NICTA Advanced Course

Theorem Proving

Principles, Techniques, Applications

$$\{\}$$

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

LAST TIME

➜ Conditional term rewriting

LAST TIME

- ➜ Conditional term rewriting
- ➜ Congruence and AC rules

LAST TIME

- ➜ Conditional term rewriting
- ➜ Congruence and AC rules
- ➜ More on confluence

LAST TIME

- ➜ Conditional term rewriting
- ➜ Congruence and AC rules
- ➜ More on confluence
- ➜ Completion

Last Time

- ➜ Conditional term rewriting
- ➜ Congruence and AC rules
- ➜ More on confluence
- ➜ Completion
- ➜ Isar: fix, obtain, abbreviations, moreover, ultimately

SETS IN ISABELLE

Type **’a set**: sets over type ’a

SETS IN ISABELLE

Type **'a set**: sets over type 'a

➜ $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$

Sets in Isabelle

Type **'a set**: sets over type 'a

- $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$
- $e \in A$, $A \subseteq B$

SETS IN ISABELLE

Type **'a set**: sets over type 'a

- $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$
- $e \in A$, $A \subseteq B$
- $A \cup B$, $A \cap B$, $A - B$, $-A$

SETS IN ISABELLE

Type **'a set**: sets over type 'a

- ➜ $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$
- ➜ $e \in A$, $A \subseteq B$
- ➜ $A \cup B$, $A \cap B$, $A - B$, $-A$
- ➜ $\bigcup x \in A.\ B\ x$, $\bigcap x \in A.\ B\ x$, $\bigcap\ A$, $\bigcup A$

Sets in Isabelle

Type **'a set**: sets over type 'a

- $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$
- $e \in A$, $A \subseteq B$
- $A \cup B$, $A \cap B$, $A - B$, $-A$
- $\bigcup x \in A.\ B\ x$, $\bigcap x \in A.\ B\ x$, $\bigcap\ A$, $\bigcup A$
- $\{i..j\}$

Sets in Isabelle

Type **'a set**: sets over type 'a

- ➜ $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$
- ➜ $e \in A$, $A \subseteq B$
- ➜ $A \cup B$, $A \cap B$, $A - B$, $-A$
- ➜ $\bigcup x \in A.\ B\ x$, $\bigcap x \in A.\ B\ x$, $\bigcap\ A$, $\bigcup A$
- ➜ $\{i..j\}$
- ➜ insert :: $\alpha \Rightarrow \alpha$ set $\Rightarrow \alpha$ set

Sets in Isabelle

Type **'a set**: sets over type 'a

- $\{\}$, $\{e_1, \ldots, e_n\}$, $\{x.\ P\ x\}$
- $e \in A$, $A \subseteq B$
- $A \cup B$, $A \cap B$, $A - B$, $-A$
- $\bigcup x \in A.\ B\ x$, $\bigcap x \in A.\ B\ x$, $\bigcap\ A$, $\bigcup A$
- $\{i..j\}$
- insert :: $\alpha \Rightarrow \alpha$ set $\Rightarrow \alpha$ set
- $f`A \equiv \{y.\ \exists x \in A.\ y = f\ x\}$
- $\ldots$

Proofs about Sets

Natural deduction proofs:

- equalityI: $[\![A \subseteq B;\ B \subseteq A]\!] \Longrightarrow A = B$

Proofs about Sets

Natural deduction proofs:

- equalityI: $[\![A \subseteq B;\ B \subseteq A]\!] \Longrightarrow A = B$
- subsetI: $(\bigwedge x.\ x \in A \Longrightarrow x \in B) \Longrightarrow A \subseteq B$

Proofs about Sets

Natural deduction proofs:

- equalityI: $[\![A \subseteq B;\ B \subseteq A]\!] \Longrightarrow A = B$
- subsetI: $(\bigwedge x.\ x \in A \Longrightarrow x \in B) \Longrightarrow A \subseteq B$
- . . .   (see Tutorial)

Bounded Quantifiers

➜ $\forall x \in A.\ P\ x$

Bounded Quantifiers

➜ $\forall x \in A.\ P\ x \equiv \forall x.\ x \in A \longrightarrow P\ x$

Bounded Quantifiers

- $\forall x \in A.\ P\ x \equiv \forall x.\ x \in A \longrightarrow P\ x$
- $\exists x \in A.\ P\ x$

Bounded Quantifiers

- ➜ $\forall x \in A.\ P\ x \equiv \forall x.\ x \in A \longrightarrow P\ x$
- ➜ $\exists x \in A.\ P\ x \equiv \exists x.\ x \in A \wedge P\ x$

Bounded Quantifiers

- $\forall x \in A.\ P\ x \equiv \forall x.\ x \in A \longrightarrow P\ x$
- $\exists x \in A.\ P\ x \equiv \exists x.\ x \in A \wedge P\ x$

- ballI: $(\bigwedge x.\ x \in A \Longrightarrow P\ x) \Longrightarrow \forall x \in A.\ P\ x$
- bspec: $[\![\forall x \in A.\ P\ x; x \in A]\!] \Longrightarrow P\ x$

Bounded Quantifiers

- ➜ $\forall x \in A.\ P\ x \equiv \forall x.\ x \in A \longrightarrow P\ x$
- ➜ $\exists x \in A.\ P\ x \equiv \exists x.\ x \in A \wedge P\ x$

- ➜ ballI: $(\bigwedge x.\ x \in A \Longrightarrow P\ x) \Longrightarrow \forall x \in A.\ P\ x$
- ➜ bspec: $[\![\forall x \in A.\ P\ x; x \in A]\!] \Longrightarrow P\ x$

- ➜ bexI: $[\![P\ x; x \in A]\!] \Longrightarrow \exists x \in A.\ P\ x$
- ➜ bexE: $[\![\exists x \in A.\ P\ x; \bigwedge x.\ [\![x \in A; P\ x]\!] \Longrightarrow Q]\!] \Longrightarrow Q$

Demo: Sets

INDUCTIVE DEFINITIONS

EXAMPLE

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

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

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

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

WHAT DOES THIS MEAN?

