Proof Systems

1 Proof systems

def: proof system

$$S = (\Sigma, L, A, R)$$

- Σ: alphabet
- $L \subset A^*$ language, decidable
- $A \subset L$ axioms, decidable
- R decidable set of proof rules of the form

$$\frac{w_1, \dots, w_i}{v} \quad \text{with} \quad w_1, \dots, w_i, v \in L$$

intention: if $w_1, \dots, w_i$ are proven it is allowed to conclude v. Example:

$$\frac{A \ , \ A \to B}{B}$$

1 Proof systems

def: proof system

$$S = (\Sigma, L, A, R)$$

- Σ: alphabet
- $L \subset A^*$ language, decidable
- $A \subset L$ axioms, decidable
- R decidable set of proof rules of the form

  $$\frac{w_1, \ldots, w_i}{v} \quad \text{with} \quad w_1, \ldots, w_i, v \in L$$

  intention: if $w_1, \ldots, w_i$ are proven it is allowed to conclude v. Example:

  $$\frac{A \ , \ A \to B}{B}$$

def:proof sequence

$$p = (w_0 \# \ldots \# w_t) \quad (\text{with } \# \notin \Sigma)$$

such that for all i

- $w_i \in A$ or
- $\exists j_1, \ldots, j_n < i. \ \frac{w_{j_1}, \ldots, w_{j_n}}{w_i} \in R$

We say: w is provable/can be derived in S and write

$$S \vdash w_t$$

1 Proof systems

def: proof system

$$S = (\Sigma, L, A, R)$$

- Σ: alphabet
- $L \subset A^*$ language, decidable
- $A \subset L$ axioms, decidable
- R decidable set of proof rules of the form

$$\frac{w_1,\dots,w_i}{v} \quad \text{with} \quad w_1,\dots,w_i,v \in L$$

intention: if $w_1,\dots,w_i$ are proven it is allowed to conclude v. Example:

$$\frac{A \;,\; A \to B}{B}$$

def:proof sequence

$$p = (w_0\#\dots\#w_t) \quad (\text{with } \# \notin \Sigma)$$

such that for all i

- $w_i \in A$ or
- $\exists j_1,\dots,j_n < i.\ \ \frac{w_{j_1},\dots,w_{j_n}}{w_i} \in R$

We say: w is provable/can be derived in S and write

$$S \vdash w_t$$

Lemma 1.
- *the set of proofs in S*

$$\{p \;:\; p \text{ is proof in } S\}$$

is decidable

why?

- *the set of provable strings*

$$\{w \;:\; S \vdash w\}$$

is recursively enumerable.

Proof. enumerate $(\Sigma \cup \{\#\})^*$. For each enumerated string $w_0\#\dots\#w_t$ test if it is a proof; if yes output w_t. □

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

2.1 Syntax

$$\Sigma_E = \{0, 1, v, c, (,), +, \cdot, =, \wedge, \vee, \sim, \rightarrow, \exists, \forall\} \}$$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

2.1 Syntax

$$\Sigma_E = \{0, 1, v, c, (,), +, \cdot, =, \wedge, \vee, \sim, \rightarrow, \exists, \forall\}$$

def: set of variables V

$$V = \{xw \ : \ w \in \mathbb{B}^*\}$$

def: set of constant symbols C

$$C = \{0, 1\} \cup \{cw \ : \ w \in \mathbb{B}^*\}$$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

2.1 Syntax

$$\Sigma_E = \{0,1,v,c,(,),+,\cdot,=,\wedge,\vee,\sim,\rightarrow,\exists,\forall\}$$

def: set of variables V

$$V = \{xw \;:\; w \in \mathbb{B}^*\}$$

def: set of constant symbols C

$$C = \{0,1\} \cup \{cw \;:\; w \in \mathbb{B}^*\}$$

warning: Distinction between variables and constant symbols is technical and subtle.

- variables: can be *free* or *bound by quantifiers* (see below)
- constant symbols:
  1. some have special meaning. Here 0 and 1.
  2. others serve as 'intermediate variables' in proofs. Example: 'Let c, c' be arbitrary but in the sequel fixed integers...' In this way one might prove

     $$c + c' = c' + c$$

     which has no free varaiables, indeed it has no variables at all. But when we are done we hopefully are able to infer

     $$\forall x \forall x' \; x + x' = x' + x$$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

2.1 Syntax

$$\Sigma_E = \{0,1,v,c,(,),+,\cdot,=,\wedge,\vee,\sim,\rightarrow,\exists,\forall\}$$

def: set of variables V

$$V = \{xw \;:\; w \in \mathbb{B}^*\}$$

def: set of constant symbols C

$$C = \{0,1\} \cup \{cw \;:\; w \in \mathbb{B}^*\}$$

warning: Distinction between variables and constant symbols is technical and subtle.

- variables: can be *free* or *bound by quantifiers* (see below)
- constant symbols:
  1. some have special meaning. Here 0 and 1.
  2. others serve as 'intermediate variables' in proofs. Example: 'Let c, c' be arbitrary but in the sequel fixed integers...' In this way one might prove
     $$c + c' = c' + c$$
     which has no free varaiables, indeed it has no variables at all. But when we are done we hopefully are able to infer
     $$\forall x \forall x' \; x + x' = x' + x$$
  3. *warning:* in the literature you read: if we have shown
     $$\exists x \, A(x)$$
     then we shoud be able to introduce a *new* constant symbol c as name for an element satisfying A
     $$A(c)$$
     this is an extension of what we treat here. It is not stright forward

