

Logic (master program)

Georg Moser

Institute of Computer Science @ UIBK

Winter 2008



Definition

- let \mathcal{S} be the set of atomic formulas
- let $\mathcal{F}(\mathcal{S})$ be the set of all formulas over \mathcal{S}
- an **assignment of \mathcal{S}** is a function $\mathcal{A}: \mathcal{S} \rightarrow \{0, 1\}$

assignment

Definition

- if $\mathcal{A}(F) = 1$ then **\mathcal{A} models F**
- we write $\mathcal{A} \models F$

models

Definition

- a formula is **valid** if it holds under every assignment
- if F is valid, we write $\models F$, F is a **tautology**

valid

Definition

- a formula is **satisfiable** if \exists assignment that satisfies it
- otherwise, a formula is **unsatisfiable**
- an unsatisfiable formula is a **contradiction**

satisfiable

Summary of Last Lecture

Primitive operators

\neg $>$ \wedge

Definition

formula formation

- the formula $\neg F$ is the **negation** of F
- the formula $F \wedge G$ is the **conjunction** of F and G

Definition

subformula

- any formula is a **subformula** of itself
- any **subformula** of F is a **subformula** of $\neg F$
- any **subformula** of F and G is a **subformula** of $(F \wedge G)$

Convention

the formula F and (F) are treated interchangeably

Definition

consequence

- G is **consequence** of F , if \forall assignments \mathcal{A} , when $\mathcal{A} \models F$, then $\mathcal{A} \models G$
- we write $F \models G$

Lemma

G is consequence of F if and only if $F \rightarrow G$ is a tautology

Definition

let $\mathcal{F} = \{F_1, F_2, \dots\}$ be a set of formulas

- \mathcal{A} **models \mathcal{F}** ($\mathcal{A} \models \mathcal{F}$), if $\forall F: \mathcal{A} \models F$
- G is a **consequence** of \mathcal{F} ($\mathcal{F} \models G$), if $\forall \mathcal{A}: \mathcal{A} \models \mathcal{F}$ implies $\mathcal{A} \models G$

Content

introduction, **propositional logic**, semantics, **formal proofs**, resolution (propositional)

first-order logic, semantics, structures, theories and models, formal proofs, Herbrand theory, resolution (first-order), completeness of first-order logic, properties of first-order logic

introduction to computability, introduction to complexity, finite model theory

beyond first order: modal logics in a general setting, higher-order logics, introduction to Isabelle

Some Rules

premise	conclusion	name
$G \in \mathcal{F}$	$\mathcal{F} \vdash G$	assumption
$\mathcal{F} \vdash G \wedge \mathcal{F} \subset \mathcal{F}'$	$\mathcal{F}' \vdash G$	monotonicity
$\mathcal{F} \vdash G$	$\mathcal{F} \vdash \neg\neg G$	double negation
$\mathcal{F} \vdash F, \mathcal{F} \vdash G$	$\mathcal{F} \vdash (F \wedge G)$	\wedge -introduction
$\mathcal{F} \vdash (F \wedge G)$	$\mathcal{F} \vdash F$	\wedge -elimination
$\mathcal{F} \vdash (F \wedge G)$	$\mathcal{F} \vdash (G \wedge F)$	\wedge -symmetry
$\mathcal{F} \vdash F$	$\mathcal{F} \vdash F \vee G$	\vee -introduction
$\mathcal{F} \vdash (F \vee G), \mathcal{F} \cup \{F\} \vdash H, \mathcal{F} \cup \{G\} \vdash H$	$\mathcal{F} \vdash H$	\vee -elimination
$\mathcal{F} \vdash (F \vee G)$	$\mathcal{F} \vdash (G \vee F)$	\vee -symmetry
$\mathcal{F} \cup \{F\} \vdash G$	$\mathcal{F} \vdash (F \rightarrow G)$	\rightarrow -introduction
$\mathcal{F} \vdash (F \rightarrow G), \mathcal{F} \vdash F$	$\mathcal{F} \vdash G$	\rightarrow -elimination

