COMP 4161

NICTA Advanced Course

Advanced Topics in Software Verification

Gerwin Klein, June Andronick, Toby Murray

$\lambda^{\rightarrow}$ and **HOL**

Exercises from last time

NICTA

- Construct a type derivation tree for the term $\lambda x\ y\ z.\ z\ x\ (y\ x)$
- Find a unifier (substitution) such that $\lambda x\ y\ z.\ ?F\ y\ z = \lambda x\ y\ z.\ z\ (?G\ x\ y)$

Content

NICTA

Rough timeline

➜ Intro & motivation, getting started [1]

➜ Foundations & Principles

- Lambda Calculus, natural deduction [2,3,4[a]]
- Higher Order Logic [5,6[b],7]
- Term rewriting [8,9,10[c]]

➜ Proof & Specification Techniques

- Isar [11,12[d]]
- Inductively defined sets, rule induction [13[e],15]
- Datatypes, recursion, induction [16,17[f],18,19]
- Calculational reasoning, mathematics style proofs [20]
- Hoare logic, proofs about programs [21[g],22,23]

[a] a1 out; [b] a1 due; [c] a2 out; [d] a2 due; [e] session break; [f] a3 out; [g] a3 due

PREVIEW: PROOFS IN ISABELLE

General schema:

lemma name: ”<goal>”
apply <method>
apply <method>
. . .
done

➜ Sequential application of methods until all **subgoals** are solved.

The Proof State

1. $\bigwedge x_1 \dots x_p . [\![A_1; \dots ; A_n]\!] \Longrightarrow B$

2. $\bigwedge y_1 \dots y_q . [\![C_1; \dots ; C_m]\!] \Longrightarrow D$

$x_1 \dots x_p$	Parameters
$A_1 \dots A_n$	Local assumptions
B	Actual (sub)goal

Isabelle Theories

Syntax:

```
theory MyTh
imports ImpTh_1 ... ImpTh_n
begin
(declarations, definitions, theorems, proofs, ...)*
end
```

- $MyTh$: name of theory. Must live in file $MyTh$`.thy`
- $ImpTh_i$: name of *imported* theories. Import transitive.

Unless you need something special:

`theory` $MyTh$ `imports Main begin` $\ldots$ `end`

Natural Deduction Rules

$$\frac{A \quad B}{A \wedge B}\ \text{conjI} \qquad \frac{A \wedge B \quad [\![A; B]\!] \Longrightarrow C}{C}\ \text{conjE}$$

$$\frac{A}{A \vee B}\ \frac{B}{A \vee B}\ \text{disjI1/2} \qquad \frac{A \vee B \quad A \Longrightarrow C \quad B \Longrightarrow C}{C}\ \text{disjE}$$

$$\frac{A \Longrightarrow B}{A \longrightarrow B}\ \text{impI} \qquad \frac{A \longrightarrow B \quad A \quad B \Longrightarrow C}{C}\ \text{impE}$$

For each connective ($\wedge, \vee$, etc):

introduction and **elemination** rules

Proof by assumption

apply assumption

proves

1. $[\![B_1; \ldots; B_m]\!] \Longrightarrow C$

by unifying C with one of the B_i

There may be more than one matching B_i and multiple unifiers.

Backtracking!

Explicit backtracking command: **back**

Intro rules

Intro rules decompose formulae to the right of $\Longrightarrow$.

apply (rule $<$intro-rule$>$)

Intro rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

➜ To prove A it suffices to show $A_1 \ldots A_n$

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:

➜ unify A and C

➜ replace C with n new subgoals $A_1 \ldots A_n$

Elim rules

Elim rules decompose formulae on the left of $\Longrightarrow$.

apply (erule <elim-rule>)

Elim rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ means

- ➜ If I know A_1 and want to prove A it suffices to show $A_2 \ldots A_n$

Applying rule $[\![A_1; \ldots; A_n]\!] \Longrightarrow A$ to subgoal C:
Like **rule** but also

- ➜ unifies first premise of rule with an assumption
- ➜ eliminates that assumption

DEMO

MORE PROOF RULES

Iff, Negation, True and False

$$\frac{A \Longrightarrow B \quad B \Longrightarrow A}{A = B}\ \text{iffI} \qquad \frac{A = B \quad [\![A \longrightarrow B; B \longrightarrow A]\!] \Longrightarrow C}{C}\ \text{iffE}$$

$$\frac{A = B}{A \Longrightarrow B}\ \text{iffD1} \qquad \frac{A = B}{B \Longrightarrow A}\ \text{iffD2}$$

$$\frac{A \Longrightarrow False}{\neg A}\ \text{notI} \qquad \frac{\neg A \quad A}{P}\ \text{notE}$$

$$\frac{}{\mathit{True}}\ \text{TrueI} \qquad \frac{\mathit{False}}{P}\ \text{FalseE}$$

Equality

$$\frac{}{t = t}\ \text{refl} \qquad \frac{s = t}{t = s}\ \text{sym} \qquad \frac{r = s \quad s = t}{r = t}\ \text{trans}$$

$$\frac{s = t \quad P\ s}{P\ t}\ \text{subst}$$

Rarely needed explicitly — used implicitly by term rewriting

Classical

$$\frac{}{P = True \vee P = False} \text{ True-False}$$

$$\frac{}{P \vee \neg P} \text{ excluded-middle}$$

$$\frac{\neg A \Longrightarrow False}{A} \text{ ccontr} \qquad \frac{\neg A \Longrightarrow A}{A} \text{ classical}$$

- **excluded-middle**, **ccontr** and **classical** not derivable from the other rules.
- if we include True-False, they are derivable

They make the logic "classical", "non-constructive"

Cases

$$\frac{}{P \vee \neg P}\ \text{excluded-middle}$$

is a case distinction on type $bool$

Isabelle can do case distinctions on arbitrary terms:

apply (case_tac $term$)

Safe and not so safe

Safe rules preserve provability

conjI, impI, notI, iffi, refl, ccontr, classical, conjE, disjE

$$\frac{A \quad B}{A \wedge B} \text{ conjI}$$

Unsafe rules can turn a provable goal into an unprovable one

disjI1, disjI2, impE, iffD1, iffD2, notE

$$\frac{A}{A \vee B} \text{ disjI1}$$

Apply safe rules before unsafe ones

DEMO

What we have learned so far...

- natural deduction rules for $\wedge$, $\vee$, $\longrightarrow$, $\neg$, iff...
- proof by assumption, by intro rule, elim rule
- safe and unsafe rules

Exercises

NICTA

- ➜ Redo the demo alone + exercises
- ➜ Assignement 1 is out today!

- ➜ Reminder: DO NOT CHEAT
  - Assignments and exams are take-home. This does NOT mean you can work in groups. Each submission is personal.
  - For more info, see Plagiarism Policy