Curs 11

Cuprins

Sisteme de rescriere abstracte

TRS - Term Rewriting System

- O regulă de rescriere (peste Y) este formată din $l, r \in T_\Sigma(Y)_s$ a.î.:
  1. l nu este variabilă,
  2. $Var(r) \subseteq Var(l) = Y$.

- Un sistem de rescriere (TRS) este o mulțime finită de reguli de rescriere.

$$
\boxed{
\begin{array}{lcl}
t \to_R t' & \Leftrightarrow & t \text{ este } c[z \leftarrow \theta_s(l)] \text{ și} \\
 & & t' \text{ este } c[z \leftarrow \theta_s(r)], \text{ unde} \\
 & & c \in T_\Sigma(X \cup \{z\}) \text{ context}, \\
 & & l \to_s r \in R \text{ cu } Var(l) = Y, \\
 & & \theta : Y \to T_\Sigma(X) \text{ substituție}
\end{array}
}
$$

Sisteme de rescriere abstracte

- Terminarea unui sistem de rescriere este nedecidabilă.
  - echivalentă cu oprirea mașinilor Turing
- Pentru sisteme de rescriere particulare putem decide asupra terminării.
  - diverse metode
- Pentru sisteme de rescriere care se termină, confluența este decidabilă.
  - algoritmul Knuth-Bendix

Terminare

Arborele de reducere

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

- Arborele de reducere al termenului t este definit astfel:
  - rădăcina arborelui are eticheta t,
  - descendenții nodului cu eticheta u sunt etichetați cu termenii u' care verifică $u \to_R u'$.

- Orice nod al unui arbore de reducere are un număr finit de descendenți deoarece R este o mulțime finită.

Arborele de reducere

Exemplu

- $R = \{x + 0 \to x, x + s(y) \to s(x + y)\}$
- Arborele de reducere al termenului $s(0) + s(0 + 0)$:

$$
\begin{tikzcd}
 & s(0) + s(0 + 0) \arrow[dl] \arrow[dr] & \\
s(0) + s(0) \arrow[d] & & s(s(0) + (0 + 0)) \arrow[d] \\
s(s(0) + 0) \arrow[d] & & s(s(0) + 0) \arrow[d] \\
s(s(0)) & & s(s(0))
\end{tikzcd}
$$

Terminare

Propoziție

Sunt echivalente:

1. R este noetherian,
2. oricărui termen t îi poate fi asociat un număr natural $\mu(t) \in \mathbb{N}$ astfel încât $t \to_R t'$ implică $\mu(t) > \mu(t')$.

Terminare

Propoziție

Sunt echivalente:

1. R este noetherian,
2. oricărui termen t îi poate fi asociat un număr natural $\mu(t) \in \mathbb{N}$ astfel încât $t \rightarrow_R t'$ implică $\mu(t) > \mu(t')$.

Demonstrație

$(2 \Rightarrow 1)$ $\mathbb{N}$ nu conține lanțuri infinite $n_1 > n_2 > \cdots > n_k > \cdots$.

Terminare

Propoziție

Sunt echivalente:

1. R este noetherian,
2. oricărui termen t îi poate fi asociat un număr natural $\mu(t) \in \mathbb{N}$ astfel încât $t \to_R t'$ implică $\mu(t) > \mu(t')$.

Demonstrație

$(2 \Rightarrow 1)$ $\mathbb{N}$ nu conține lanțuri infinite $n_1 > n_2 > \cdots > n_k > \cdots$.

$(1 \Rightarrow 2)$ Într-un sistem de rescriere noetherian orice termen are un arbore de reducere finit și definim

$$\mu(t) = \text{înălțimea arborelui de reducere asociat lui } t.$$

Evident $t \to_R t' \Rightarrow \mu(t) > \mu(t')$. $\square$

Ordine de reducere

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

O ordine strictă $>$ pe $T_\Sigma(X)$ se numește o ordine de reducere dacă:

Ordine de reducere

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

O ordine strictă $>$ pe $T_\Sigma(X)$ se numește o ordine de reducere dacă:

- este *well-founded*:
  - orice mulțime de termeni are un cel mai mic element în raport cu relația $>$