Definition

proof system

- a **proof system** consists of rules for derivations
- which allow to **deduce**, step by step, formulas from set of formulas
- the list of steps is called **formal proof**

Soundness

if a formula G is derivable from a set of formulas \mathcal{F} , then G is a consequence of \mathcal{F}

Completeness

if a formula G is a consequence of a set of formulas \mathcal{F} , then G is derivable from \mathcal{F}

Decidability

\exists an (efficient) procedure that decides whether G is derivable from \mathcal{F}

More Rules

premise	conclusion	name
$\mathcal{F} \vdash F$	$\mathcal{F} \vdash (F)$	$()$ -introduction
$\mathcal{F} \vdash (F)$	$\mathcal{F} \vdash F$	$()$ -elimination
$\mathcal{F} \vdash ((F \wedge G) \wedge H)$	$\mathcal{F} \vdash (F \wedge G \wedge H)$	\wedge -parentheses
$\mathcal{F} \vdash ((F \vee G) \vee H)$	$\mathcal{F} \vdash (F \vee G \vee H)$	\vee -parentheses
$\mathcal{F} \vdash (F \vee G)$	$\mathcal{F} \vdash \neg(\neg F \wedge \neg G)$	\vee -definition
$\mathcal{F} \vdash \neg(\neg F \wedge \neg G)$	$\mathcal{F} \vdash (F \vee G)$	
$\mathcal{F} \vdash (F \rightarrow G)$	$\mathcal{F} \vdash (\neg F \vee G)$	\rightarrow -definition
$\mathcal{F} \vdash (\neg F \vee G)$	$\mathcal{F} \vdash (F \rightarrow G)$	
$\mathcal{F} \vdash (F \leftrightarrow G)$	$\mathcal{F} \vdash (F \rightarrow G) \wedge (G \rightarrow F)$	\leftrightarrow -definition
$\mathcal{F} \vdash (F \rightarrow G) \wedge (G \rightarrow F)$	$\mathcal{F} \vdash (F \leftrightarrow G)$	

Derived Rules

Definition

formal proof

- a **formal proof** is a **finite** sequence of statements of the form $\mathcal{F} \vdash G$
- each statement follows from previous ones, by the stated rules
- we say G is **derived** from \mathcal{F} if there is a formal proof of $\mathcal{F} \vdash G$

Example

 \forall -modus ponens

we show the derivability of the following rule

$$\frac{\mathcal{F} \vdash F \quad \mathcal{F} \vdash (\neg F \vee G)}{\mathcal{F} \vdash G}$$

- | | |
|---|----------------------------|
| 1: $\mathcal{F} \vdash F$ | assumption |
| 2: $\mathcal{F} \vdash (\neg F \vee G)$ | assumption |
| 3: $\mathcal{F} \vdash (F \rightarrow G)$ | \rightarrow -definition |
| 4: $\mathcal{F} \vdash G$ | \rightarrow -elimination |

Example

let $\mathcal{H} = \{(\neg A \vee B), (\neg A \vee C), (A \vee \neg D)\}$, derivation of

$$D \rightarrow (A \wedge B \wedge C)$$

from \mathcal{H} (on blackboard)

Example

derived rules

$$\frac{}{\mathcal{F} \vdash (\neg G \vee G)}$$

tautology

$$\frac{\mathcal{F} \vdash (\neg F \wedge F)}{\mathcal{F} \vdash G}$$

contradiction

$$\frac{\mathcal{F} \cup \{F\} \vdash G}{\mathcal{F} \cup \{\neg G\} \vdash \neg F}$$

contraposition

$$\frac{\mathcal{F} \cup \{F\} \vdash G \quad \mathcal{F} \cup \{\neg F\} \vdash G}{\mathcal{F} \vdash G}$$

proof by cases

Soundness Theorem

Theorem

if a formula G is derivable from a set of formulas \mathcal{F} , then G is a consequence of \mathcal{F}