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

2.1 Syntax

$$\Sigma_E = \{0,1,v,c,(,),+,\cdot,=,\wedge,\vee,\sim,\rightarrow,\exists,\forall\}$$

def: set of variables V

$$V = \{xw \,:\, w \in \mathbb{B}^*\}$$

def: set of constant symbols C

$$C = \{0,1\} \cup \{cw \,:\, w \in \mathbb{B}^*\}$$

def: set of terms T

1. $C \subset T$
2. $V \subset T$
3. if $a, b \in T$ then also
$$(a+b) \quad , \quad (a \cdot b)$$
4. these are all

example:

$$((v10+1) \cdot c11) \in T$$

def: set of arithmetic predicates P

1. for terms a, b:
$$(a = b) \in P$$
2. if $A, B \in P$ then also
$$(\sim A), (A \wedge B), (A \vee B), (A \rightarrow B)$$
3. if $y \in V$ and $A \in P$ the the following are in P
$$(\exists y\, A) \quad , \quad \forall y\, A)$$
4. these are all

example

$$(\forall v0(\forall v_1((v0+v1) = ((v1+v0)+v10))))$$

saving brackets: by priorities

$$\cdot \,,\, + \,,\, = \,,\, \forall \,,\, \exists \,,\, \sim \,,\, \wedge \,,\, \vee \,,\, \rightarrow$$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

2.1 Syntax

$$\Sigma_E = \{0,1,v,c,(,),+,\cdot,=,\wedge,\vee,\sim,\rightarrow,\exists,\forall\}$$

def: set of variables V

$$V = \{xw \;:\; w \in \mathbb{B}^*\}$$

def: set of constant symbols C

$$C = \{0,1\} \cup \{cw \;:\; w \in \mathbb{B}^*\}$$

def: set of terms T

1. $C \subset T$
2. $V \subset T$
3. if $a, b \in T$ then also

$$(a+b) \quad , \quad (a \cdot b)$$

4. these are all

example:

$$((v10+1) \cdot c11) \in T$$

def: set of arithmetic predicates P

1. for terms a, b:

$$(a = b) \in P$$

2. if $A, B \in P$ then also

$$(\sim A), (A \wedge B), (A \vee B), (A \rightarrow B)$$

3. if $y \in V$ and $A \in P$ the the following are in P

$$(\exists y\, A) \quad , \quad \forall y\, A)$$

4. these are all

example

$$(\forall v0(\forall v_1((v0+v1) = ((v1+v0)+v10))))$$

saving brackets: by priorities

$$\cdot\, , +\, , =\, , \forall\, , \exists\, , \sim\, , \wedge\, , \vee\, , \rightarrow$$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

def: set of arithmetic predicates P

1. for terms a, b:

$$(a = b) \in P$$

2. if $A, B \in P$ then also

$$(\sim A), (A \wedge B), (A \vee B), (A \rightarrow B)$$

3. if $y \in V$ and $A \in P$ the the following are in P

$$(\exists y\, A) \quad , \quad \forall y\, A)$$

4. these are all

example

$$(\forall v0(\forall v_1((v0 + v1) = ((v1 + v0) + v10))))$$

saving brackets: by priorities

$$\cdot\, ,\, + \,,\, = \,,\, \forall \,,\, \exists \,,\, \sim \,,\, \wedge \,,\, \vee \,,\, \rightarrow$$

def: free and bound occurrences of variables in predicates Let $y \in V$ be a variable

1. if $a, b \in T$ and y occurs in a or b, then y occurs *free* in $(a = b)$

2. if $A \in P$ and y occurs in A, then y occurs *bound* in $(\exists y\, A)$ and $(\forall y\, A)$

3. if $A, B \in P$ then i) the free and bound occurrences of y in $(\sim A)$ are the free and bound occurrences of y in A and ii) the free and bound occurrences of y in $(A \wedge B), (A \vee B), (A \rightarrow B)$ are the free and bound occurrences of y in A and the free and bound occurrences of y in B.

in above example: x_1 and x_0 occur bound, $x10$ occurs free.

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

def: set of arithmetic predicates P

1. for terms a, b:
$$(a = b) \in P$$
2. if $A, B \in P$ then also
$$(\sim A), (A \wedge B), (A \vee B), (A \to B)$$
3. if $y \in V$ and $A \in P$ the the following are in P
$$(\exists y\, A) \quad , \quad \forall y\, A)$$
4. these are all

example
$$(\forall v0(\forall v_1((v0 + v1) = ((v1 + v0) + v10))))$$

saving brackets: by priorities
$$\cdot \, , + , = , \forall , \exists , \sim , \wedge , \vee , \to$$

def: free and bound occurrences of variables in predicates Let $y \in V$ be a variable

1. if $a, b \in T$ and y occurs in a or b, then y occurs *free* in $(a = b)$
2. if $A \in P$ and y occurs in A, then y occurs *bound* in $(\exists y\, A)$ and $(\forall y\, A)$
3. if $A, B \in P$ then i) the free and bound occurrences of y in $(\sim A)$ are the free and bound occurrences of y in A and ii) the free and bound occurrences of y in $(A \wedge B), (A \vee B), (A \to B)$ are the free and bound occurrences of y in A and the free and bound occurrences of y in B.

in above example: x_1 and x_0 occur bound, $x10$ occurs free.

def: statement: a predicate without free variables.
For constant symbols c, c' and variables x, x'

- predicate: $c + c' = c' + c$
- not a predicate: $x + x'$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

def: set of arithmetic predicates P

1. for terms a, b:
$$(a = b) \in P$$
2. if $A, B \in P$ then also
$$(\sim A), (A \wedge B), (A \vee B), (A \to B)$$
3. if $y \in V$ and $A \in P$ the the following are in P
$$(\exists y\, A) \quad , \quad \forall y\, A)$$
4. these are all

example
$$(\forall v0 (\forall v_1 ((v0 + v1) = ((v1 + v0) + v10))))$$

saving brackets: by priorities
$$\cdot\, , + , = , \forall , \exists , \sim , \wedge , \vee , \to$$

def: free and bound occurrences of variables in predicates Let $y \in V$ be a variable

