Ein Beispiel (Forts.)

Beweis zu den Lemmata

Beweis zu $L1$: Induktion über $\sigma(y)$.

$\sigma(y) \equiv 0 : \quad 0 + 0 =_R 0$

$\sigma(y) \equiv s(t) : \quad 0 + s(t) =_R s(0 + t) =_{IV} s(t)$

Beweis zu $L2$: Induktion über $\sigma(y)$, sei $\sigma(x) = \bar{x}$.

$\sigma(y) \equiv 0 : \quad s(\bar{x}) + 0 =_R s(\bar{x}) =_R s(\bar{x} + 0)$

$$\begin{aligned} \sigma(y) \equiv s(t) : \quad s(\bar{x}) + s(t) &=_R s(s(\bar{x}) + t) =_{IV} s(s(\bar{x} + t)) \\ &=_R s(\bar{x} + s(t)) \end{aligned}$$

Beweis zur Beh.: Induktion über $\sigma(y)$, sei $\sigma(x) = \bar{x}$.

$\sigma(y) \equiv 0 : \quad \bar{x} + 0 =_R \bar{x} =_{L1} 0 + \bar{x}$

$$\begin{aligned} \sigma(y) \equiv s(t) : \quad \bar{x} + s(t) &=_R s(\bar{x} + t) =_{IV} s(t + \bar{x}) \\ &=_{L2} s(t) + \bar{x} \end{aligned}$$

3 Beweisverfahren für induktive Theoreme

Problemstellung

Gegeben: spec $= (\text{sig}, E)$, sig $= (S, F, \alpha)$
$C \subseteq F$ Konstruktormenge
$L \subseteq ITh(\text{spec})$ Lemmata
$s = t$

Frage: Gilt $s = t \in ITh(\text{spec})$?

Ansatz

Zeige für alle $\sigma \in C$-GSub$(s = t)$, dass $\sigma(s) =_{E \cup L} \sigma(t)$ gilt.

Beweis-Idee

Sei $\{x_1, \ldots, x_n\} \subseteq \text{Var}(s, t)$.

Definiere Ordnung $\succ_{ind}$ auf Term$(C)^n$ und damit die Induktionsordnung $\succ_i$ durch
$\sigma \succ_i \tau$ gdw $(\sigma(x_1), \ldots, \sigma(x_n)) \succ_{ind} (\tau(x_1), \ldots, \tau(x_n))$.
Zeige $\sigma(s) =_{E \cup L} \sigma(t)$ durch Noethersche Induktion nach $\succ_i$.

Voraussetzung hier $\sigma(y) \equiv \tau(y) \equiv y$ für alle $y \in \text{Var}(s, t) - \{x_1, \ldots, x_n\}$.
Die x_i heißen **Induktionsvariablen**.

Verallgemeinerung der Problemstellung

Beachte

E enthält bedingte Gleichungen der Form
$u_1 = v_1, \ldots, u_n = v_n \Rightarrow s = t$.

Schreibweise

1. $u = v \Rightarrow s = t$ oder $s = t \Leftarrow u = v$, falls
   $u = (u_1, \ldots, u_n)$, $v = (v_1, \ldots, v_n)$.
2. $\sigma(u) = \sigma(v)$ statt $\sigma(u_1) = \sigma(v_1), \ldots, \sigma(u_n) = \sigma(v_n)$.

Beachte

$u = v \Rightarrow s = t$ gilt in $\mathcal{I} = \mathcal{I}(\text{spec})$, falls für alle
$\sigma \in C\text{-GSub}(u = v \Rightarrow s = t)$ gilt:

Gilt $\sigma(u) = \sigma(v)$ in $\mathcal{I}$, so gilt auch $\sigma(s) = \sigma(t)$ in $\mathcal{I}$.

Warnung: Satz 2.4 gilt nur für unbedingte Gleichungen.

Ab jetzt

$ITh(\text{spec}) = \{u = v \Rightarrow s = t \mid u = v \Rightarrow s = t \text{ gilt in } \mathcal{I}(\text{spec})\}$

Die Lemmata haben die Form $u = v \Rightarrow s = t$.