Ordine de reducere

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

O ordine strictă $>$ pe $T_\Sigma(X)$ se numește o ordine de reducere dacă:

- este *well-founded*:
  - orice mulțime de termeni are un cel mai mic element în raport cu relația $>$
- este compatibilă cu operațiile:
  - dacă $s_1 > s_2$, atunci $\sigma(t_1, \ldots, t_{i-1}, s_1, t_{i+1}, \ldots, t_n) > \sigma(t_1, \ldots, t_{i-1}, s_2, t_{i+1}, \ldots, t_n)$, pentru orice $\sigma : s_1 \ldots s_n \to s$

Ordine de reducere

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

O ordine strictă $>$ pe $T_\Sigma(X)$ se numește o ordine de reducere dacă:

- este *well-founded*:
  - orice mulțime de termeni are un cel mai mic element în raport cu relația $>$
- este compatibilă cu operațiile:
  - dacă $s_1 > s_2$, atunci $\sigma(t_1, \ldots, t_{i-1}, s_1, t_{i+1}, \ldots, t_n) > \sigma(t_1, \ldots, t_{i-1}, s_2, t_{i+1}, \ldots, t_n)$, pentru orice $\sigma : s_1 \ldots s_n \to s$
- este închisă la substituții:
  - dacă $s_1 > s_2$, atunci $\theta(s_1) > \theta(s_2)$, pentru orice substituție θ

Ordine de reducere

Exemplu

Relația de ordine strictă $>$ pe $T_\Sigma(X)$ definită prin

$$s > t \quad \text{ddacă} \quad |s| > |t|,$$

unde $|t|$ este lungimea termenului t (numărul de simboluri din t)

Ordine de reducere

Exemplu

Relația de ordine strictă $>$ pe $T_\Sigma(X)$ definită prin

$$s > t \quad \text{ddacă} \quad |s| > |t|,$$

unde $|t|$ este lungimea termenului t (numărul de simboluri din t)

- este well-founded și compatibilă cu operațiile

Ordine de reducere

Exemplu

Relația de ordine strictă $>$ pe $T_\Sigma(X)$ definită prin

$$s > t \quad \text{ddacă} \quad |s| > |t|,$$

unde $|t|$ este lungimea termenului t (numărul de simboluri din t)

- este well-founded și compatibilă cu operațiile
- în general, nu este închisă la substituții:

$$|f(f(x,x),y)| = 5 > 3 = |f(y,y)|$$

Ordine de reducere

Exemplu

Relația de ordine strictă $>$ pe $T_\Sigma(X)$ definită prin

$$s > t \quad \text{ddacă} \quad |s| > |t|,$$

unde $|t|$ este lungimea termenului t (numărul de simboluri din t)

- este well-founded și compatibilă cu operațiile
- în general, nu este închisă la substituții:

$$|f(f(x,x),y)| = 5 > 3 = |f(y,y)|$$

dar pentru substituția $\theta(y) = f(x,x)$ avem

$$\begin{aligned} |\theta(f(f(x,x),y)| &= |f(f(x,x),f(x,x)| = 7 \\ |\theta(f(y,y))| &= |f(f(x,x),f(x,x)| = 7 \end{aligned}$$

Ordine de reducere

Exemplu

Relația de ordine strictă $>$ pe $T_\Sigma(X)$ definită prin

$$s > t \quad \text{ddacă} \quad |s| > |t|,$$

unde $|t|$ este lungimea termenului t (numărul de simboluri din t)

- este well-founded și compatibilă cu operațiile
- în general, nu este închisă la substituții:

$$|f(f(x,x),y)| = 5 > 3 = |f(y,y)|$$

dar pentru substituția $\theta(y) = f(x,x)$ avem

$$\begin{aligned} |\theta(f(f(x,x),y)| &= |f(f(x,x),f(x,x)| = 7 \\ |\theta(f(y,y))| &= |f(f(x,x),f(x,x)| = 7 \end{aligned}$$

Deci nu este, în general, ordine de reducere.

Ordine de reducere

Exemplu

Relația de ordine strictă $>$ pe $T_\Sigma(X)$ definită prin

$$s > t \quad \text{ddacă} \quad |s| > |t| \text{ și } nr_x(s) \geq nr_x(t), \text{ pentru orice } x \in X$$

este o ordine de reducere.