WHAT DOES THIS MEAN?

➜ $\langle c, \sigma \rangle \longrightarrow \sigma'$ fancy syntax for a relation $(c, \sigma, \sigma') \in E$

What does this mean?

➜ $\langle c, \sigma \rangle \longrightarrow \sigma'$ fancy syntax for a relation $(c, \sigma, \sigma') \in E$

➜ relations are sets: $E :: (\mathsf{com} \times \mathsf{state} \times \mathsf{state})\ \mathsf{set}$

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

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?

Simpler Example

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

Simpler Example

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

➜ N is the set of natural numbers $\mathbb{N}$

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? $0 \in \mathbb{R}$, $n \in \mathbb{R} \Longrightarrow n + 1 \in \mathbb{R}$

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.

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?

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.

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)

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

Formally

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

define set $X \subseteq A$

Formally:

FORMALLY

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

defi ne set $X \subseteq A$

Formally: set of rules $R \subseteq A$ set $\times A$ $\quad$ (R, X possibly infi nite)

Applying rules R to a set B:

FORMALLY

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

define set $X \subseteq A$

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

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

Example:

FORMALLY

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

defi ne set $X \subseteq A$

Formally: set of rules $R \subseteq A$ set $\times A$ (R, X possibly infi nite)

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

Example:

$$
\begin{aligned}
R \quad &\equiv \quad \{(\{\}, 0)\} \cup \{(\{n\}, n+1).\ n \in \mathbb{R}\} \\
\hat{R}\, \{3, 6, 10\} \quad &=
\end{aligned}
$$

FORMALLY

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

define set $X \subseteq A$

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

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

Example:

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

THE SET

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

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:

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: B_1 R-closed $\wedge$ B_2 R-closed $\Longrightarrow B_1 \cap B_2$ R-closed

THE SET

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

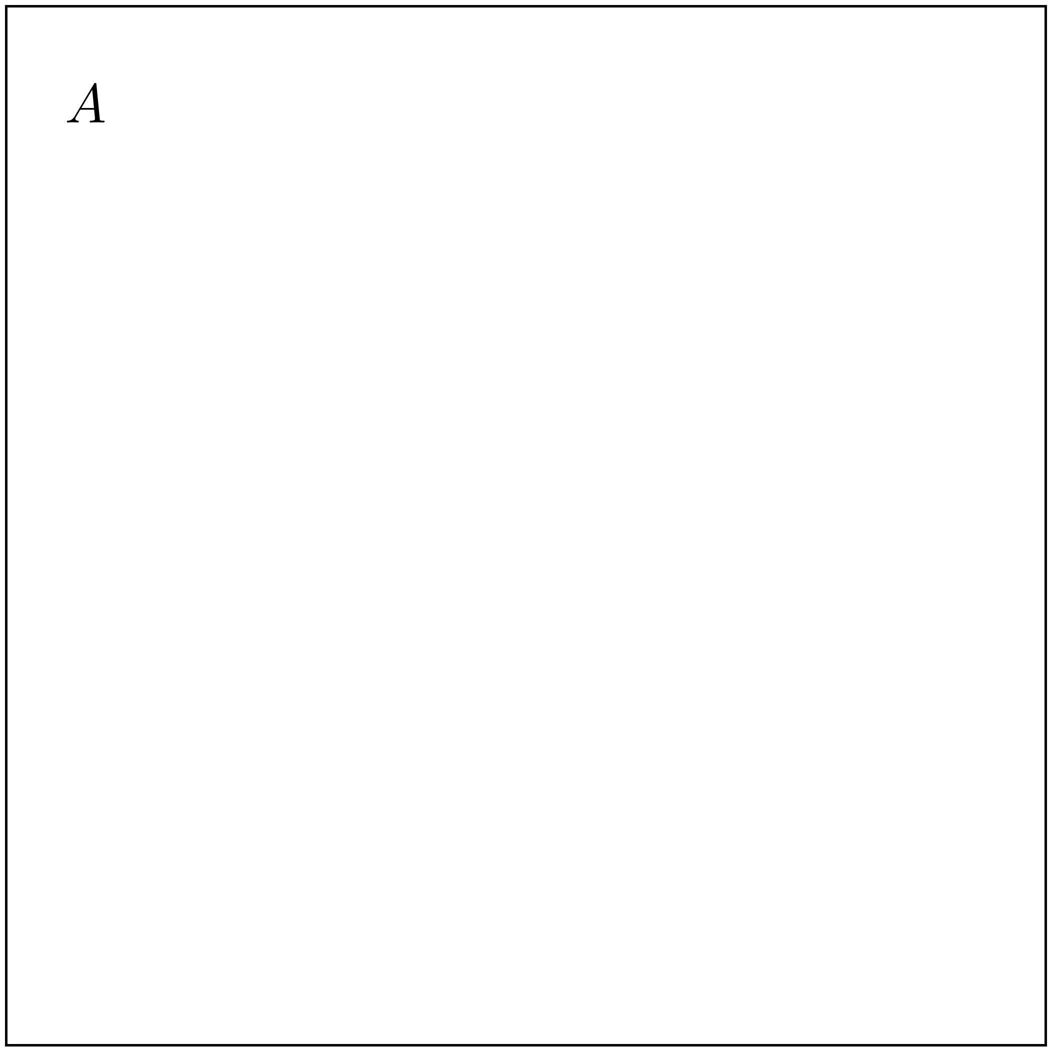
Definition: X is the least R-closed subset of A

