Programare logică

Rescriere

Contexte

(S, Σ) signatură, X mulţime de variabile

- Pentru $t \in T_\Sigma(X)$ şi $y \in X$ notăm
  $nr_y(t) =$ numărul de apariţii ale lui y în t

- Dacă $z \notin X$ atunci un termen $c \in T_\Sigma(X \cup \{z\})$ se numeşte context dacă $nr_z(c) = 1$.

- Dacă $t_0 \in T_\Sigma(X)$ şi t_0 are acelaşi sort cu z, atunci notăm
  $$c[z \leftarrow t_0] := \{z \leftarrow t_0\}(c)$$
  pentru un context $c \in T_\Sigma(X \cup \{z\})$

Regula de deducţie SR_Γ

(S, Σ, Γ) specificaţie

$$\text{SR}_\Gamma \quad \frac{(\forall X)\theta(u) \doteq_{s'} \theta(v) \text{ or. } u \doteq_s v \in H}{(\forall X)c[z \leftarrow \theta(l)] \doteq_{s''} c[z \leftarrow \theta(r)]}$$

unde $(\forall Y)l \doteq_s r \ if \ H \in \Gamma$, $\theta : Y \to T_\Sigma(X)$ substituţie

$$c \in T_\Sigma(X \cup \{z\})_{s''},\ z \notin X,\ nr_z(c) = 1$$

Propoziţie. SR_Γ este regulă de deducţie corectă.

Teoremă. Sunt echivalente:

(1) $\Gamma \vdash (\forall X)t \doteq_s t'$

(2) $\Gamma \vdash_{\text{R,S,T, SR}_\Gamma} (\forall X)t \doteq_s t'$

SR$_E$

(S, Σ) signatură, E mulţime de ecuaţii necondiţionate

$$\text{SR}_E \quad \frac{}{(\forall X)c[z \leftarrow \theta(l)] \doteq_{s''} c[z \leftarrow \theta(r)]}$$

unde $(\forall Y)l \doteq_s r \in E$, $\theta : Y \to T_\Sigma(X)$ substituţie

$c \in T_\Sigma(X \cup \{z\})_{s''}$, $z \notin X$, $nr_z(c) = 1$

Exemplu

$E = \{x + 0 \doteq x, x + succ(y) \doteq succ(x + y)\}$

$E \vdash_{\mathbf{SR}_\Gamma} 0 + succ(0) \doteq succ(0)$

(1) $x + succ(y) \doteq succ(x + y) \in E$

(2) $0 + succ(0) \doteq succ(0 + 0)$ $(1, Sub\{x, y \leftarrow 0\})$

(3) $x + 0 = x \in E$

(4) $0 + 0 \doteq 0$ $(3, Sub\{x \leftarrow 0\})$

(5) $succ(0 + 0) \doteq succ(0)$ $(4, C)$

(6) $0 + succ(0) \doteq succ(0)$ $(2, 5, T)$

(1') $x + succ(y) \doteq succ(x + y) \in E$

(2') $0 + succ(0) \doteq succ(0 + 0)$ $(1', SR_E, c := z, \theta := \{x, y \leftarrow 0\})$

(3') $x + 0 = x \in E$

(4') $succ(0 + 0) \doteq succ(0)$ $(3', SR_E, c := succ(z), \theta := \{x \leftarrow 0\})$

(5') $0 + succ(0) \doteq succ(0)$ $(2', 4', T)$

(S, Σ, Γ) **specificaţie,** X **mulţime de variabile,** t**,** $t' \in T_\Sigma(X)_s$

Teoremă. Sunt echivalente:

(1) $\Gamma \vdash (\forall X)t \doteq_s t'$

(2) $\Gamma \vdash_{\text{R,S,T, SR}_\Gamma} (\forall X)t \doteq_s t'$

Demonstraţie. Fie $\sim_\Gamma$ echivalenţa sintactică.
Definim $\sim_{SR} \subseteq T_\Sigma(X) \times T_\Sigma(X)$ prin
$t \sim_{SR} t' \Leftrightarrow \Gamma \vdash_{\text{R,S,T, SR}_\Gamma} (\forall X)t \doteq_s t'$

(2) $\Rightarrow$ (1) ($\sim_{SR} \subseteq \sim_\Gamma$) Rezultă din corectitudinea regulii SR$_\Gamma$.

(1) $\Rightarrow$ (2) ($\sim_\Gamma \subseteq \sim_{SR}$) Dacă $t \sim_\Gamma t'$, atunci $\Gamma \models (\forall X)t \doteq_s t'$.
Demonstrăm că $\sim_{SR}$ este congruenţă care verifică proprietatea $CS(\Gamma, T_\Sigma(X))$. Atunci $T_\Sigma(X)/_{\sim_{SR}} \models \Gamma$, deci $T_\Sigma(X)/_{\sim_{SR}} \models (\forall X)t \doteq_s t'$.
Dacă $[\cdot]_{SR} : T_\Sigma(X) \to T_\Sigma(X)/_{\sim_{SR}}$ este surjecţia canonică, atunci $[t]_{SR} = [t']_{SR}$, deci $t \sim_{SR} t'$.