Die Behauptung hat die Form $u = v \Rightarrow s = t$.

3.1 Explizite Induktion

Sei sig, spec, C, L wie immer, $\mathcal{I} = \mathcal{I}(\text{spec})$.

Ist $e \equiv u = v \Rightarrow s = t$ eine Gleichung so ist
$\sigma(e) \equiv \sigma(u) = \sigma(v) \Rightarrow \sigma(s) = \sigma(t)$.

Ansatz

Sei $e \equiv u = v \Rightarrow s = t$, zu zeigen ist $e \in ITh(\text{spec})$. Wähle Noethersche Partialordnung $\succ_i$ auf C-GSub(e).

Zeige für alle $\sigma \in C$-GSub(e): Gilt $\tau(e)$ in $\mathcal{I}$ für alle $\tau \in C$-GSub(e) mit $\sigma \succ_i \tau$, so gilt $\sigma(e)$ in $\mathcal{I}$.

Dann gilt $e \in ITh(\text{spec})$.

Fragen

a) Wie wählt man die Induktionsvariablen
   $x_1, \ldots, x_n \in \text{Var}(e)$?
b) Wie wählt man die Induktionsordnung $\succ_i$?

Antwort

Das Definitionsschema der $f \in F - C$ hilft hier.

Bemerkungen

1. Bemerkung

Sei $\sigma \succ_i \tau$ gdw $(\sigma(x), \sigma(y)) >_{TT}^{lex} (\tau(x), \tau(y))$.

Dies entspricht der geschachtelten Induktion nach $>_{TT}$:
erst nach x, bei gleichem x nach y.

2. Bemerkung

Wir schreiben x statt $\sigma(x)$. Dies formalisiert „Sei x beliebig, aber fest".

3. Bemerkung

Häufig wird $\succ_i$ mit syntaktischen Ordnungen definiert, z. B. mit $>_{TT}$ oder einer LPO. Man kann aber auch semantische Ordnungen wählen, sie müssen nur Noethersch sein (z. B. die Länge von Listen).

4. Bemerkung

Wir benutzen auch die Fallunterscheidung als Beweismittel.

Ein Beispiel (1)

Sei $F = \{\text{true}, \text{false}, 0, s, \leq\}$, $C = \{\text{true}, \text{false}, 0, s\}$.

E: $0 \leq x = \text{true}$
$s(x) \leq 0 = \text{false}$
$s(x) \leq s(y) = x \leq y$

Dann ist C eine Konstruktormenge.

1. Behauptung $x \leq x = \text{true}$

Beweis: $x \equiv 0$: Es gilt $0 \leq 0 =_E \text{true}$.

$x \rightsquigarrow s(x)$: zu zeigen $s(x) \leq s(x) = \text{true}$.

Es gilt $s(x) \leq s(x) =_E x \leq x =_{IV} \text{true}$.

Beachte: Hier ist $\sigma \succ_i \tau$ gdw $\sigma(x) \equiv s(x) >_{TT} \tau(x) \equiv x$.

2. Behauptung $e \equiv x \leq 0 = \text{true} \Rightarrow x = 0$

Beweis: $x \equiv 0$: Es gilt $0 \leq 0 = \text{true} \Rightarrow 0 = 0$.

$x \rightsquigarrow s(x)$: Es gilt $s(x) \leq 0 \neq_E \text{true}$, also gilt
$s(x) \leq 0 = \text{true} \Rightarrow s(x) = 0$ in $\mathcal{I}$.

Ein Beispiel (2)

Schreibweise: t statt $t =$ true, falls t von der Sorte BOOL ist.

3. Behauptung

$e \equiv x \leq y =$ true, $y \leq z =$ true $\Rightarrow x \leq z =$ true

Wähle die Induktionsordnung $\succ_i$ als
$\sigma \succ_i \tau \text{ gdw } (\sigma(x), \sigma(y), \sigma(z)) >_{TT}^{lex} (\tau(x), \tau(y), \tau(z))$.