This does always exist:

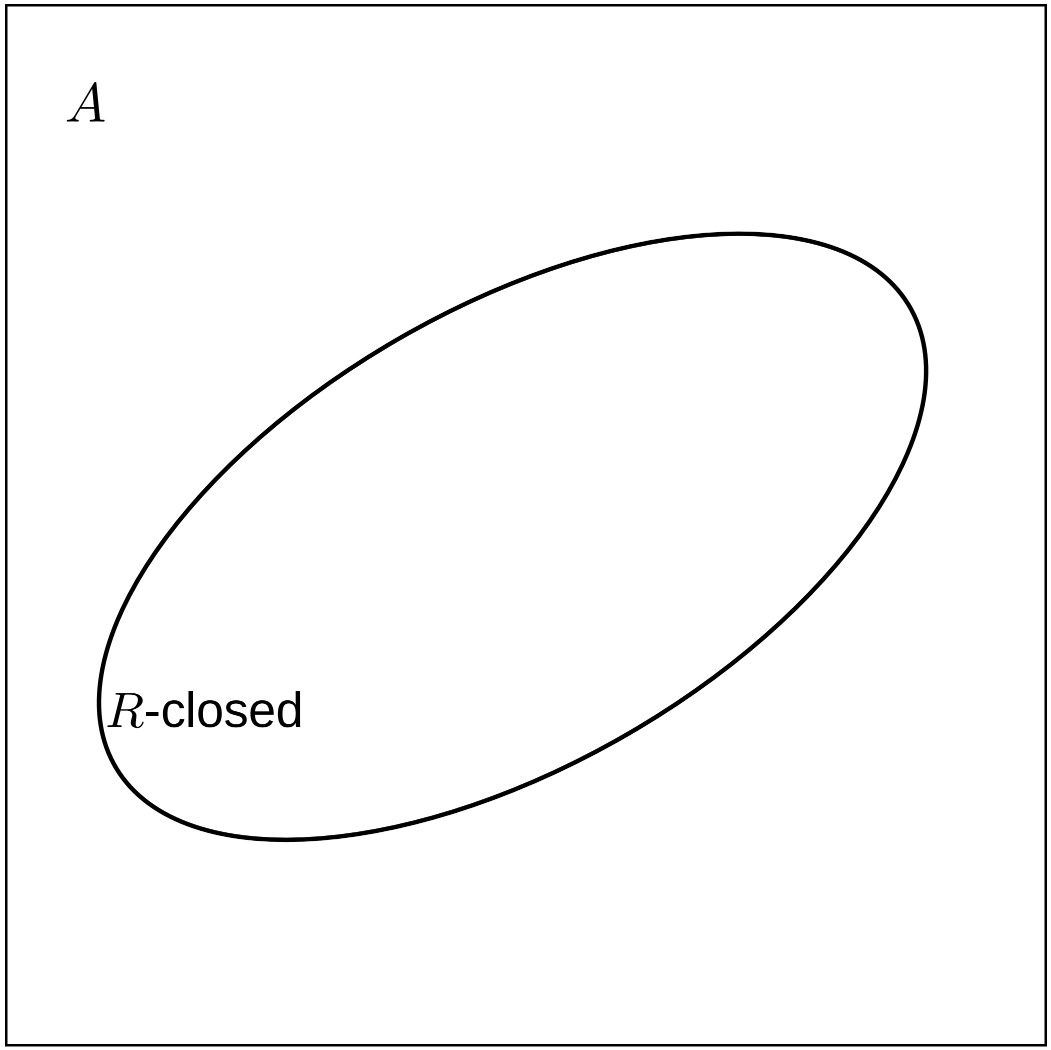
Fact: B_1 R-closed $\wedge$ B_2 R-closed $\Longrightarrow B_1 \cap B_2$ R-closed