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_\Sigma(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_\Sigma(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

$s >_{lpo} t$ ddacă

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_{\Sigma}(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

$s >_{lpo} t$ ddacă

(LPO1) $t \in X$ și $s \neq t$, sau

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_\Sigma(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

$s >_{lpo} t$ ddacă

(LPO1) $t \in X$ și $s \neq t$, sau

(LPO2) $s = f(s_1, \ldots, s_m)$, $t = g(t_1, \ldots, t_n)$ și

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_\Sigma(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

$s >_{lpo} t$ ddacă

(LPO1) $t \in X$ și $s \neq t$, sau

(LPO2) $s = f(s_1, \ldots, s_m)$, $t = g(t_1, \ldots, t_n)$ și

(LPO2a) există i, $1 \leq i \leq m$ astfel încât $s_i \geq_{lpo} t$, sau

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_\Sigma(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

$s >_{lpo} t$ ddacă

(LPO1) $t \in X$ și $s \neq t$, sau

(LPO2) $s = f(s_1, \ldots, s_m)$, $t = g(t_1, \ldots, t_n)$ și

(LPO2a) există i, $1 \leq i \leq m$ astfel încât $s_i \geq_{lpo} t$, sau

(LPO2b) $f > g$ și $s >_{lpo} t_j$, pentru orice j, $1 \leq j \leq n$

Ordine de reducere

Exemplu

Ordinea lexicografică $>_{lpo}$ indusă pe mulțimea de termeni $T_{\Sigma}(X)$ de o relație de ordine strictă $>$ pe signatură este o ordine de reducere.

$s >_{lpo} t$ ddacă

(LPO1) $t \in X$ și $s \neq t$, sau

(LPO2) $s = f(s_1, \ldots, s_m)$, $t = g(t_1, \ldots, t_n)$ și

(LPO2a) există i, $1 \leq i \leq m$ astfel încât $s_i \geq_{lpo} t$, sau

(LPO2b) $f > g$ și $s >_{lpo} t_j$, pentru orice j, $1 \leq j \leq n$

(LPO2c) $f = g$, $s >_{lpo} t_j$, pentru orice j, $1 \leq j \leq n$, și există i, $1 \leq i \leq m$ astfel încât $s_1 = t_1, \ldots, s_{i-1} = t_{i-1}$ și $s_i >_{lpo} t_i$.

Ordine de reducere

Exemplu

- Fie $S = \{s\}$ și $\Sigma = \{f : s\ s \to s, i : s \to s, e : -s\}$
- Considerăm $i > f > e$.
- Atunci avem:
  - $f(x, e) >_{lpo} x$ din (LPO1)
  - $i(e) >_{lpo} e$ din (LPO2a) deoarece $e \geq_{lpo} e$
  - $i(f(x, y)) >_{lpo} f(i(y), i(x))$ din (LPO2b) deoarece $i > f$ si, din (LPO2c), avem $i(f(x, y)) >_{lpo} i(y)$ și $i(f(x, y)) >_{lpo} i(x)$

(LPO1) $t \in X$ și $s \neq t$, sau

(LPO2) $s = f(s_1, \ldots, s_m)$, $t = g(t_1, \ldots, t_n)$ și

(LPO2a) există i, $1 \leq i \leq m$ astfel încât $s_i \geq_{lpo} t$, sau

(LPO2b) $f > g$ și $s >_{lpo} t_j$, pentru orice j, $1 \leq j \leq n$

(LPO2c) $f = g$, $s >_{lpo} t_j$, pentru orice j, $1 \leq j \leq n$, și există i, $1 \leq i \leq m$ astfel încât $s_1 = t_1, \ldots, s_{i-1} = t_{i-1}$ și $s_i >_{lpo} t_i$.

Ordine de reducere

Teorema (*)

Următoarele sunt echivalente:

1. *Un sistem de rescrire R este noetherian.*
2. *Există o ordine de reducere $>$ care satisface $l > r$ pentru orice $l \to r \in R$.*

Confluență. Perechi critice.