1. $\sigma(x) \equiv 0$: Wegen $0 \leq z =$ true gilt $\sigma(e)$ in $\mathcal{I}$.

2. $\sigma(x) \equiv s(x)$: Zu zeigen $s(x) \leq y, y \leq z \Rightarrow s(x) \leq z$.

2.1 $\sigma(y) \equiv 0$: Wegen $s(x) \leq 0 \neq_E$ true gilt $\sigma(e)$ in $\mathcal{I}$.

2.2 $\sigma(y) \equiv s(y)$:
Zu zeigen $s(x) \leq s(y)$, $s(y) \leq z \Rightarrow s(x) \leq z$.

2.2.1 $\sigma(z) \equiv 0$: Wegen $s(y) \leq 0 \neq_E$ true gilt $\sigma(e)$ in $\mathcal{I}$.

2.2.2 $\sigma(z) \equiv s(z)$:
Zu zeigen $s(x) \leq s(y), s(y) \leq s(z) \Rightarrow s(x) \leq s(z)$.
Dies ist E-gleich zu $\tau(e) \equiv x \leq y, y \leq z \Rightarrow x \leq z$.
Dies gilt nach IV mit $\tau =$ id, also gilt $\sigma(e)$ in $\mathcal{I}$.

Ein Beispiel (3)

4. Behauptung $x \leq y =$ false $\Rightarrow y \leq x =$ true.

Beweis: Übung.

5. Behauptung $x \leq y =$ true $\vee\, x \leq y =$ false.

Beweis: Übung.

Beachte

a) $t_1 \neq_E t_2$ steht hier für $\mathcal{I} \models t_1 \neq t_2$ mit $\mathcal{I} = \mathcal{I}$(spec).

b) Es gilt $\mathcal{I} \models u = v \Rightarrow s = t$ genau dann, wenn für jede Grundsubstitution σ gilt: $\sigma(u) \neq_E \sigma(v)$ oder $\sigma(s) =_E \sigma(t)$.

c) Also: Aus $u \neq_E v$ folgt $\mathcal{I} \models u = v \Rightarrow s = t$.

Beachte zum Nachweis von $t_1 =_E t_2$ bzw. $t_1 \neq_E t_2$:

Richten von E ergebe konvergentes R.

Also gilt $t_1 =_E t_2 \text{ gdw } t_1 \downarrow_R t_2$.
$t_1 \neq_E t_2$ falls $t_i \xrightarrow{*}_R c_i$, $c_1 \not\equiv c_2$ irreduzible Grundterme.

Ein Beispiel (4)

Sei

E erweitert um („Sortieren durch Einfügen“)

$\text{ins}(x, \text{nil}) = x.\text{nil}$
$\text{ins}(x, y.l) = x.y.l \Leftarrow x \leq y = \text{true}$
$\text{ins}(x, y.l) = y.\text{ins}(x, l) \Leftarrow x \leq y = \text{false}$
$\text{sorted}(\text{nil}) = \text{true}$
$\text{sorted}(x.\text{nil}) = \text{true}$
$\text{sorted}(x.y.l) = \text{sorted}(y.l) \Leftarrow x \leq y = \text{true}$
$\text{sorted}(x.y.l) = \text{false} \Leftarrow x \leq y = \text{false}$

Beachte E gerichtet zu R ist konfluent und terminierend.

$C = \{\text{true}, \text{false}, 0, s, \text{nil}, .\}$ ist eine Konstruktormenge.

Behauptung $e \equiv \text{sorted}(l) \Rightarrow \text{sorted}(\text{ins}(x, l))$

Das Definitionsschema für sorted legt nahe:

a) Induktion nach l mit $>_{TT}$ als Induktionsordnung.
b) 3 Fälle: $l \equiv \text{nil}$ und $l \equiv y.\text{nil}$ als Basisfälle und $l \equiv y.z.l'$ für den Induktionsschritt.