Hence: $X = \bigcap\{B \subseteq A.\ B\ R{-}\text{closed}\}$

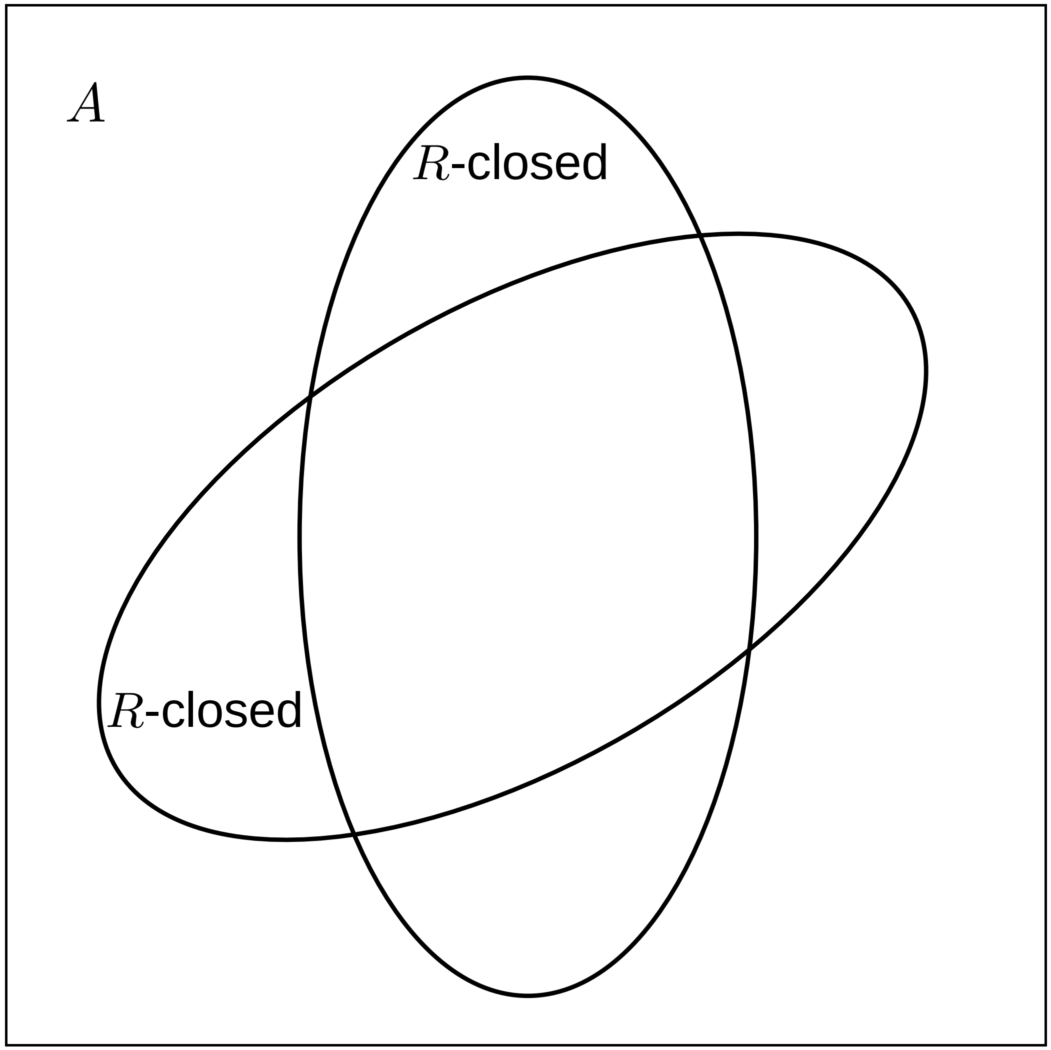
Generation from Above



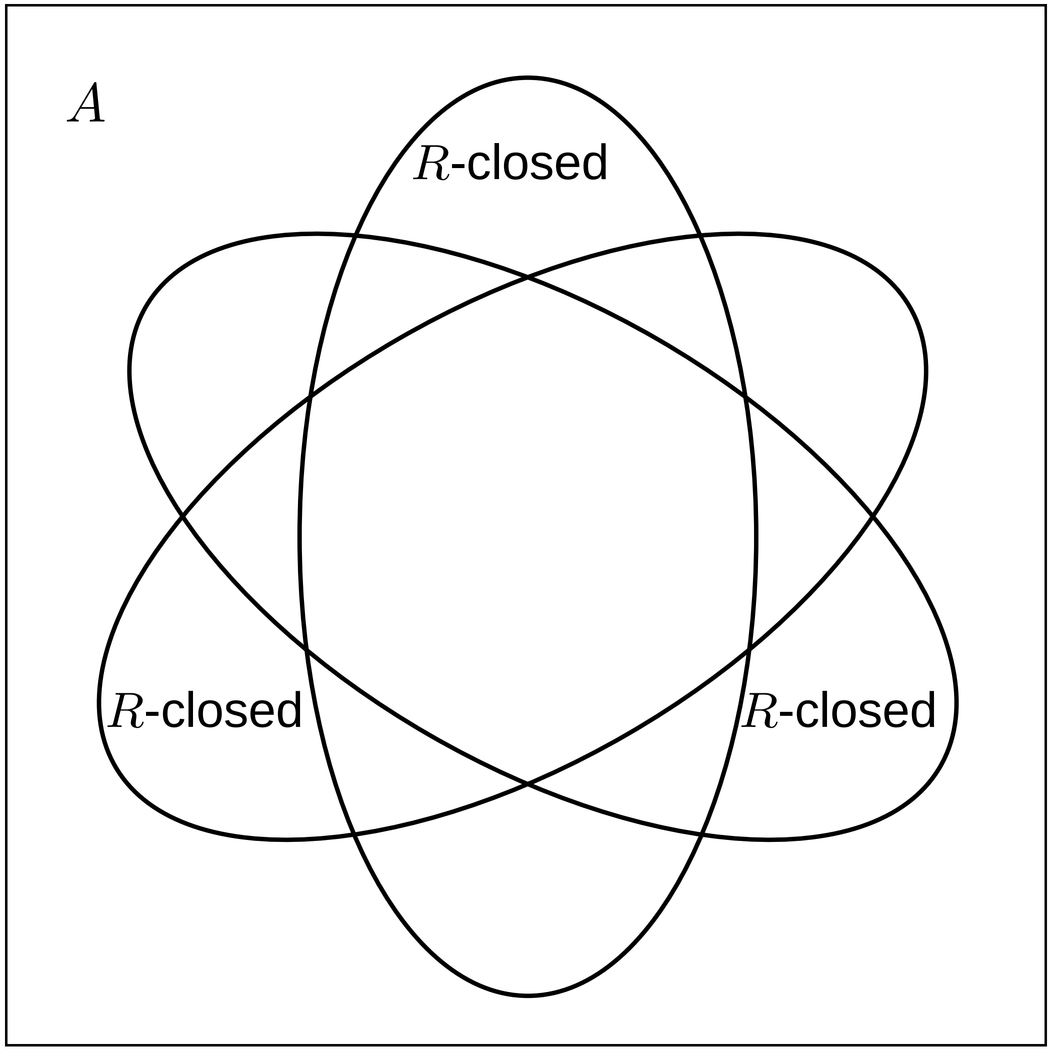
Generation from Above



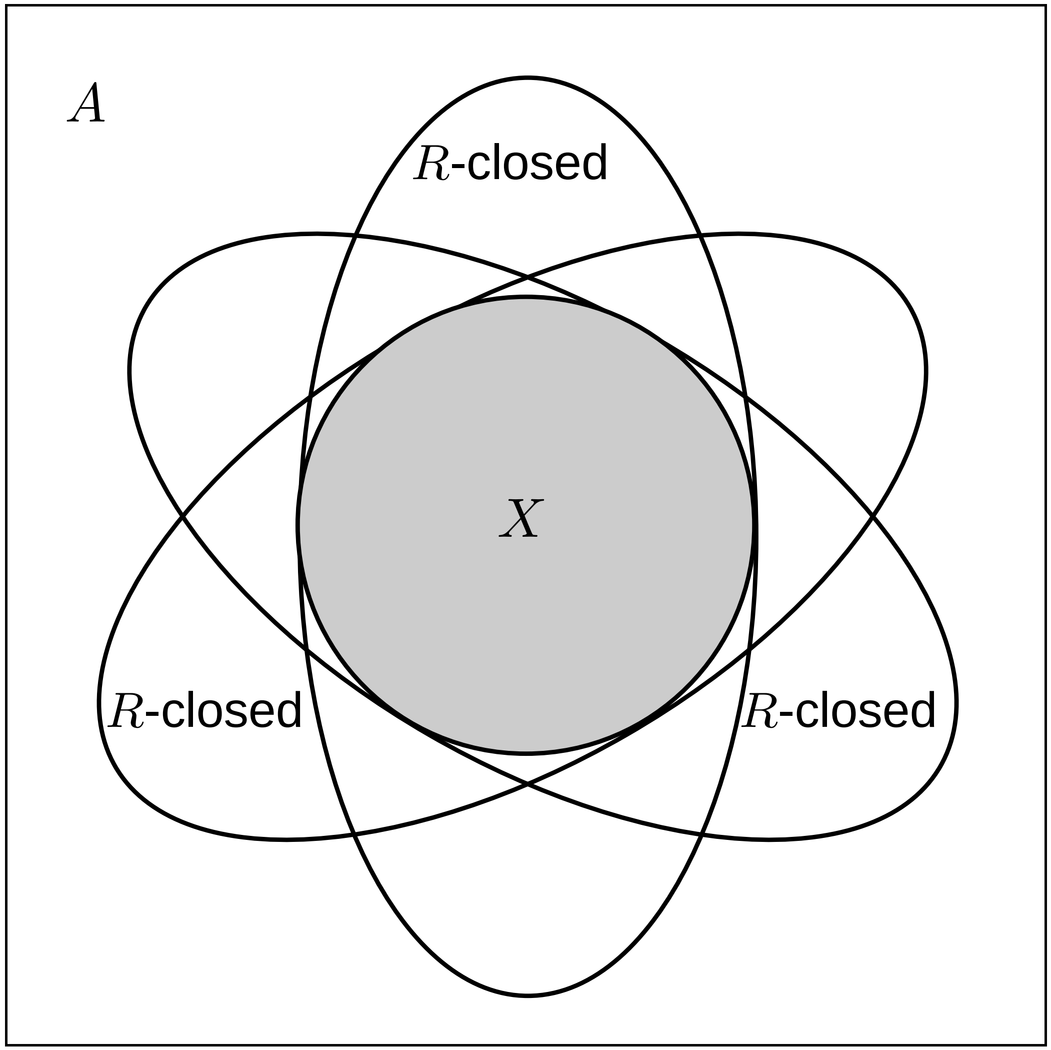
Generation from Above



Generation from Above



Generation from Above



RULE INDUCTION

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

induces induction principle

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

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:

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

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

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

$\{x.\ Px\}$ is R-closed

but:

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

$\{x.\ Px\}$ is R-closed

but: X is the least R-closed set

hence:

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

$\{x.\ Px\}$ is R-closed

but: X is the least R-closed set

hence: $X \subseteq \{x.\ P\ x\}$

which means:

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

$\{x.\ Px\}$ is R-closed

but: X is the least R-closed set

hence: $X \subseteq \{x.\ P\ x\}$

which means: $\forall x \in X.\ P\ x$

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

$\{x.\ Px\}$ 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

Rules with side conditions