Perechi critice

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

Fie $l_1 \to r_1$, $l_2 \to r_2 \in R$ astfel încât:

Perechi critice

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

Fie $l_1 \to r_1$, $l_2 \to r_2 \in R$ astfel încât:

1. $Var(l_1) \cap Var(l_2) = \emptyset$,

Perechi critice

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

Fie $l_1 \to r_1$, $l_2 \to r_2 \in R$ astfel încât:

1. $Var(l_1) \cap Var(l_2) = \emptyset$,
2. există un subtermen t al lui l_1 care nu este variabilă
   ($l_1 = c[z \leftarrow t]$, unde $nr_z(c) = 1$, t nu este variabilă)

Perechi critice

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

Fie $l_1 \to r_1$, $l_2 \to r_2 \in R$ astfel încât:

1. $Var(l_1) \cap Var(l_2) = \emptyset$,
2. există un subtermen t al lui l_1 care nu este variabilă
   ($l_1 = c[z \leftarrow t]$, unde $nr_z(c) = 1$, t nu este variabilă)
3. există θ c.g.u pentru t și l_2 (i.e. $\theta(t) = \theta(l_2)$).

Perechi critice

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

Definitie

Fie $l_1 \to r_1$, $l_2 \to r_2 \in R$ astfel încât:

1. $Var(l_1) \cap Var(l_2) = \emptyset$,
2. există un subtermen t al lui l_1 care nu este variabilă
   ($l_1 = c[z \leftarrow t]$, unde $nr_z(c) = 1$, t nu este variabilă)
3. există θ c.g.u pentru t și l_2 (i.e. $\theta(t) = \theta(l_2)$).

Perechea $(\theta(r_1), \theta(c)[z \leftarrow \theta(r_2)])$ se numește pereche critică.

$$\begin{tikzcd} & \theta(l_1) \arrow[dl, "R"'] \arrow[dr, "R"] & \\ \theta(r_1) & & \theta(c)[z \leftarrow \theta(r_2)] \end{tikzcd}$$

Exemplu

Exemplu

$R = \{f(f(x,y),u) \to f(x, f(y,u)),\ f(i(x_1), x_1) \to e\}$

1. $Var(f(f(x,y),u)) = \{x,y,u\}$ și $Var(f(i(x_1),x_1)) = \{x_1\}$
2. Luăm subtermenul $t = f(x,y)$ al lui $l_1 = f(f(x,y),u)$
   - $l_1 = c[z \leftarrow t]$ pt. contextul $c = f(z,u)$
3. $\theta = \{x \mapsto i(x_1),\ y \mapsto x_1\}$ c.g.u. pt. t și $l_2 = f(i(x_1), x_1)$.

$$
\begin{tikzcd}
 & f(f(i(x_1),x_1),u) \arrow[dl, "R"'] \arrow[dr, "R"] & \\
f(i(x_1), f(x_1,u)) & & f(e,u)
\end{tikzcd}
$$

Pereche critică: $(f(i(x_1), f(x_1,u)), f(e,u))$

Confluență și perechi critice

Fie (S, Σ) o signatură, Y mulțime de variabile și R un TRS.

$$\begin{tikzcd}[column sep=small]
 & \theta(l_1) \arrow[dl] \arrow[dr] & \\
\theta(r_1) & & \theta(c)[z \leftarrow \theta(r_2)]
\end{tikzcd}$$

$l_1 \to r_1,\ l_2 \to r_2 \in R,$
$Var(l_1) \cap Var(l_2) = \emptyset,$
$l_1 = c[z \to t],$
$\theta(t) = \theta(l_2)$ c.g.u

Teorema (Teorema Perechilor Critice *)

Dacă R este noetherian, atunci sunt echivalente:

1. *R este confluent,*
2. *$t_1 \downarrow_R t_2$ pentru orice pereche critică (t_1, t_2).*

Consecință

Corolar

Confluența unui TRS noetherian este decidabilă.

Algoritm:

- pt. or. pereche de reguli de rescriere $l_1 \to r_1$ și $l_2 \to r_2$
- se încearcă generarea perechilor critice (t_1, t_2)
- pt. or. pereche critică (t_1, t_2), se arată că $t_1 \downarrow_R t_2$

Exemplu

Exemplu

$R = \{f(f(x)) \to x\}$ este confluent.