1. if $a, b \in T$ and y occurs in a or b, then y occurs *free* in $(a = b)$
2. if $A \in P$ and y occurs in A, then y occurs *bound* in $(\exists y\, A)$ and $(\forall y\, A)$
3. if $A, B \in P$ then i) the free and bound occurrences of y in $(\sim A)$ are the free and bound occurrences of y in A and ii) the free and bound occurrences of y in $(A \wedge B), (A \vee B), (A \to B)$ are the free and bound occurrences of y in A and the free and bound occurrences of y in B.

in above example: x_1 and x_0 occur bound, $x10$ occurs free.

def: statement: a predicate without free variables.
For constant symbols c, c' and variables x, x'

- predicate: $c + c' = c' + c$
- not a predicate: $x + x'$

def: language L_E of Z_E
$$L_E = \{p \in P \ : \ p \quad \text{is statement}\}$$

$Z_E = (\Sigma_E, L_E, A_E, R_E)$

2 The classical proof system Z_E: elementary number theory

def: set of arithmetic predicates P

1. for terms a, b:
$$(a = b) \in P$$
2. if $A, B \in P$ then also
$$(\sim A), (A \wedge B), (A \vee B), (A \rightarrow B)$$
3. if $y \in V$ and $A \in P$ the the following are in P
$$(\exists y\, A) \quad , \quad \forall y\, A)$$
4. these are all

example
$$(\forall v0(\forall v_1((v0 + v1) = ((v1 + v0) + v10))))$$

saving brackets: by priorities
$$\cdot\,,\, +\,,\, =\,,\, \forall\,,\, \exists\,,\, \sim\,,\, \wedge\,,\, \vee\,,\, \rightarrow$$

def: free and bound occurrences of variables in predicates Let $y \in V$ be a variable

1. if $a, b \in T$ and y occurs in a or b, then y occurs *free* in $(a = b)$
2. if $A \in P$ and y occurs in A, then y occurs *bound* in $(\exists y\, A)$ and $(\forall y\, A)$
3. if $A, B \in P$ then i) the free and bound occurrences of y in $(\sim A)$ are the free and bound occurrences of y in A and ii) the free and bound occurrences of y in $(A \wedge B), (A \vee B), (A \rightarrow B)$ are the free and bound occurrences of y in A and the free and bound occurrences of y in B.

in above example: x_1 and x_0 occur bound, $x10$ occurs free.

def: statement: a predicate without free variables.
For constant symbols c, c' and variables x, x'

- predicate: $c + c' = c' + c$
- not a predicate: $x + x'$

def: language L_E of Z_E
$$L_E = \{p \in P \;:\; p \quad \text{is statement}\}$$

Lemma 2. *L_E is decidable*

Proof. just a syntax check.

2.2 Truth (in a model)

model basis for defining the meaning of statements. Specifies

- the base set U, from which elements are drawn. Here usually $U = \mathbb{N}_0$
- an interpretation of the function symbols (here $+, *$) used.

2.2 Truth (in a model)

model basis for defining the meaning of statements. Specifies

- the base set U, from which elements are drawn. Here usually $U = \mathbb{N}_0$
- an interpretation of the function symbols (here $+, *$) used.

The standard model of Z_E:

- $U = \mathbb{N}_0$
- $+, *$: the usual addition and multiplication of natural numbers

there are others
promise: you will see
a nonstandard number

2.2 Truth (in a model)

model basis for defining the meaning of statements. Specifies

- the base set U, from which elements are drawn. Here usually $U = \mathbb{N}_0$
- an interpretation of the function symbols (here $+, *$) used.

The standard model of Z_E:

- $U = \mathbb{N}_0$
- $+, *$: the usual addition and multiplication of natural numbers

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0, 1\} :\rightarrow \mathbb{N}_0$$

2.2 Truth (in a model)

model basis for defining the meaning of statements. Specifies

- the base set U, from which elements are drawn. Here usually $U = \mathbb{N}_0$
- an interpretation of the function symbols (here $+, *$) used.

The standard model of Z_E:

- $U = \mathbb{N}_0$
- $+, *$: the usual addition and multiplication of natural numbers

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0,1\} :\to \mathbb{N}_0$$

extending φ to expressions

$$\begin{aligned}
\varphi(0) &= 0 \\
\varphi(1) &= 1 \\
\varphi(a+b) &= \varphi(a) + \varphi(b) \\
\varphi(a \cdot b) &= \varphi(a) + \varphi(b)
\end{aligned}$$

2.2 Truth (in a model)

model basis for defining the meaning of statements. Specifies

- the base set U, from which elements are drawn. Here usually $U = \mathbb{N}_0$
- an interpretation of the function symbols (here $+, *$) used.

The standard model of Z_E:

- $U = \mathbb{N}_0$
- $+, *$: the usual addition and multiplication of natural numbers

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0,1\} :\rightarrow \mathbb{N}_0$$

extending φ to expressions

$$\begin{aligned} \varphi(0) &= 0 \\ \varphi(1) &= 1 \\ \varphi(a+b) &= \varphi(a) + \varphi(b) \\ \varphi(a \cdot b) &= \varphi(a) + \varphi(b) \end{aligned}$$

attention:

- overloaded notation. $+$ and $\cdot$ on right side are interpretation of function symbols from model. The stuff we learned at elementary school (or I2CA).
- speaking of all valuations φ means: ranging over all possible values for variables *and* constant symbols

2.2 Truth (in a model)

model basis for defining the meaning of statements. Specifies

- the base set U, from which elements are drawn. Here usually $U = \mathbb{N}_0$
- an interpretation of the function symbols (here $+, *$) used.

The standard model of Z_E:

- $U = \mathbb{N}_0$
- $+, *$: the usual addition and multiplication of natural numbers

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0,1\} :\to \mathbb{N}_0$$

extending φ to expressions

$$\begin{aligned}
\varphi(0) &= 0 \\
\varphi(1) &= 1 \\
\varphi(a+b) &= \varphi(a) + \varphi(b) \\
\varphi(a \cdot b) &= \varphi(a) + \varphi(b)
\end{aligned}$$

attention:

- overloaded notation. $+$ and $\cdot$ on right side are interpretation of function symbols from model. The stuff we learned at elementary school (or I2CA).
- speaking of all valuations φ means: ranging over all possible values for variables *and* constant symbols

some notation

- For $n \in \mathbb{N}_0$ we define term $\overline{n} \in T$ by: $\overline{0} = 0$ and for $n > 0$:

$$\overline{n} = ((1+1)+\ldots+1) \quad n \text{ times}$$

- For predicates $A \in P$ with free variable x and terms $t \in T$ we denote by

$$A|_{x:=t}$$

the predicate obtained by substituting all free occurrences of x in A by t.