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

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:

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

$$\Longrightarrow$$

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

X AS FIXPOINT

How to compute X?

How to compute X?

$X = \bigcap\{B \subseteq A.\ B\ R-\text{closed}\}$ hard to work with.

Instead:

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 fi xpoint, X least set with $\hat{R}\ X = X$.

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 fi xpoint, X least set with $\hat{R}\ X = X$.

Fixpoints can be approximated by iteration:

$$X_0 = \hat{R}^0\ \{\} = \{\}$$

How to compute X?

$X = \bigcap\{B \subseteq A.\ B\ R - \text{closed}\}$ hard to work with.

Instead: view X as least fi xpoint, X least set with $\hat{R}\ X = X$.

Fixpoints can be approximated by iteration:

$$X_0 = \hat{R}^0\ \{\} = \{\}$$

$$X_1 = \hat{R}^1\ \{\} = \text{rules without hypotheses}$$

$$\vdots$$

How to compute X?

$X = \bigcap\{B \subseteq A.\ B\ R-\text{closed}\}$ hard to work with.

Instead: view X as least fi xpoint, 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 AS FIXPOINT

How to compute X?

$X = \bigcap\{B \subseteq A.\ B\ R - \text{closed}\}$ hard to work with.

Instead: view X as least fi xpoint, X least set with $\hat{R}\ X = X$.

