COMP 4161

NICTA Advanced Course

Advanced Topics in Software Verification

Simon Winwood, Toby Murray, June Andronick, Gerwin Klein

$\{\}$

Slide 1

Content

- ➜ Intro & motivation, getting started with Isabelle
- ➜ Foundations & Principles
  - Lambda Calculus
  - Higher Order Logic, natural deduction
  - Term rewriting
- ➜ **Proof & Specification Techniques**
  - **Inductively defined sets, rule induction**
  - Datatypes, recursion, induction
  - Calculational reasoning, mathematics style proofs
  - Hoare logic, proofs about programs

Slide 2

Last Time

- ➜ More confluence
- ➜ Knuth-Bendix Algorithm, Waldmeister
- ➜ More Isar: forward, backward, obtain, abbreviations, moreover
- ➜ Specification techniques: Sets

Slide 3

Inductive Definitions

Slide 4

Example

$$\frac{}{\langle \mathsf{skip}, \sigma\rangle \longrightarrow \sigma} \qquad \frac{[\![e]\!]\sigma = v}{\langle \mathsf{x := e}, \sigma\rangle \longrightarrow \sigma[x \mapsto v]}$$

$$\frac{\langle c_1, \sigma\rangle \longrightarrow \sigma' \quad \langle c_2, \sigma'\rangle \longrightarrow \sigma''}{\langle c_1; c_2, \sigma\rangle \longrightarrow \sigma''}$$

$$\frac{[\![b]\!]\sigma = \mathsf{False}}{\langle \mathsf{while}\ b\ \mathsf{do}\ c, \sigma\rangle \longrightarrow \sigma}$$

$$\frac{[\![b]\!]\sigma = \mathsf{True} \quad \langle c, \sigma\rangle \longrightarrow \sigma' \quad \langle \mathsf{while}\ b\ \mathsf{do}\ c, \sigma'\rangle \longrightarrow \sigma''}{\langle \mathsf{while}\ b\ \mathsf{do}\ c, \sigma\rangle \longrightarrow \sigma''}$$

Slide 5

What does this mean?

- ➜ $\langle c, \sigma\rangle \longrightarrow \sigma'$ fancy syntax for a relation $(c, \sigma, \sigma') \in E$
- ➜ relations are sets: $E :: (\mathsf{com} \times \mathsf{state} \times \mathsf{state})\ \mathsf{set}$
- ➜ the rules define a set inductively

But which set?

Slide 6

Simpler Example

$$\frac{}{0 \in N} \qquad \frac{n \in N}{n + 1 \in N}$$

- ➜ N is the set of natural numbers $\mathbb{N}$
- ➜ But why not the set of real numbers? $0 \in \mathbb{R}$, $n \in \mathbb{R} \Longrightarrow n + 1 \in \mathbb{R}$
- ➜ $\mathbb{N}$ is the **smallest** set that is **consistent** with the rules.

Why the smallest set?

- ➜ Objective: **no junk**. Only what must be in X shall be in X.
- ➜ Gives rise to a nice proof principle (rule induction)
- ➜ Alternative (greatest set) occasionally also useful: coinduction

Slide 7

Formally

Rules $\dfrac{a_1 \in X \quad \dots \quad a_n \in X}{a \in X}$ with $a_1, \dots, a_n, a \in A$

define set $X \subseteq A$

Formally: set of rules $R \subseteq A\ \mathsf{set} \times A$ $\quad(R, X$ possibly infinite)

Applying rules R to a set B: $\quad \hat{R}\ B \equiv \{x.\ \exists H.\ (H, x) \in R \wedge H \subseteq B\}$

Example:

$$\begin{array}{lcl} R & \equiv & \{(\{\}, 0)\} \cup \{(\{n\}, n+1).\ n \in \mathbb{R}\} \\ \hat{R}\ \{3, 6, 10\} & = & \{0, 4, 7, 11\} \end{array}$$

