CS3202: Logic, Specification and Verification

CS3202-LSV 2006–07

cs3202.lec@cs.st-andrews.ac.uk

Dr. James McKinna, Rm 1.03
Dr. Stéphane Lengrand, Rm. 1.02
Lecture 2 (12/02/2007):

Propositional logic —

The notion of proof in Natural Deduction
Review of propositional logic

- A **syntax** to express statements (wwf)
- A **semantics** to interpret the statements in the world \{True, False\} given by *valuations*
Review of propositional logic: Consequence

- First approach: via the semantics
  use the interpretation in the world to determine whether formulae are true or false

  \textit{Tautology}: wff with value true in \textit{all valuations}

Even better: use the interpretation in the world to determine whether a formula $\phi$ is a \textit{logical consequence} from other formulae $\phi_1, \ldots, \phi_n$:
  every model of $\phi_1, \ldots, \phi_n$ is a model of $\phi$

$$\phi_1, \ldots, \phi_n \models \phi$$
Review of propositional logic: Consequence
• Second approach: directly in the **syntax** … *Why?*

design a syntactic relation

\[ \phi_1, \ldots, \phi_n \vdash \phi \]

via the notion of *proof*.

This is defined by a proof-theoretic formalism, e.g. *Natural deduction* (but there are others)

Question: Is \( \phi_1, \ldots, \phi_n \models \phi \) equivalent to \( \phi_1, \ldots, \phi_n \vdash \phi \)?
Natural deduction: the general idea

- Proof-theoretic formalism in which:

  A proof is a (labelled & well-formed) tree.

  - Nodes are labelled with wff. Example:
    \[
    \begin{array}{c}
    b \\
    \hline
    a \\
    \hline
    a \\
    (b \lor c) \\
    \hline
    a \land (b \lor c)
    \end{array}
    \]

  - Internal nodes and subtrees follow rules: *inference rules*. Example:
    \[
    \begin{array}{c}
    A \\
    \hline
    B \\
    \hline
    A \land B
    \end{array}
    \]
Schematic rules and instances

- **Schema:**
  \[
  \begin{array}{c}
  A \\
  B \\
  \hline
  A \land B
  \end{array}
  \]

  where \( A, B \) range over wff

- **Instances (examples):**
  \[
  \begin{array}{cc}
  c & c' \\
  \hline
  \neg c & \neg c' \\
  c \land c' & \neg c \land \neg c'
  \end{array}
  \]

  for the particular atoms \( c \) and \( c' \)
Natural deduction: the syntactic consequence

- Proof-theoretic formalism in which:

  A proof is a (labelled & well-formed) tree.

  We define the relation $\phi_1, \ldots, \phi_n \vdash \phi$ as:

  “there exists a (proof-)tree whose leaves (a.k.a. hypotheses) are labelled with wff among $\phi_1, \ldots, \phi_n$ and whose root (a.k.a. conclusion) is labelled with $\phi$”
Natural deduction: the actual rules for $\land$

- $\land$-introduction:

\[
\begin{array}{c}
A \\
B
\end{array} \quad \begin{array}{c}
A \\
B
\end{array} \\
\hline
A \land B
\end{array}
\]

- $\land$-elimination:

\[
\begin{array}{c}
A \land B
\end{array} \quad \begin{array}{c}
A \land B
\end{array} \\
\hline
A \\
B
\end{array}
\]
Natural deduction: the actual rules for $\Rightarrow$

- $\Rightarrow$-introduction:

\[
\frac{[A] \quad \cdot \quad \cdot \quad \cdot \quad B}{B} \quad A \Rightarrow B
\]

$A$ is discharged.

$\Rightarrow$-elimination (a.k.a. Modus Ponens):

\[
\frac{A \Rightarrow B \quad A}{B}
\]
Natural deduction: the actual rules for $\Rightarrow$

- Need to adapt the notion of proof:
  - Labelled & well-formed tree + subset of leaves (active leaves).
  - Discharge $A = \text{remove from the set some leaves of the subtree labelled with } A$

- We define the relation $\phi_1, \ldots, \phi_n \vdash \phi$ as:
  - “there exists a (proof-)tree whose active leaves are labelled with wff among $\phi_1, \ldots, \phi_n$ and whose root is labelled with $\phi$”
Natural deduction: the actual rules for $\bot$, $\neg$ and $\lor$

- $\bot$-introduction: none

\[ [A] \quad \bot \quad \neg A \]

- $\neg$-introduction:

\[ \bot \quad \neg A \]

- $\lor$-introduction:

\[ A \quad B \quad A \lor B \quad A \lor B \]

- $\bot$-elimination:

\[ A \]

- $\neg$-elimination:

\[ A \quad \neg A \quad \bot \]

- $\lor$-elimination:

\[ [A] \quad [B] \quad \vdots \quad \vdots \quad A \lor B \quad C \quad C \quad C \]

\[ C \]
Natural deduction: Soundness

• $\phi_1, \ldots, \phi_n \vdash \phi$ implies $\phi_1, \ldots, \phi_n \models \phi$?

We prove it by induction on the height of tree. The inductive step amounts to analysing whether each inference rule is correct.

• Later: a lecture on induction.

(structural) induction as the reasoning counterpart to function definition by (structural) recursion
Natural deduction: Completeness

- $\phi_1, \ldots, \phi_n \models \phi$ implies $\phi_1, \ldots, \phi_n \vdash \phi$?

Are the rules enough to characterise semantic consequence?

We shall see tomorrow.
Questions?