Proof

premise

- $G \in \mathcal{F}$
- $\mathcal{F} \models G \wedge \mathcal{F} \subset \mathcal{F}'$
- $\mathcal{F} \models G$
- $\mathcal{F} \models F, \mathcal{F} \models G$
- $\mathcal{F} \models (F \wedge G)$
- $\mathcal{F} \models F$
- $\mathcal{F} \models (F \vee G), \mathcal{F} \cup \{F\} \models H, \mathcal{F} \cup \{G\} \models H$
- $\mathcal{F} \models (F \vee G)$

conclusion

- $\mathcal{F} \models G$
- $\mathcal{F}' \models G$
- $\mathcal{F} \models \neg\neg G$
- $\mathcal{F} \models (F \wedge G)$
- $\mathcal{F} \models F$
- $\mathcal{F} \models F \vee G$
- $\mathcal{F} \models H$
- $\mathcal{F} \models (G \vee F)$

name

- assumption
- monotonicity
- double neg
- \wedge -introduction
- \wedge -elimination
- \vee -introduction
- \vee -elimination
- \vee -symmetry



Corollary

if G is derived from \emptyset , then G is tautology

Proof

- if $\emptyset \vdash G$, then $\mathcal{F} \vdash G$ by monotonicity
- by soundness $\mathcal{F} \models G$ for any set of formulas \mathcal{F}
- $(F \vee \neg F) \models G$
- $\forall \mathcal{A} (\mathcal{A} \models (F \vee \neg F) \text{ implies } \mathcal{A} \models G)$
- $\forall \mathcal{A} \mathcal{A} \models G$



Corollary

if $\neg G$ can be derived from \emptyset , then G is a contradiction

Corollary

- formulas F and G are **provably equivalent** if $\{F\} \vdash G \wedge \{G\} \vdash F$
- provable equivalent formulas are equivalent

Theorem

- 1 suppose $F \equiv G$
- 2 suppose H contains F as subformula and let $H' = H\{F \mapsto G\}$
- 3 then $H \equiv H'$

Proof

by structural induction on the H

Definition

- a **literal** is an atomic formula or its negation
- a formula F is in **conjunctive normal form (CNF)** if

$$F = \bigwedge_{i=1}^n \left(\bigvee_{j=1}^m L_{ij} \right) \quad L_{ij} \text{ a literal}$$

- a formula F is in **disjunctive normal form (DNF)** if

$$F = \bigvee_{i=1}^n \left(\bigwedge_{j=1}^m L_{ij} \right) \quad L_{ij} \text{ a literal}$$

Resolution

Reminder

compare week 3 @ LICS

Definition

clause

- \square is a **clause**
- literals are **clauses**
- if C, D are clauses, then $C \vee D$ is a **clause**

we use the equivalences $C \vee \square \vee D \equiv C \vee D, \square \vee \square \equiv \square$

Example

let A, B be atomic formulas

$$A \vee B \quad A \vee \neg B \quad A \vee \neg B \vee A \quad A \vee \square$$

Question

is $\{A, \neg B\}$ a clause?

Observation

the set notation of clauses implicitly assumes equivalence under:

$$\underbrace{A \vee A \equiv A}_{\text{idempotency}} \quad \underbrace{A \vee B \equiv B \vee A}_{\text{commutativity}} \quad \underbrace{(A \vee B) \vee C \equiv A \vee (B \vee C)}_{\text{associativity}}$$

Convention

we assert equivalence under **commutativity** and **associativity** and $\neg\neg A \equiv A$ for literals

Definition

$$\frac{C \vee A \quad D \vee \neg A}{\underbrace{C \vee D}_{\text{resolvent}}} \quad \text{(propositional) resolution}$$

Definition

$$\frac{C \vee A \vee A}{\underbrace{C \vee A}_{\text{factor}}} \quad \text{(propositional) factoring}$$

Completeness & Compactness

Lemma

let X be an infinite set of finite binary strings

- \exists an infinite binary string w so that any prefix of w is also a prefix of infinitely many $x \in X$

Theorem

compactness

a set of formulas of propositional logic is satisfiable iff every finite subset is satisfiable

Proof

use the lemma

Theorem

completeness

if a formula G is a consequence of a set of formulas \mathcal{F} , then G is derivable from \mathcal{F}