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0,1\} :\rightarrow \mathbb{N}_0$$

extending φ to expressions

$$\begin{aligned}
\varphi(0) &= 0 \\
\varphi(1) &= 1 \\
\varphi(a+b) &= \varphi(a)+\varphi(b) \\
\varphi(a\cdot b) &= \varphi(a)+\varphi(b)
\end{aligned}$$

some notation

- For $n \in \mathbb{N}_0$ we define term $\overline{n} \in T$ by: $\overline{0} = 0$ and for $n > 0$:

  $$\overline{n} = ((1+1)+\ldots+1) \quad n \text{ times}$$

- For predicates $A \in P$ with free variable x and terms $t \in T$ we denote by

  $$A|_{x:=t}$$

  the predicate obtained by substituting all free occurrences of x in A by t.

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0,1\} :\rightarrow \mathbb{N}_0$$

extending φ to expressions

$$\begin{aligned}
\varphi(0) &= 0 \\
\varphi(1) &= 1 \\
\varphi(a+b) &= \varphi(a) + \varphi(b) \\
\varphi(a \cdot b) &= \varphi(a) + \varphi(b)
\end{aligned}$$

some notation

- For $n \in \mathbb{N}_0$ we define term $\overline{n} \in T$ by: $\overline{0} = 0$ and for $n > 0$:

  $$\overline{n} = ((1+1)+\ldots+1) \quad n \text{ times}$$

- For predicates $A \in P$ with free variable x and terms $t \in T$ we denote by

  $$A|_{x:=t}$$

  the predicate obtained by substituting all free occurrences of x in A by t.

def: truth of statements For $a, b \in T$ and $A, B \in P$ and $x \in V$

$a = b$	true	$\Leftrightarrow$	$\varphi(a) = \varphi(b)$ for all valuations φ
$\sim A$	true	$\Leftrightarrow$	A not true
$A \wedge B$	true	$\Leftrightarrow$	A true and B true
$A \vee B$	true	$\Leftrightarrow$	A true or B true
$A \rightarrow B$	true	$\Leftrightarrow$	$(\sim A) \vee B$ true
$\exists x\, A$	true	$\Leftrightarrow$	there is $n \in \mathbb{N}_0$ such that $A_{x:=\overline{n}}$ true
$\forall x\, A$	true	$\Leftrightarrow$	for all $n \in \mathbb{N}_0$ such that $A_{x:=\overline{n}}$ true

def: valuation: plugging in constants for variables *and* constant symbols $\neq 0, 1$

$$\varphi : V \cup C \setminus \{0,1\} :\rightarrow \mathbb{N}_0$$

extending φ to expressions

$$\begin{aligned}
\varphi(0) &= 0 \\
\varphi(1) &= 1 \\
\varphi(a+b) &= \varphi(a)+\varphi(b) \\
\varphi(a \cdot b) &= \varphi(a)+\varphi(b)
\end{aligned}$$

some notation

- For $n \in \mathbb{N}_0$ we define term $\overline{n} \in T$ by: $\overline{0} = 0$ and for $n > 0$:

  $$\overline{n} = ((1+1)+\ldots+1) \quad n \text{ times}$$

- For predicates $A \in P$ with free variable x and terms $t \in T$ we denote by

  $$A|_{x:=t}$$

  the predicate obtained by substituting all free occurrences of x in A by t.

def: truth of statements For $a, b \in T$ and $A, B \in P$ and $x \in V$

$$\begin{array}{rlcl}
a = b & \text{true} & \Leftrightarrow & \varphi(a) = \varphi(b) \text{ for all valuations } \varphi \\
\sim A & \text{true} & \Leftrightarrow & A \text{ not true} \\
A \wedge B & \text{true} & \Leftrightarrow & A \text{ true and } B \text{ true} \\
A \vee B & \text{true} & \Leftrightarrow & A \text{ true or } B \text{ true} \\
A \rightarrow B & \text{true} & \Leftrightarrow & (\sim A) \vee B \text{ true} \\
\exists x\, A & \text{true} & \Leftrightarrow & \text{there is } n \in \mathbb{N}_0 \text{ such that } A_{x:=\overline{n}} \quad \text{true} \\
\forall x\, A & \text{true} & \Leftrightarrow & \text{for all } n \in \mathbb{N}_0 \text{ such that } A_{x:=\overline{n}} \quad \text{true}
\end{array}$$

attention:

- first line gives a kind of implicit universal quantification over constant symbols.
- right hand sides formulated in (hopefully) precise subset of natural language

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\ldots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\ldots,x_n\} \rightarrow \mathbb{B}$$

relax for a change:

this is just to show you a
'sufficiently powerful' proof system

we will not construct
formal proofs (here)

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1, \ldots, x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1, \ldots, x_n\} \to \mathbb{B}$$

relax for a change:

this is just to show you a
'sufficiently powerful' proof system

we will not construct
formal proofs (here)

for Goedel's second
incompleteness theorem

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\dots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\dots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\dots,x_n)$ tautology, $P_1,\dots,P_N$ statements. Obtain $A(P_1,\dots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\dots,P_n) \quad \text{axiom}$$

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\dots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\dots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\dots,x_n)$ tautology, $P_1,\dots,P_N$ statements. Obtain $A(P_1,\dots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\dots,P_n) \quad \text{axiom}$$

modus ponens:

