\mathcal{I} \models \varphi$ if and only if $\mathcal{I} \models \psi$.

German: logisch äquivalent

Equivalent Formulas: Example

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

$\mathcal{I} \models$	$\mathcal{I} \models$	$\mathcal{I} \models$	$\mathcal{I} \models$	$\mathcal{I} \models$	$\mathcal{I} \models$	$\mathcal{I} \models$
φ	ψ	χ	$(\varphi \lor \psi)$	$(\psi \lor \chi)$	$((\varphi \lor \psi) \lor \chi)$	$(\varphi \lor (\psi \lor \chi))$
No	No	No	No	No	No	No
No	No	Yes	No	Yes	Yes	Yes
No	Yes	No	Yes	Yes	Yes	Yes
No	Yes	Yes	Yes	Yes	Yes	Yes
Yes	No	No	Yes	No	Yes	Yes
Yes	No	Yes	Yes	Yes	Yes	Yes
Yes	Yes	No	Yes	Yes	Yes	Yes
Yes	Yes	Yes	Yes	Yes	Yes	Yes

Some Equivalences (1)

$$
\begin{aligned}
(\varphi \land \varphi) &\equiv \varphi \\
(\varphi \lor \varphi) &\equiv \varphi && \text{(idempotence)} \\[4pt]
(\varphi \land \psi) &\equiv (\psi \land \varphi) \\
(\varphi \lor \psi) &\equiv (\psi \lor \varphi) && \text{(commutativity)} \\[4pt]
((\varphi \land \psi) \land \chi) &\equiv (\varphi \land (\psi \land \chi)) \\
((\varphi \lor \psi) \lor \chi) &\equiv (\varphi \lor (\psi \lor \chi)) && \text{(associativity)}
\end{aligned}
$$

German: Idempotenz, Kommutativität, Assoziativität

Some Equivalences (2)

$$
\begin{aligned}
(\varphi \land (\varphi \lor \psi)) &\equiv \varphi \\
(\varphi \lor (\varphi \land \psi)) &\equiv \varphi && \text{(absorption)} \\[4pt]
(\varphi \land (\psi \lor \chi)) &\equiv ((\varphi \land \psi) \lor (\varphi \land \chi)) \\
(\varphi \lor (\psi \land \chi)) &\equiv ((\varphi \lor \psi) \land (\varphi \lor \chi)) && \text{(distributivity)}
\end{aligned}
$$

German: Absorption, Distributivität

Some Equivalences (3)

$$
\begin{aligned}
\lnot\lnot\varphi &\equiv \varphi && \text{(Double negation)} \\[4pt]
\lnot(\varphi \land \psi) &\equiv (\lnot\varphi \lor \lnot\psi) \\
\lnot(\varphi \lor \psi) &\equiv (\lnot\varphi \land \lnot\psi) && \text{(De Morgan's rules)} \\[4pt]
(\varphi \lor \psi) &\equiv \varphi \text{ if } \varphi \text{ tautology} \\
(\varphi \land \psi) &\equiv \psi \text{ if } \varphi \text{ tautology} && \text{(tautology rules)} \\[4pt]
(\varphi \lor \psi) &\equiv \psi \text{ if } \varphi \text{ unsatisfiable} \\
(\varphi \land \psi) &\equiv \varphi \text{ if } \varphi \text{ unsatisfiable} && \text{(unsatisfiability rules)}
\end{aligned}
$$

German: Doppelnegation, De Morgansche Regeln, Tautologieregeln, Unerfüllbarkeitsregeln

Substitution Theorem

Theorem (Substitution Theorem)

Let φ and φ' be equivalent propositional formulas over A. Let ψ be a propositional formula with (at least) one occurrence of the subformula φ.

Then ψ is equivalent to ψ', where ψ' is constructed from ψ by replacing an occurrence of φ in ψ with φ'.