Exemplu

Exemplu

$R = \{f(f(x)) \to x\}$ este confluent.

- R este noetherian.

Exemplu

Exemplu

$$R = \{f(f(x)) \to x\} \text{ este confluent.}$$

- R este noetherian.
- Determinăm perechile critice:

Exemplu

Exemplu

$$R = \{f(f(x)) \to x\} \text{ este confluent.}$$

- R este noetherian.
- Determinăm perechile critice:
  - Regulile $l_1 = f(f(x)) \to x = r_1$ și $l_2 = f(f(y)) \to y = r_2$.

Exemplu

Exemplu

$$R = \{f(f(x)) \to x\} \text{ este confluent.}$$

- R este noetherian.
- Determinăm perechile critice:
  - Regulile $l_1 = f(f(x)) \to x = r_1$ și $l_2 = f(f(y)) \to y = r_2$.
    Subtermenii lui l_1 care nu sunt variabile sunt $f(f(x))$ și $f(x)$.

Exemplu

Exemplu

$$R = \{f(f(x)) \to x\} \text{ este confluent.}$$

- R este noetherian.
- Determinăm perechile critice:
  - Regulile $l_1 = f(f(x)) \to x = r_1$ și $l_2 = f(f(y)) \to y = r_2$.
    Subtermenii lui l_1 care nu sunt variabile sunt $f(f(x))$ și $f(x)$.
    - $t := f(f(x))$, $c = z$, $\theta := \{x \leftarrow y\}$
      Perechea critică: $\theta(r_1) = y$, $\theta(c)[z \leftarrow \theta(r_2)] = y$

Exemplu

Exemplu

$$R = \{f(f(x)) \to x\} \text{ este confluent.}$$

- R este noetherian.
- Determinăm perechile critice:
  - Regulile $l_1 = f(f(x)) \to x = r_1$ și $l_2 = f(f(y)) \to y = r_2$.
    Subtermenii lui l_1 care nu sunt variabile sunt $f(f(x))$ și $f(x)$.
    - $t := f(f(x))$, $c = z$, $\theta := \{x \leftarrow y\}$
      Perechea critică: $\theta(r_1) = y$, $\theta(c)[z \leftarrow \theta(r_2)] = y$
    - $t := f(x)$, $c = f(z)$, $\theta := \{x \leftarrow f(y)\}$
      Perechea critică: $\theta(r_1) = f(y)$, $\theta(c)[z \leftarrow \theta(r_2)] = f(y)$

Exemplu

Exemplu

$$R = \{f(f(x)) \to x\} \text{ este confluent.}$$

- R este noetherian.
- Determinăm perechile critice:
  - Regulile $l_1 = f(f(x)) \to x = r_1$ și $l_2 = f(f(y)) \to y = r_2$.
    Subtermenii lui l_1 care nu sunt variabile sunt $f(f(x))$ și $f(x)$.
    - $t := f(f(x))$, $c = z$, $\theta := \{x \leftarrow y\}$
      Perechea critică: $\theta(r_1) = y$, $\theta(c)[z \leftarrow \theta(r_2)] = y$
    - $t := f(x)$, $c = f(z)$, $\theta := \{x \leftarrow f(y)\}$
      Perechea critică: $\theta(r_1) = f(y)$, $\theta(c)[z \leftarrow \theta(r_2)] = f(y)$
- Perechile critice sunt (y, y) și $(f(y), f(y))$.

Exemplu

Exemplu

$R = \{f(f(x)) \to x\}$ este confluent.

- R este noetherian.
- Determinăm perechile critice:
  - Regulile $l_1 = f(f(x)) \to x = r_1$ și $l_2 = f(f(y)) \to y = r_2$.
    Subtermenii lui l_1 care nu sunt variabile sunt $f(f(x))$ și $f(x)$.
    - $t := f(f(x))$, $c = z$, $\theta := \{x \leftarrow y\}$
      Perechea critică: $\theta(r_1) = y$, $\theta(c)[z \leftarrow \theta(r_2)] = y$
    - $t := f(x)$, $c = f(z)$, $\theta := \{x \leftarrow f(y)\}$
      Perechea critică: $\theta(r_1) = f(y)$, $\theta(c)[z \leftarrow \theta(r_2)] = f(y)$
- Perechile critice sunt (y, y) și $(f(y), f(y))$.
- Deoarece $y \downarrow y$ și $f(y) \downarrow f(y)$, sistemul de rescriere R este confluent.