- A, B statements

$$\frac{A \ , \ A \to B}{B} \quad \text{proof rule}$$

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\dots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\dots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\dots,x_n)$ tautology, $P_1,\dots,P_N$ statements. Obtain $A(P_1,\dots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\dots,P_n) \quad \text{axiom}$$

modus ponens:

- A,B statements

$$\frac{A\ ,\ A \to B}{B} \quad \text{proof rule}$$

change of variables

- obtain A' from A by exchanging all occurrences of variable x by variable x'. Then

$$A \leftrightarrow A' \quad \text{axiom}$$

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\dots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\dots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\dots,x_n)$ tautology, $P_1,\dots,P_N$ statements. Obtain $A(P_1,\dots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\dots,P_n) \quad \text{axiom}$$

modus ponens:

- A, B statements

$$\frac{A\ ,\ A \to B}{B} \quad \text{proof rule}$$

change of variables

- obtain A' from A by exchanging all occurrences of variable x by variable x'. Then

$$A \leftrightarrow A' \quad \text{axiom}$$

quantors $A(x)$ predicate with free variable x, C predicate with no free occurrence of x, c constant symbol.

- axiom

$$\forall x\, A(x) \ \to\ A(c)$$

- axioms

$$\begin{aligned} \sim \forall x\, A(x) &\ \leftrightarrow\ \exists x \sim A(x) \\ C \vee \forall x\, A(x) &\ \leftrightarrow\ \forall x\, (C \vee A(x) \\ C \wedge \forall x\, A(x) &\ \leftrightarrow\ \forall x\, (C \wedge A(x) \end{aligned}$$

- proof rule

$$\frac{A(c) \to C}{\exists x\, A(x) \to C} \qquad ??$$

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\ldots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\ldots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\ldots,x_n)$ tautology, $P_1,\ldots,P_N$ statements. Obtain $A(P_1,\ldots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\ldots,P_n) \quad \text{axiom}$$

modus ponens:

- A, B statements

$$\frac{A\ ,\ A \to B}{B} \quad \text{proof rule}$$

change of variables

- obtain A' from A by exchanging all occurrences of variable x by variable x'. Then

$$A \leftrightarrow A' \quad \text{axiom}$$

quantors $A(x)$ predicate with free variable x, C predicate with no free occurrence of x, c constant symbol.

- axiom

$$\forall x\, A(x) \ \to\ A(c)$$

- axioms

$$\begin{aligned} \sim \forall x\, A(x) &\ \leftrightarrow\ \exists x \sim A(x) \\ C \vee \forall x\, A(x) &\ \leftrightarrow\ \forall x\, (C \vee A(x) \\ C \wedge \forall x\, A(x) &\ \leftrightarrow\ \forall x\, (C \wedge A(x) \end{aligned}$$

- proof rule

$$\frac{A(c) \to C}{\exists x\, A(x) \to C} \qquad ??$$

Lemma 3. *From a proof of $A(c)$ you can conclude $\forall x\, A(x)$.*

$$\frac{A(c)}{\forall x\, A(x)}$$

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\ldots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\ldots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\ldots,x_n)$ tautology, $P_1,\ldots,P_N$ statements. Obtain $A(P_1,\ldots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\ldots,P_n) \quad \text{axiom}$$

modus ponens:

- A, B statements

$$\frac{A \;,\; A \to B}{B} \quad \text{proof rule}$$

change of variables

- obtain A' from A by exchanging all occurrences of variable x by variable x'. Then

$$A \leftrightarrow A' \quad \text{axiom}$$

quantors $A(x)$ predicate with free variable x, C predicate with no free occurrence of x, c constant symbol.

- axiom

$$\forall x\, A(x) \;\to\; A(c)$$

- axioms

$$\begin{aligned}
\sim \forall x\, A(x) &\;\leftrightarrow\; \exists x \sim A(x) \\
C \vee \forall x\, A(x) &\;\leftrightarrow\; \forall x\, (C \vee A(x) \\
C \wedge \forall x\, A(x) &\;\leftrightarrow\; \forall x\, (C \wedge A(x)
\end{aligned}$$

- proof rule

$$\frac{A(c) \to C}{\exists x\, A(x) \to C}$$

??

Lemma 3. *From a proof of $A(c)$ you can conclude $\forall x\, A(x)$.*

$$\frac{A(c)}{\forall x\, A(x)}$$

- set

$$B = \sim A$$

then

$$(\sim B(c) \to 0) \leftrightarrow (0 \vee \sim B) \leftrightarrow A(c)$$

2.3 Axioms and proof rules of Z_E

def:tautology A boolean expression $A(x_1,\dots,x_n)$ is a *tautology* if

$$\varphi(A) = 1 \quad \text{for all valuations } \varphi : \{x_1,\dots,x_n\} \to \mathbb{B}$$

predicate calculus

- $A(x_1,\dots,x_n)$ tautology, $P_1,\dots,P_N$ statements. Obtain $A(P_1,\dots,P_n)$ by substituting x_i by P_i for all i.

$$A(P_1,\dots,P_n) \quad \text{axiom}$$

modus ponens:

- A, B statements

$$\frac{A \;,\; A \to B}{B} \quad \text{proof rule}$$

change of variables

- obtain A' from A by exchanging all occurrences of variable x by variable x'. Then

$$A \leftrightarrow A' \quad \text{axiom}$$

quantors $A(x)$ predicate with free variable x, C predicate with no free occurrence of x, c constant symbol.

- axiom

$$\forall x\, A(x) \;\to\; A(c)$$

- axioms

$$\begin{aligned} \sim \forall x\, A(x) \;&\leftrightarrow\; \exists x \sim A(x) \\ C \vee \forall x\, A(x) \;&\leftrightarrow\; \forall x\, (C \vee A(x) \\ C \wedge \forall x\, A(x) \;&\leftrightarrow\; \forall x\, (C \wedge A(x) \end{aligned}$$

- proof rule

$$\frac{A(c) \to C}{\exists x\, A(x) \to C}$$

??