German: Ersetzbarkeitstheorem

(without proof)

Application of Equivalences: Example

$$\begin{aligned}(P \land (Q \lor \neg P)) &\equiv ((P \land Q) \lor (P \land \neg P)) && \text{(distributivity)}\\ &\equiv ((P \land \neg P) \lor (P \land Q)) && \text{(commutativity)}\\ &\equiv (P \land Q) && \text{(unsatisfiability rule)}\end{aligned}$$

Exercise

Use the equivalence rules to show that $(A \lor \neg(B \lor \neg A)) \equiv A$.

B2.4 Simplified Notation

Parentheses

Associativity:

$$\begin{aligned}((\varphi \land \psi) \land \chi) &\equiv (\varphi \land (\psi \land \chi))\\ ((\varphi \lor \psi) \lor \chi) &\equiv (\varphi \lor (\psi \lor \chi))\end{aligned}$$

- 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:

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

$$\sum_{x \in \{x_1, \dots, x_n\}} x = x_1 + x_2 + \cdots + x_n$$

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 $X = \{\varphi_1, \dots, \varphi_n\}$

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 $X = \{\chi\}$?

convention:

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

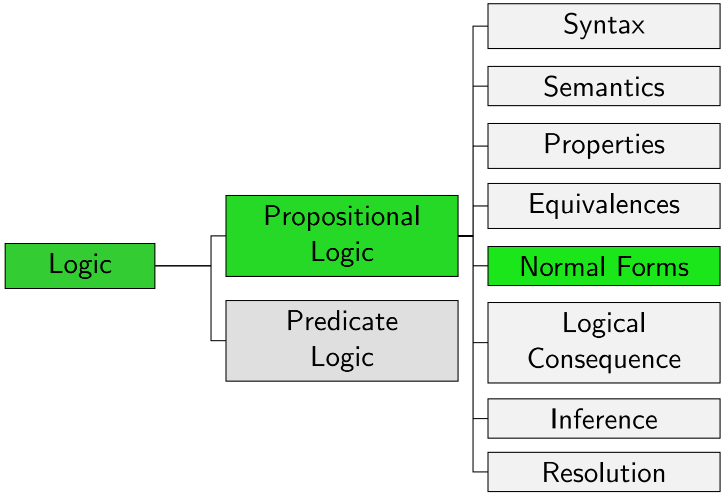
⇝ Why?

Exercise

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

B2.5 Normal Forms

Logic: Overview



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 ¬A).
- A clause is a disjunction of literals (e. g., (Q ∨ ¬P ∨ ¬S ∨ R)).
- A monomial is a conjunction of literals (e. g., (Q ∧ ¬P ∧ ¬S ∧ 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

$$\left(\bigwedge_{i=1}^{n}\left(\bigvee_{j=1}^{m_i} L_{ij}\right)\right)$$

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

$$\left(\bigvee_{i=1}^{n}\left(\bigwedge_{j=1}^{m_i} L_{ij}\right)\right)$$

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?

- $((P \lor \neg Q) \land P)$
- $((R \lor Q) \land P \land (R \lor S))$
- $(P \lor (\neg Q \land R))$
- $((P \lor \neg Q) \rightarrow P)$
- 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 $\lor$, $\land$, $\neg$
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

B2.6 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.

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

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 $\mathrm{KB} = \{(\mathrm{A} \land \neg \mathrm{B}), \neg(\mathrm{B} \lor \mathrm{A})\}$ have?

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

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

Example II

Which of the properties does

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

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

Motivation for next lecture

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? ▷ logical consequences

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

B2.7 Summary

Summary

- satisfiability and validity are important properties of formulas and knowledge bases.
- truth tables systematically consider all possible interpretations
- truth tables are only useful for small formulas
- Logical equivalence describes when formulas are semantically indistinguishable.
- Equivalence rewriting is used to simplify formulas and to bring them in normal forms.
- CNF: formula is a conjunction of clauses
- DNF: formula is a disjunction of monomials
- every formula has equivalent formulas in DNF and in CNF