Fixpoints can be approximated by iteration:

$$X_0 = \hat{R}^0\ \{\} = \{\}$$

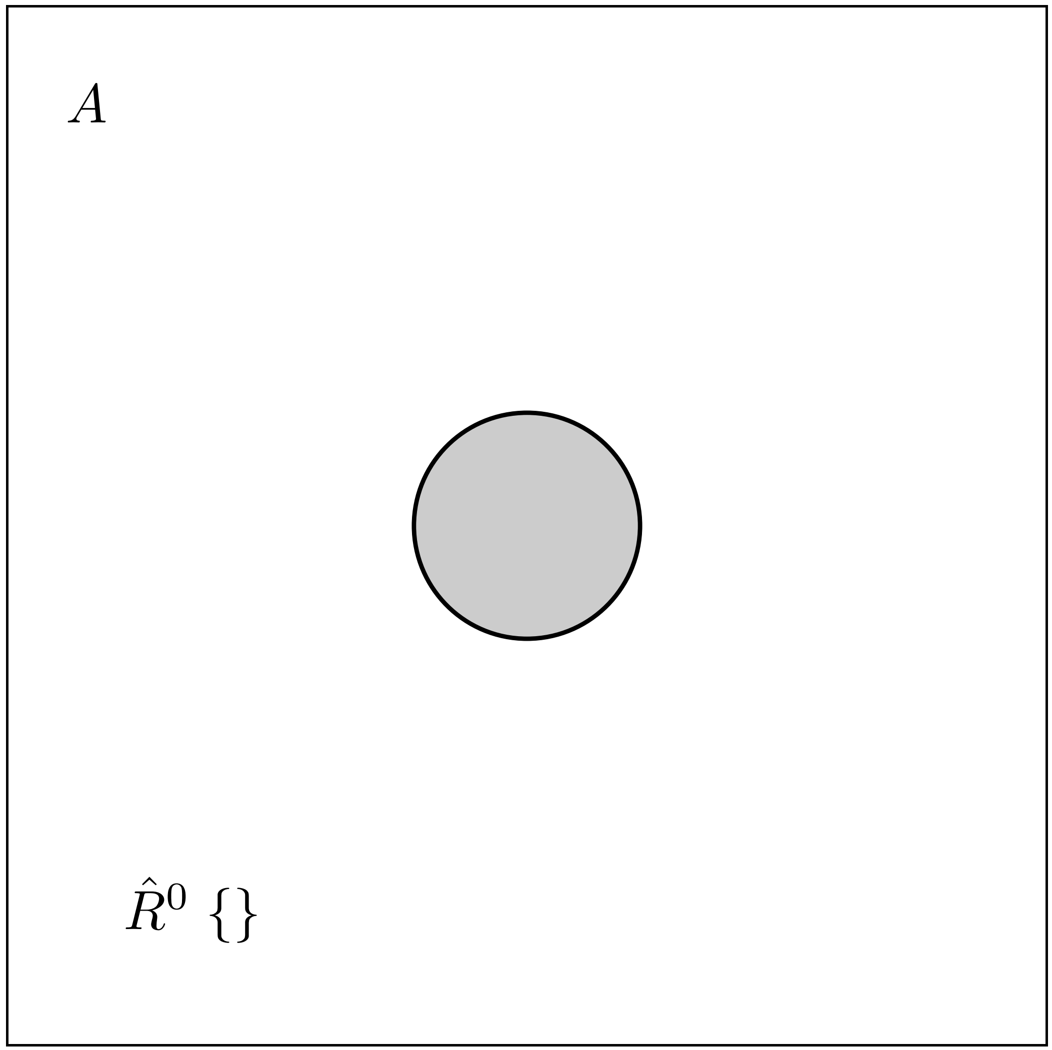
$$X_1 = \hat{R}^1\ \{\} = \text{rules without hypotheses}$$