Algoritmul Knuth-Bendix

Algoritmul Knuth-Bendix

- Procedură pentru a completa un TRS noetherian.

- Intrare: R un sistem de rescriere (TRS) noetherian.
- Ieșire:
  - T un sistem de rescriere (TRS) = completarea lui R.
  - eșec

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: $T := R$ și $>$ ordine de reducere pentru T

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: T := R și $>$ ordine de reducere pentru T
- Se execută următorii pași, cât timp este posibil:

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: T := R și > ordine de reducere pentru T
- Se execută următorii pași, cât timp este posibil:
  1. CP := CP(T) = $\{(t_1, t_2) \mid (t_1, t_2)$ pereche critică în $T\}$

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: T := R și $>$ ordine de reducere pentru T
- Se execută următorii pași, cât timp este posibil:
  1. CP := CP(T) $= \{(t_1, t_2) \mid (t_1, t_2)$ pereche critică în $T\}$
  2. Dacă $t_1 \downarrow t_2$, oricare $(t_1, t_2) \in CP$, atunci STOP *(T completarea lui R)*.

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: T := R și $>$ ordine de reducere pentru T
- Se execută următorii pași, cât timp este posibil:
  1. CP := CP(T) $= \{(t_1, t_2) \mid (t_1, t_2)$ pereche critică în $T\}$
  2. Dacă $t_1 \downarrow t_2$, oricare $(t_1, t_2) \in CP$, atunci STOP *(T completetarea lui R)*.
  3. Dacă $(t_1, t_2) \in CP$, $t_1 \not\downarrow t_2$ atunci:
     - dacă $fn(t_1) > fn(t_2)$ atunci T := T $\cup \{fn(t_1) \to fn(t_2)\}$,
     - dacă $fn(t_2) > fn(t_1)$ atunci T := T $\cup \{fn(t_2) \to fn(t_1)\}$,
     - altfel, STOP *(completare eșuată)*.

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: T := R și $>$ ordine de reducere pentru T
- Se execută următorii pași, cât timp este posibil:
  1. CP := CP(T) $= \{(t_1, t_2) \mid (t_1, t_2)$ pereche critică în $T\}$
  2. Dacă $t_1 \downarrow t_2$, oricare $(t_1, t_2) \in CP$, atunci STOP *(T completarea lui R)*.
  3. Dacă $(t_1, t_2) \in CP$, $t_1 \not\downarrow t_2$ atunci:
     - dacă $fn(t_1) > fn(t_2)$ atunci T := T $\cup \{fn(t_1) \to fn(t_2)\}$,
     - dacă $fn(t_2) > fn(t_1)$ atunci T := T $\cup \{fn(t_2) \to fn(t_1)\}$,
     - altfel, STOP *(completare eșuată)*.
- IEȘIRE: T completarea lui R sau eșec.

Algoritmul Knuth-Bendix

- INTRARE: R un sistem de rescriere (TRS) noetherian.
- INIȚIALIZARE: T := R și $>$ ordine de reducere pentru T
- Se execută următorii pași, cât timp este posibil:
  1. CP := CP(T) $= \{(t_1, t_2) \mid (t_1, t_2)$ pereche critică în $T\}$
  2. Dacă $t_1 \downarrow t_2$, oricare $(t_1, t_2) \in CP$, atunci STOP *(T completarea lui R)*.
  3. Dacă $(t_1, t_2) \in CP$, $t_1 \not\downarrow t_2$ atunci:
     - dacă $fn(t_1) > fn(t_2)$ atunci $\mathrm{T} := \mathrm{T} \cup \{fn(t_1) \to fn(t_2)\}$,
     - dacă $fn(t_2) > fn(t_1)$ atunci $\mathrm{T} := \mathrm{T} \cup \{fn(t_2) \to fn(t_1)\}$,
     - altfel, STOP *(completare eșuată)*.
- IEȘIRE: T completarea lui R sau eșec.

Atenție! Succesul completării depinde de ordinea de reducere $>$.