Lemma 3. *From a proof of $A(c)$ you can conclude $\forall x\, A(x)$.*

$$\frac{A(c)}{\forall x\, A(x)}$$

- set

$$B = \sim A$$

then

$$(\sim B(c) \to 0) \leftrightarrow (0 \vee \sim B) \leftrightarrow A(c)$$

- conclude

$$\begin{aligned} \exists x B(x) \to 0 \;&\leftrightarrow\; 0 \vee \sim \exists x\, B(x) \\ &\leftrightarrow\; \forall x \sim B(x) \\ &\leftrightarrow\; \forall x\, A(x) \end{aligned}$$

equality: axioms a, b, c terms

- $$a = a$$

- $$a = b \rightarrow b = a$$

- $$a = b \land b = c \rightarrow b = c$$

equivalence relation

equality: axioms a, b, c terms

- $$a = a$$
- $$a = b \rightarrow b = a$$
- $$a = b \wedge b = c \rightarrow b = c$$

equivalence relation

natural numbers: axioms for all predicates $A(x)$, all terms a, b

- induction
$$A(0) \wedge \forall x \ (A(x) \rightarrow A(x+1)) \rightarrow A(y)$$
- $$a+1 = b+1 \ \rightarrow a = b$$
- $$\sim a+1 = 0$$
- $$a = b \ \rightarrow a+1 = b+1$$

Peano axioms

equality: axioms a, b, c terms

- $$a = a$$
- $$a = b \to b = a$$
- $$a = b \wedge b = c \to b = c$$

equivalence relation

natural numbers: axioms for all predicates $A(x)$, all terms a, b

- induction
$$A(0) \wedge \forall x\ (A(x) \to A(x+1)) \to A(y)$$
- $$a + 1 = b + 1 \to a = b$$
- $$\sim a + 1 = 0$$
- $$a = b \to a + 1 = b + 1$$

Peano axioms

arithmetic operations: axioms a, b terms

- $$a + 0 = a$$
- $$a + (b + 1) = (a + b) + 1$$
- $$a \cdot 0 = 0$$
- $$a \cdot (b + 1) = a \cdot b + a$$

inductive definitions of + and $\bullet$

3 A glimpse at model theory

def: consistent set of statements: A set S of statements is *consistent* if no statement of the form $A \wedge \sim A$ can be derived from S.

By tedious bookkeeping one can show

Lemma 4. *If A is provable, then it is true in every model.*

Lemma 5. *If a set S of statements has a model, then it is consistent.*

4 A bit of history

- Russel and Whitehead 1910-1913: Tried to develop mathematics very much in the style of the above historical proof system. Hard to read.

4 A bit of history

- Russel and Whitehead 1910-1913: Tried to develop mathematics very much in the style of the above historical proof system. Hard to read.

page 200 of 'Principia Mathematica'

***32·241.** $\vdash . \text{gs}\text{‘}\breve{R} = \text{sg}\text{‘}R$ [Similar proof]

***32·25.** $\vdash : A \,\text{sg}\, R . \equiv . A = \text{sg}\text{‘}R$ [*30·4 . *32·22]

***32·251.** $\vdash : A \,\text{gs}\, R . \equiv . A = \text{gs}\text{‘}R$ [*30·4 . *32·221]

***32·3.** $\vdash . \{\text{sg}\text{‘}(R \dot{\cap} S)\}\text{‘}y = \overrightarrow{R}\text{‘}y \cap \overrightarrow{S}\text{‘}y$

Note that we do *not* have

$$\text{sg}\text{‘}(R \dot{\cap} S) = \text{sg}\text{‘}R \dot{\cap} \text{sg}\text{‘}S.$$

Dem.

$\vdash . *32{\cdot}23{\cdot}13 . \supset \vdash . \{\text{sg}\text{‘}(R \dot{\cap} S)\}\text{‘}y = \hat{x}\{x(R \dot{\cap} S)y\}$

[*23·33] $= \hat{x}(xRy . xSy)$

[*22·39] $= \hat{x}(xRy) \cap \hat{x}(xSy)$

[*32·13] $= \overrightarrow{R}\text{‘}y \cap \overrightarrow{S}\text{‘}y . \supset \vdash . \text{Prop}$

***32·31.** $\vdash . \{\text{gs}\text{‘}(R \dot{\cap} S)\}\text{‘}x = \overleftarrow{R}\text{‘}x \cap \overleftarrow{S}\text{‘}x$

***32·32.** $\vdash . \{\text{sg}\text{‘}(R \dot{\cup} S)\}\text{‘}y = \overrightarrow{R}\text{‘}y \cup \overrightarrow{S}\text{‘}y$

***32·33.** $\vdash . \{\text{gs}\text{‘}(R \dot{\cup} S)\}\text{‘}x = \overleftarrow{R}\text{‘}x \cup \overleftarrow{S}\text{‘}x$

***32·34.** $\vdash . \{\text{sg}\text{‘}(\dot{-} R)\}\text{‘}y = - \overrightarrow{R}\text{‘}y$

***32·35.** $\vdash . \{\text{gs}\text{‘}(\dot{-} R)\}\text{‘}x = - \overleftarrow{R}\text{‘}x$

The proofs of the above propositions are similar to that of *32·2.

***32·4.** $\vdash :. \text{E}\,!\, \overrightarrow{R}\text{‘}z . \equiv : \exists\,!\, \overrightarrow{R}\text{‘}z : x, y \in \overrightarrow{R}\text{‘}z . \supset_{x,y} . x = y$ [*30·21 . *32·18]

***32·41.** $\vdash :. \text{E}\,!\, S\text{‘}y . \supset : \overrightarrow{R}\text{‘}y = \overrightarrow{S}\text{‘}y . \equiv . R\text{‘}y = S\text{‘}y$

Dem.