$$\vdots$$

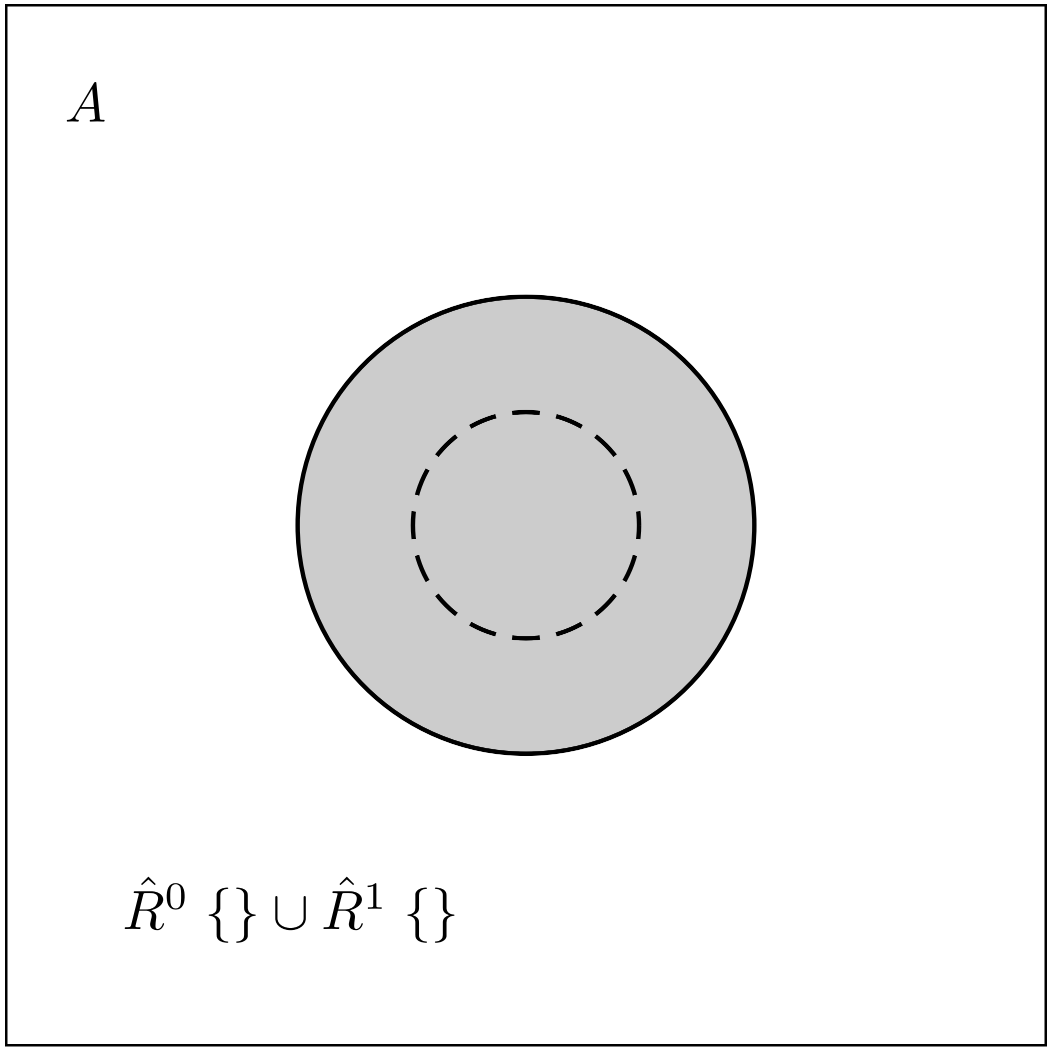
$$X_n = \hat{R}^n\ \{\}$$

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

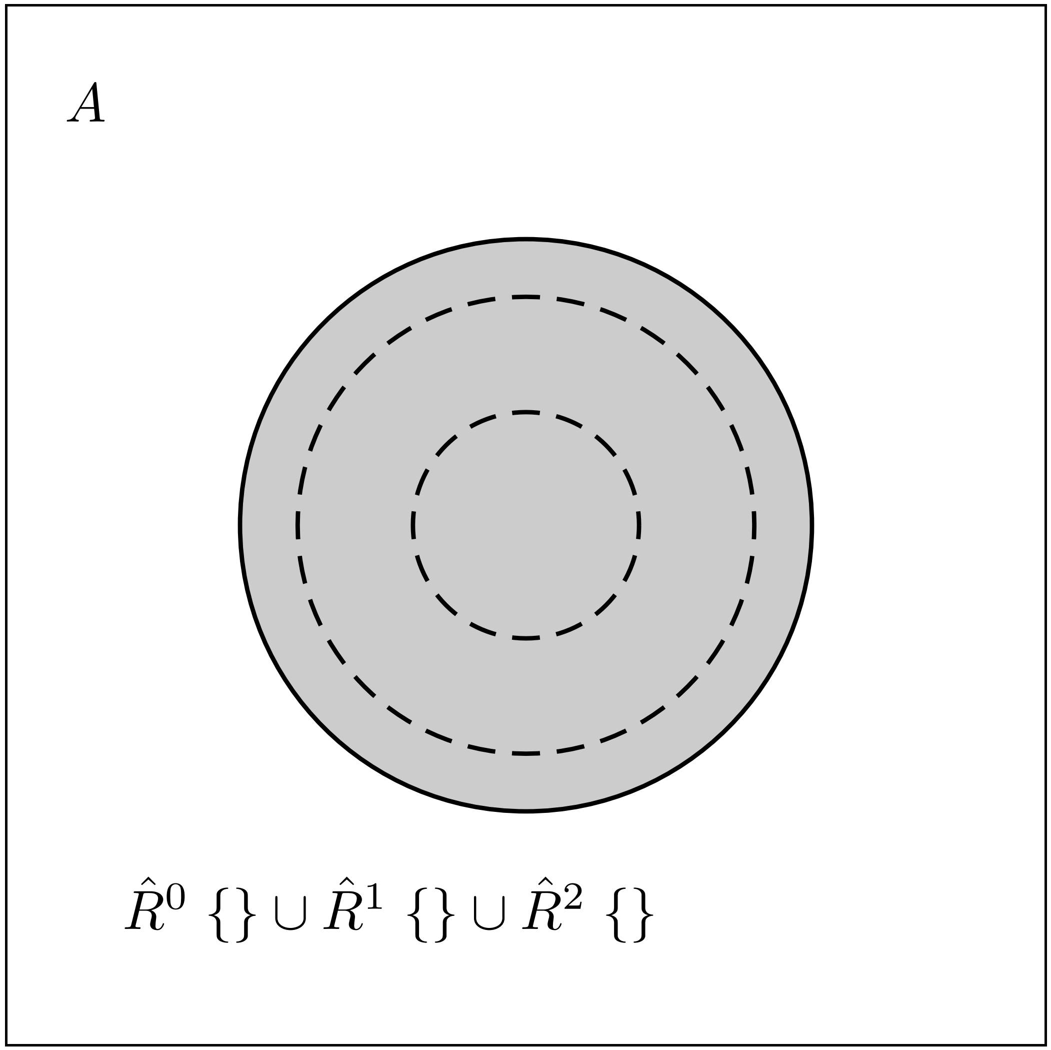
Generation from Below



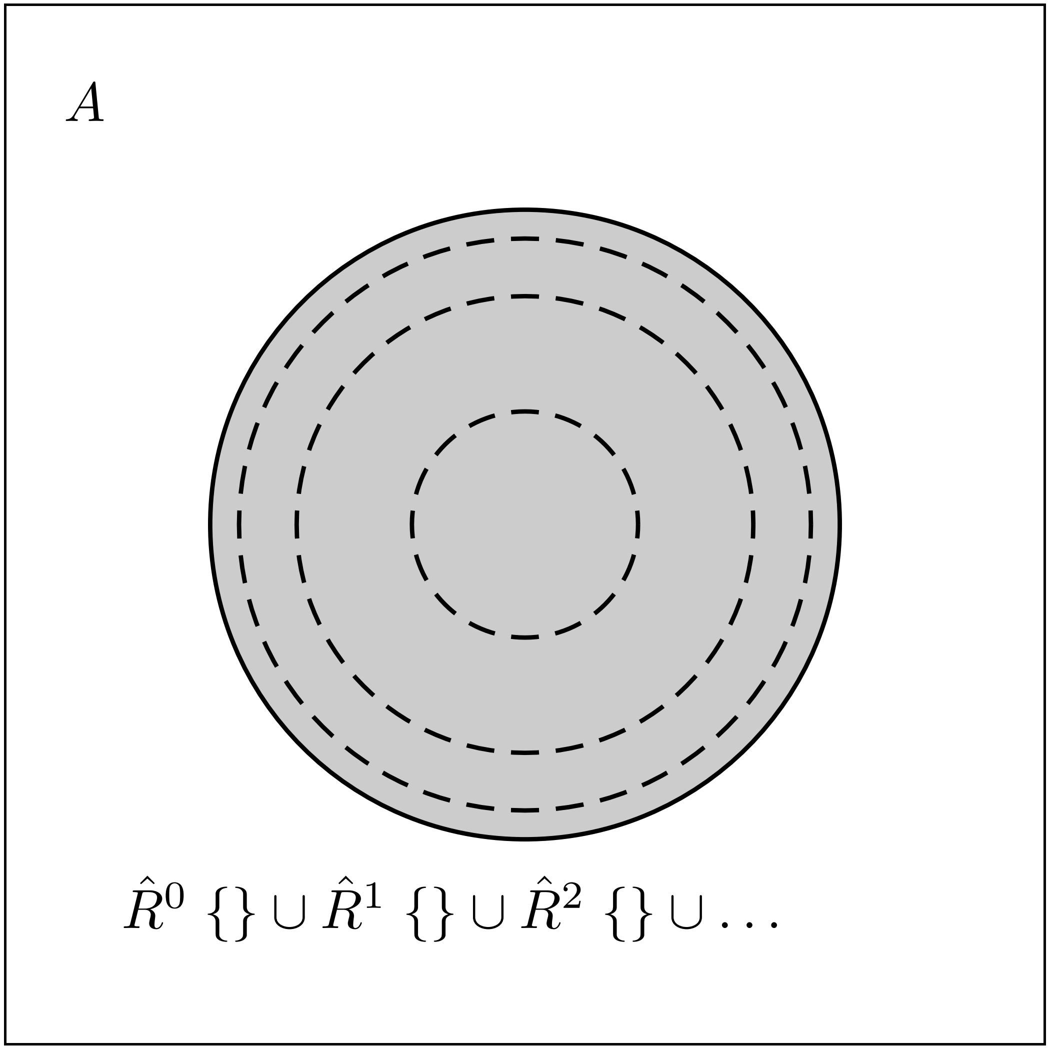
GENERATION FROM BELOW



Generation from Below



Generation from Below



DEMO: INDUCTIVE DEFINITONS

ISAR

INDUCTIVE DEFINITION IN ISABELLE

inductive S

intros

rule$_1$: "$[\![s \in S; A]\!] \Longrightarrow s' \in S$"

$\vdots$

rule$_n$: $\ldots$

Rule induction

show "$x \in S \Longrightarrow P\ x$"

proof (induct rule: S.induct)

  fix s and s' **assume** "$s \in S$" and "A" and "$P\ s$"

  ...

  show "$P\ s'$"

next

⋮

qed

Abbreviations

show "$x \in S \Longrightarrow P\ x$"
proof (induct rule: S.induct)
  case rule$_1$
  . . .
  show ?case
next
⋮
next
  case rule$_n$
  . . .
  show ?case
qed

Implicit selection of induction rule

assume A: "$x \in S$"

⋮

show "$P\ x$"

using A **proof** induct

⋮

qed

IMPLICIT SELECTION OF INDUCTION RULE

assume A: "$x \in S$"

⋮

show "$P\ x$"

using A **proof** induct

⋮

qed

lemma assumes A: "$x \in S$" **shows** "$P\ x$"

using A **proof** induct

⋮

qed

Renaming free variables in rule

case (rule$_i$ $x_1 \ldots x_k$)

Renames first k (alphabetically!) variables in rule$_i$ to $x_1 \ldots x_k$.

A REMARK ON STYLE

➜ **case** (rule$_i$ x y) . . . **show** ?case