Ein Beispiel (5)

1. $\sigma(l) \equiv \text{nil}$:
Wegen $\text{sorted}(\text{ins}(x, \text{nil})) =_E \text{sorted}(x.\text{nil}) =_E \text{true}$ gilt $\sigma(e)$.

2. $\sigma(l) \equiv y.\text{nil}$:
Wir zeigen $\text{sorted}(\text{ins}(x, y.\text{nil})) =_E \text{true}$, dann gilt $\sigma(e)$.
Beachte, dass $x \leq y = \text{true} \vee x \leq y = \text{false}$ ein Lemma ist.
Wir machen eine Fallunterscheidung.

2.1 $x \leq y = \text{true}$: Es gilt
$\text{sorted}(\text{ins}(x, y.\text{nil})) =_E \text{sorted}(x.y.\text{nil}) =_E \text{sorted}(y.\text{nil}) =_E \text{true}$.

2.2 $x \leq y = \text{false}$, also $y \leq x = \text{true}$ (siehe oben):
$\text{sorted}(\text{ins}(x, y.\text{nil})) =_E \text{sorted}(y.x.\text{nil}) =_E \text{true}$

3. $\sigma(l) \equiv y.z.l$:
Zu zeigen ist $\sigma(e) \equiv \text{sorted}(y.z.l) \Rightarrow \text{sorted}(\text{ins}(x, y.z.l))$.
Wir benutzen wieder die Fallunterscheidung.

Ein Beispiel (6)

3.1 $y \le z =$ false: Wegen sorted$(y.z.l) \neq_E$ true gilt $\sigma(e)$.

3.2 $y \le z =$ true: Es gilt sorted$(y.z.l) =_E$ sorted$(z.l)$.

3.2.1 $x \le y =$ true:
$\sigma(e)$ reduziert sich mit E zu sorted$(z.l) \Rightarrow$ sorted$(z.l)$.
Dies gilt in I(spec).

3.2.2 $x \le y =$ false, also $y \le x =$ true.

3.2.2.1 $x \le z =$ true:
Umschreiben von $\sigma(e)$ mit E liefert sorted$(z.l) \Rightarrow$ sorted$(z.l)$.
Dies gilt in I(spec).

3.2.2.2 $x \le z =$ false:
Umschreiben von $\sigma(e)$ liefert sorted$(z.l) \Rightarrow$ sorted(ins$(x, z.l)$).
(denn sorted(ins$(x, y.z.l)$) $=_E$ sorted$(y$.ins$(x, z.l))$ $=_E$ sorted$(y.z$.ins$(x.l))$ $=_E$ sorted$(z$.ins$(x, l))$ $=_E$ sorted(ins$(x, z.l)$)).
Dies gilt nach IV.

Flexible Wahl der Induktionsordnung

Beachte

Einzige Voraussetzung an die Induktionsordnung ist: $\succ_i$ ist Noethersch.

Sei $F = \{0, s, \max\}$, $C = \{0, s\}$

E: $$\max(x, 0) = x$$ $$\max(0, y) = y$$ $$\max(s(x), s(y)) = s(\max(x, y))$$

Dann ist C eine Konstruktormenge.

Behauptung

Es gilt $\max(x, y) = \max(y, x)$ in $\mathcal{I}$.

Beweis:

Sei $\sigma \succ_i \tau$ gdw $\{\sigma(x), \sigma(y)\} \gg_{TT} \{\tau(x), \tau(y)\}$.

1. Fall: $\sigma = \{y \leftarrow 0\}$.

Es gilt: $\max(x, 0) =_E x =_E \max(0, x)$.

2. Fall: $\sigma = \{x \leftarrow 0\}$.

Es gilt: $\max(0, y) =_E y =_E \max(y, 0)$.

3. Fall: $\sigma = \{x \leftarrow s(x), y \leftarrow s(y)\}$.

Es gilt: $\max(s(x), s(y)) =_E s(\max(x, y)) =_{IV} s(\max(y, x)) =_E \max(s(y), s(x))$.