$\vdash . *4{\cdot}86 . \quad \supset \vdash :: xSy . \equiv_x . x = b : \supset :.$

$xRy . \equiv_x . xSy : \equiv : xRy . \equiv_x . x = b$ (1)

$\vdash . (1) . *5{\cdot}32 . \supset \vdash :. xSy . \equiv_x . x = b : xRy . \equiv_x . xSy : \equiv :$

$xSy . \equiv_x . x = b : xRy . \equiv_x . x = b$ (2)

$\vdash . (2) . *10{\cdot}11{\cdot}281 . *32{\cdot}18{\cdot}181 . \supset$

$\vdash :. (\exists b) : xSy . \equiv_x . x = b : \overrightarrow{R}\text{‘}y = \overrightarrow{S}\text{‘}y : \equiv : (\exists b) : xSy . \equiv_x . x = b : xRy . \equiv_x . x = b :$

[*30·3.*14·13] $\equiv : (\exists b) : xSy . \equiv_x . x = b : R\text{‘}y = b :$

[*14·101] $\equiv : R\text{‘}y = S\text{‘}y$ (3)

$\vdash . (3) . *30{\cdot}2 . \supset \vdash :. \text{E}\,!\, S\text{‘}y . \overrightarrow{R}\text{‘}y = \overrightarrow{S}\text{‘}y . \equiv . R\text{‘}y = S\text{‘}y :. \supset \vdash . \text{Prop}$

4 A bit of history

- Russel and Whitehead 1910-1913: Tried to develop mathematics very much in the style of the above historical proof system. Hard to read.

page 200 of 'Principia Mathematica' →

$*32{\cdot}241.\ \vdash . \mathrm{gs}‘\breve{R} = \mathrm{sg}‘R$ [Similar proof]

$*32{\cdot}25.\ \vdash : A \,\mathrm{sg}\, R . \equiv . A = \mathrm{sg}‘R$ $[*30{\cdot}4 . *32{\cdot}22]$

$*32{\cdot}251.\ \vdash : A \,\mathrm{gs}\, R . \equiv . A = \mathrm{gs}‘R$ $[*30{\cdot}4 . *32{\cdot}221]$

$*32{\cdot}3.\ \vdash . \{\mathrm{sg}‘(R \dot\cap S)\}‘y = \overrightarrow{R}‘y \cap \overrightarrow{S}‘y$

Note that we do *not* have

$$\mathrm{sg}‘(R \dot\cap S) = \mathrm{sg}‘R \dot\cap \mathrm{sg}‘S.$$

Dem.

$\vdash . *32{\cdot}23{\cdot}13 . \supset \vdash . \{\mathrm{sg}‘(R \dot\cap S)\}‘y = \hat{x}\{x(R \dot\cap S)y\}$

$[*23{\cdot}33]\quad = \hat{x}(xRy . xSy)$

$[*22{\cdot}39]\quad = \hat{x}(xRy) \cap \hat{x}(xSy)$

$[*32{\cdot}13]\quad = \overrightarrow{R}‘y \cap \overrightarrow{S}‘y . \supset \vdash . \mathrm{Prop}$

$*32{\cdot}31.\ \vdash . \{\mathrm{gs}‘(R \dot\cap S)\}‘x = \overleftarrow{R}‘x \cap \overleftarrow{S}‘x$

$*32{\cdot}32.\ \vdash . \{\mathrm{sg}‘(R \dot\cup S)\}‘y = \overrightarrow{R}‘y \cup \overrightarrow{S}‘y$

$*32{\cdot}33.\ \vdash . \{\mathrm{gs}‘(R \dot\cup S)\}‘x = \overleftarrow{R}‘x \cup \overleftarrow{S}‘x$

$*32{\cdot}34.\ \vdash . \{\mathrm{sg}‘(\dot- R)\}‘y = - \overrightarrow{R}‘y$

$*32{\cdot}35.\ \vdash . \{\mathrm{gs}‘(\dot- R)\}‘x = - \overleftarrow{R}‘x$

The proofs of the above propositions are similar to that of $*32{\cdot}2$.

$*32{\cdot}4.\ \vdash :. E ! R‘z . \equiv : \exists ! \overrightarrow{R}‘z : x, y \,\epsilon\, \overrightarrow{R}‘z . \supset_{x,y} . x = y$ $[*30{\cdot}21 . *32{\cdot}18]$

$*32{\cdot}41.\ \vdash :. E ! S‘y . \supset : \overrightarrow{R}‘y = \overrightarrow{S}‘y . \equiv . R‘y = S‘y$

Dem.

$\vdash . *4{\cdot}86 . \quad \supset \vdash :: xSy . \equiv_x . x = b : \supset :.$

$xRy . \equiv_x . xSy : \equiv : xRy . \equiv_x . x = b \quad (1)$

$\vdash . (1) . *5{\cdot}32 . \supset \vdash :. xSy . \equiv_x . x = b : xRy . \equiv_x . xSy : \equiv :$

$xSy . \equiv_x . x = b : xRy . \equiv_x . x = b \quad (2)$

$\vdash . (2) . *10{\cdot}11{\cdot}281 . *32{\cdot}18{\cdot}181 . \supset$

$\vdash :. (\exists b) : xSy . \equiv_x . x = b : \overrightarrow{R}‘y = \overrightarrow{S}‘y : \equiv : (\exists b) : xSy . \equiv_x . x = b : xRy . \equiv_x . x = b :$

$[*30{\cdot}3 . *14{\cdot}13]\quad \equiv : (\exists b) : xSy . \equiv_x . x = b : R‘y = b :$

$[*14{\cdot}101]\quad \equiv : R‘y = S‘y \quad (3)$

$\vdash . (3) . *30{\cdot}2 . \supset \vdash :. E ! S‘y . \overrightarrow{R}‘y = \overrightarrow{S}‘y . \equiv . R‘y = S‘y :. \supset \vdash . \mathrm{Prop}$

- around 2000: reasonably comfortable computer aided verification (CAV) sysytems: PVS, Isabelle/Hol, Coq.
- interest to guarantee correctness of (portions of) computer system; much increased by 'Pentium Bug'.
- 2003-2007: proofs of our textbook 'system architecture' (and many others) formally verified in Isabelle/Hol