is easy to write and maintain

A REMARK ON STYLE

➜ **case** (rule$_i$ x y) . . . **show** ?case
is easy to write and maintain

➜ **fix** x y **assume** $formula$ . . . **show** $formula'$
is easier to read:

- all information is shown locally
- no contextual references (e.g. ?case)

DEMO

We have seen today ...

➜ Sets in Isabelle

We have seen today ...

- ➜ Sets in Isabelle
- ➜ Inductive Definitions

We have seen today ...

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

We have seen today ...

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

We have seen today ...

- ➜ Sets in Isabelle
- ➜ Inductive Definitions
- ➜ Rule induction
- ➜ Fixpoints
- ➜ Isar: induct and cases

EXERCISES

Formalize this lecture in Isabelle:

➜ Define **closed** $f\ A :: (\alpha \text{ set} \Rightarrow \alpha \text{ set}) \Rightarrow \alpha \text{ set} \Rightarrow \text{bool}$

➜ Show closed $f\ A \wedge$ closed $f\ B \Longrightarrow$ closed $f\ (A \cap B)$ if f is monotone (**mono** is predefined)

➜ Define **lfpt** f as the intersection of all f-closed sets

➜ Show that lfpt f is a fixpoint of f if f is monotone

➜ Show that lfpt f is the least fixpoint of f

➜ Declare a constant $R :: (\alpha \text{ set} \times \alpha) \text{ set}$

➜ Define $\hat{R} :: \alpha \text{ set} \Rightarrow \alpha \text{ set}$ in terms of R

➜ Show soundness of rule induction using R and lfpt $\hat{R}$