Rescrierea

(S, Σ) signatură

- O regulă de rescriere este formată dintr-o mulţime de variabile Y şi doi termeni l, r de acelaşi sort din $T_\Sigma(Y)$ a.î.
  - l nu este variabilă
  - $Var(r) \subseteq Var(l) = Y$.

  Vom nota $l \to r$ ($l \to_s r$).
- Un sistem de rescriere (TRS) este o mulţime finită de reguli de rescriere.
- $R = \{x + 0 \to x, x + succ(y) \to succ(x + y)\}$

Relaţia $\rightarrow_R$

- Fie R un sistem de rescriere. Dacă t, $t' \in T_\Sigma(X)_s$ definim relaţia $t \rightarrow_R t'$ astfel:

$$
\boxed{
\begin{array}{lll}
t \rightarrow_R t' & \Leftrightarrow & t \text{ este } c[z \leftarrow \theta(l)] \text{ şi} \\
 & & t' \text{ este } c[z \leftarrow \theta(r)] \text{, unde} \\
 & & c \in T_\Sigma(X \cup \{z\}) \text{, } z \not\in X \text{, } nr_z(c) = 1 \\
 & & l \rightarrow r \in R \text{ cu } Var(l) = Y \text{,} \\
 & & \theta : Y \rightarrow T_\Sigma(X) \text{ este o substituţie.}
\end{array}
}
$$

- $t \rightarrow_R t'$ dacă şi numai dacă t' se obţine din t înlocuind o instanţă a lui l cu o instanţă a lui r.

Sistemul de rescriere determinat de E

(S, Σ) specificaţie, E mulţime de ecuaţii necondiţionate

Dacă $\{l \to r \mid (\forall Y) l \doteq_s r \in E\}$ este un sistem de rescriere, atunci $R_E := \{l \to r \mid (\forall Y) l \doteq_s r \in E\}$ este sistemul de rescriere determinat de E.

În acest caz vom nota $\to_E := \to_{R_E}$

$E = \{x + 0 \doteq x, x + succ(y) \doteq succ(x + y)\}$,

$R_E = \{x + 0 \to x, x + succ(y) \to succ(x + y)\}$

$0 + succ(0) \to_E succ(0 + 0)$

$(l := x + succ(y), r := succ(x + y), c := z, \theta := \{x, y \leftarrow 0\})$

$succ(0 + 0) \to_E succ(0)$

$(l := x + 0, r := x, c := succ(z), \theta := \{x \leftarrow 0\})$

ecuaţiile devin reguli de rescriere prin orientare

CafeObj

```
mod! MYNATSTR{ protecting (MYNAT) [ Nat < NatStr ]
op nil : -> NatStr
op _;_ : NatStr NatStr -> NatStr { assoc }
var L : NatStr
eq nil ;  L = L .
eq L ; nil = L . }
%MYNATSTR> reduce (nil ; s 0) ; s s 0 ; nil ; 0 .
1>[1] rule: eq (nil ; L:NatStr) = L
    { L:NatStr |-> ((s 0) ; ((s (s 0)) ; (nil ; 0))) }
1<[1] ((nil ; (s 0)) ; ((s (s 0)) ; (nil ; 0))) -->
              ((s 0) ; ((s (s 0)) ; (nil ; 0)))
1>[2] rule: eq (A1 ; (nil ; L:NatStr)) = (A1 ; L)
    { L:NatStr |-> 0, A1 |-> ((s 0) ; (s (s 0))) }
1<[2] (((s 0) ; (s (s 0))) ; (nil ; 0)) -->
      (((s 0) ; (s (s 0))) ; 0)
```

Exemplu

$E = \{x + 0 \doteq x, x + succ(y) \doteq succ(x + y)\}$

- $E \vdash_{\Gamma} 0 + succ(0) \doteq succ(0)$

- $E \vdash_{\text{R,S,T, SR}_\Gamma} 0 + succ(0) \doteq succ(0)$

(1’) $x + succ(y) \doteq succ(x + y) \in E$
(2’) $0 + succ(0) \doteq succ(0 + 0)$ $(1', SR_E, c := z, \theta := \{x, y \leftarrow 0\})$
(3’) $x + 0 = x \in E$
(4’) $succ(0 + 0) \doteq succ(0)$ $(3', SR_E, c := succ(z), \theta := \{x \leftarrow 0\})$
(5’) $0 + succ(0) \doteq succ(0)$ $(2', 4', T)$

- $0 + succ(0) \rightarrow_E succ(0 + 0) \rightarrow_E succ(0)$