4 A bit of history

- Russel and Whitehead 1910-1913: Tried to develop mathematics very much in the style of the above historical proof system. Hard to read.

page 200 of 'Principia Mathematica'

$*32{\cdot}241.\ \vdash .\, \mathrm{gs}\text{‘}\breve{R} = \mathrm{sg}\text{‘}R$ [Similar proof]

$*32{\cdot}25.\ \vdash : A\ \mathrm{sg}\ R \,.\!\equiv\!.\, A = \mathrm{sg}\text{‘}R$ [$*30{\cdot}4 . *32{\cdot}22$]

$*32{\cdot}251.\ \vdash : A\ \mathrm{gs}\ R \,.\!\equiv\!.\, A = \mathrm{gs}\text{‘}R$ [$*30{\cdot}4 . *32{\cdot}221$]

$*32{\cdot}3.\ \vdash .\, \{\mathrm{sg}\text{‘}(R \dot{\cap} S)\}\text{‘}y = \overrightarrow{R}\text{‘}y \cap \overrightarrow{S}\text{‘}y$

Note that we do *not* have

$$\mathrm{sg}\text{‘}(R \dot{\cap} S) = \mathrm{sg}\text{‘}R \dot{\cap} \mathrm{sg}\text{‘}S.$$

Dem.

$\vdash . *32{\cdot}23{\cdot}13 . \supset \vdash .\, \{\mathrm{sg}\text{‘}(R \dot{\cap} S)\}\text{‘}y = \hat{x}\{x (R \dot{\cap} S) y\}$

[$*23{\cdot}33$] $= \hat{x}(xRy \,.\, xSy)$

[$*22{\cdot}39$] $= \hat{x}(xRy) \cap \hat{x}(xSy)$

[$*32{\cdot}13$] $= \overrightarrow{R}\text{‘}y \cap \overrightarrow{S}\text{‘}y \,.\supset \vdash .$ Prop

$*32{\cdot}31.\ \vdash .\, \{\mathrm{gs}\text{‘}(R \dot{\cap} S)\}\text{‘}x = \overleftarrow{R}\text{‘}x \cap \overleftarrow{S}\text{‘}x$

$*32{\cdot}32.\ \vdash .\, \{\mathrm{sg}\text{‘}(R \dot{\cup} S)\}\text{‘}y = \overrightarrow{R}\text{‘}y \cup \overrightarrow{S}\text{‘}y$

$*32{\cdot}33.\ \vdash .\, \{\mathrm{gs}\text{‘}(R \dot{\cup} S)\}\text{‘}x = \overleftarrow{R}\text{‘}x \cup \overleftarrow{S}\text{‘}x$

$*32{\cdot}34.\ \vdash .\, \{\mathrm{sg}\text{‘}(\dot{-} R)\}\text{‘}y = -\overrightarrow{R}\text{‘}y$

$*32{\cdot}35.\ \vdash .\, \{\mathrm{gs}\text{‘}(\dot{-} R)\}\text{‘}x = -\overleftarrow{R}\text{‘}x$

The proofs of the above propositions are similar to that of $*32{\cdot}2$.

$*32{\cdot}4.\ \vdash :.\, \mathrm{E!}\, R\text{‘}z \,.\!\equiv :\, \exists !\, \overrightarrow{R}\text{‘}z : x, y \in \overrightarrow{R}\text{‘}z \,.\supset_{x,y} .\, x = y$ [$*30{\cdot}21 . *32{\cdot}18$]

$*32{\cdot}41.\ \vdash :.\, \mathrm{E!}\, S\text{‘}y \,.\supset :\, \overrightarrow{R}\text{‘}y = \overrightarrow{S}\text{‘}y \,.\!\equiv\!.\, R\text{‘}y = S\text{‘}y$

Dem.

$\vdash . *4{\cdot}86 . \quad \supset \vdash :: xSy \,.\!\equiv_x\!.\, x = b :\supset :.$

$xRy \,.\!\equiv_x\!.\, xSy : \equiv : xRy \,.\!\equiv_x\!.\, x = b$ (1)

$\vdash . (1) . *5{\cdot}32 . \supset \vdash :.\, xSy \,.\!\equiv_x\!.\, x = b : xRy \,.\!\equiv_x\!.\, xSy : \equiv :$

$xSy \,.\!\equiv_x\!.\, x = b : xRy \,.\!\equiv_x\!.\, x = b$ (2)

$\vdash . (2) . *10{\cdot}11{\cdot}281 . *32{\cdot}18{\cdot}181 . \supset$

$\vdash :.\, (\exists b) : xSy \,.\!\equiv_x\!.\, x = b : \overrightarrow{R}\text{‘}y = \overrightarrow{S}\text{‘}y : \equiv : (\exists b) : xSy \,.\!\equiv_x\!.\, x = b : xRy \,.\!\equiv_x\!.\, x = b :$

[$*30{\cdot}3 . *14{\cdot}13$] $\equiv : (\exists b) : xSy \,.\!\equiv_x\!.\, x = b : R\text{‘}y = b :$

[$*14{\cdot}101$] $\equiv : R\text{‘}y = S\text{‘}y$ (3)

$\vdash . (3) . *30{\cdot}2 . \supset \vdash :.\, \mathrm{E!}\, S\text{‘}y \,.\, \overrightarrow{R}\text{‘}y = \overrightarrow{S}\text{‘}y \,.\!\equiv\!.\, R\text{‘}y = S\text{‘}y :. \supset \vdash .$ Prop

- around 2000: reasonably comfortable computer aided verification (CAV) sysytems: PVS, Isabelle/Hol, Coq.
- interest to guarantee correctness of (portions of) computer system; much increased by 'Pentium Bug'.
- 2003-2007: proofs of our textbook 'system architecture' (and many others) formally verified in Isabelle/Hol

precursors of proofs in 2016 textbook

results in book are 'better' but
errors have been reintroduced