Slide 8

The Set

Definition: B is R-closed iff $\hat{R}\ B \subseteq B$

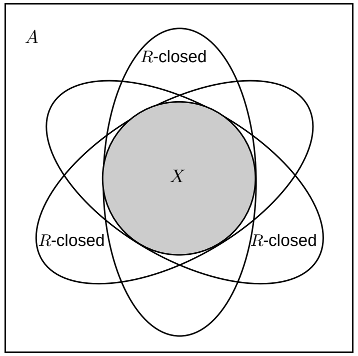
Definition: X is the least R-closed subset of A

This does always exist:

Fact: $X = \bigcap \{B \subseteq A.\ B\ R\text{–closed}\}$

Slide 9

Generation from Above



Slide 10

Rule Induction

$$\frac{}{0 \in N} \qquad \frac{n \in N}{n+1 \in N}$$

induces induction principle

$$[\![P\ 0;\ \bigwedge n.\ P\ n \Longrightarrow P\ (n+1)]\!] \Longrightarrow \forall x \in X.\ P\ x$$

In general:

$$\frac{\forall (\{a_1, \ldots a_n\}, a) \in R.\ P\ a_1 \wedge \ldots \wedge P\ a_n \Longrightarrow P\ a}{\forall x \in X.\ P\ x}$$

Slide 11

Why does this work?

$$\frac{\forall (\{a_1, \ldots a_n\}, a) \in R.\ P\ a_1 \wedge \ldots \wedge P\ a_n \Longrightarrow P\ a}{\forall x \in X.\ P\ x}$$

$\forall (\{a_1, \ldots a_n\}, a) \in R.\ P\ a_1 \wedge \ldots \wedge P\ a_n \Longrightarrow P\ a$
says
$\{x.\ P\ x\}$ is R-closed

but: X is the least R-closed set
hence: $X \subseteq \{x.\ P\ x\}$
which means: $\forall x \in X.\ P\ x$

qed

Slide 12

Rules with side conditions

$$\frac{a_1 \in X \quad \dots \quad a_n \in X \qquad C_1 \quad \dots \quad C_m}{a \in X}$$

induction scheme:

$$(\forall (\{a_1, \dots a_n\}, a) \in R.\ P\ a_1 \wedge \dots \wedge P\ a_n \wedge$$
$$C_1 \wedge \dots \wedge C_m \wedge$$
$$\{a_1, \dots, a_n\} \subseteq X \Longrightarrow P\ a)$$

$$\Longrightarrow$$

$$\forall x \in X.\ P\ x$$

Slide 13

X as Fixpoint

How to compute X?
$X = \bigcap\{B \subseteq A.\ B\ R-\text{closed}\}$ hard to work with.
Instead: view X as least fixpoint, X least set with $\hat{R}\ X = X$.

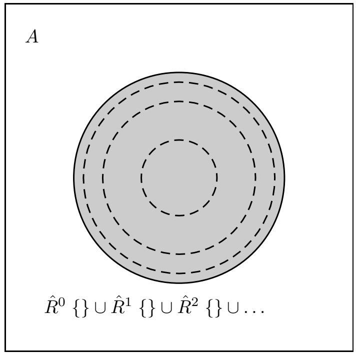
Fixpoints can be approximated by iteration:

$$X_0 = \hat{R}^0\ \{\} = \{\}$$
$$X_1 = \hat{R}^1\ \{\} = \text{rules without hypotheses}$$
$$\vdots$$
$$X_n = \hat{R}^n\ \{\}$$

$$X_\omega = \bigcup_{n\in\mathbb{N}} (R^n\ \{\}) = X$$

Slide 14

Generation from Below



Slide 15

DEMO: INDUCTIVE DEFINITONS

Slide 16

We have seen today ...

NICTA

- ➜ Sets in Isabelle
- ➜ Inductive Definitions
- ➜ Rule induction
- ➜ Fixpoints

Slide 17