Exemplu

Exemplu

- $S := \{s\}$, $\Sigma := \{* : ss \to s\}$, $E := \{\forall\{x, y, v\}(x * y) * (y * v) \doteq y\}$
- INIȚIALIZARE:
  - $T = R_E := \{(x * y) * (y * v) \to y\}$,
  - Ordine de reducere:
    $s > t$ ddacă $|s| > |t|$ și $nr_x(s) \geq nr_x(t)$, pentru orice $x \in X$

Exemplu

Exemplu

- Determinăm perechile critice pentru

  $l_1 := (x * y) * (y * v),\ r_1 := y,\ l_2 := (x' * y') * (y' * v'),\ r_2 := y'$

Exemplu

Exemplu

- Determinăm perechile critice pentru

  $l_1 := (x * y) * (y * v),\ r_1 := y,\ l_2 := (x' * y') * (y' * v'),\ r_2 := y'$

  Subtermenii lui l_1 care nu sunt variabile:

  $$(x * y),\ (y * v),\ (x * y) * (y * v)\ .$$

Exemplu

Exemplu

- Determinăm perechile critice pentru
  $l_1 := (x * y) * (y * v),\ r_1 := y,\ l_2 := (x' * y') * (y' * v'),\ r_2 := y'$
  Subtermenii lui l_1 care nu sunt variabile:
  $$(x * y),\ (y * v),\ (x * y) * (y * v)\ .$$
  - $t := x * y,\ c = z * (y * v),\ \theta := \{x \leftarrow x' * y', y \leftarrow y' * v'\}$
    $\theta(r_1) = y' * v',\ \theta(c)[z \leftarrow \theta(r_2)] = y' * ((y' * v') * v)$
    Perechea critică: $(y' * v', y' * ((y' * v') * v))$.

Exemplu

Exemplu

- Determinăm perechile critice pentru

  $l_1 := (x * y) * (y * v),\ r_1 := y,\ l_2 := (x' * y') * (y' * v'),\ r_2 := y'$

  Subtermenii lui l_1 care nu sunt variabile:

  $$(x * y),\ (y * v),\ (x * y) * (y * v)\ .$$

  - $t := x * y,\ c = z * (y * v),\ \theta := \{x \leftarrow x' * y', y \leftarrow y' * v'\}$
    $\theta(r_1) = y' * v',\ \theta(c)[z \leftarrow \theta(r_2)] = y' * ((y' * v') * v)$
    Perechea critică: $(y' * v', y' * ((y' * v') * v))$.
  - $t := y * v,\ c = (x * y) * z,\ \theta := \{y \leftarrow x' * y', v \leftarrow y' * v'\}$
    $\theta(r_1) = x' * y',\ \theta(c)[z \leftarrow \theta(r_2)] = (x * (x' * y')) * y'$
    Perechea critică: $(x' * y', (x * (x' * y')) * y')$.

Exemplu

Exemplu

- Determinăm perechile critice pentru

  $l_1 := (x * y) * (y * v),\ r_1 := y,\ l_2 := (x' * y') * (y' * v'),\ r_2 := y'$

  Subtermenii lui l_1 care nu sunt variabile:

  $$(x * y),\ (y * v),\ (x * y) * (y * v)\ .$$

  - $t := x * y,\ c = z * (y * v),\ \theta := \{x \leftarrow x' * y', y \leftarrow y' * v'\}$
    $\theta(r_1) = y' * v',\ \theta(c)[z \leftarrow \theta(r_2)] = y' * ((y' * v') * v)$
    Perechea critică: $(y' * v', y' * ((y' * v') * v))$.
  - $t := y * v,\ c = (x * y) * z,\ \theta := \{y \leftarrow x' * y', v \leftarrow y' * v'\}$
    $\theta(r_1) = x' * y',\ \theta(c)[z \leftarrow \theta(r_2)] = (x * (x' * y')) * y'$
    Perechea critică: $(x' * y', (x * (x' * y')) * y')$.
  - $t := (x * y) * (y * v),\ c = z,\ \theta := \{x \leftarrow x', y \leftarrow y', v \leftarrow v'\}$
    $\theta(r_1) = y',\ \theta(c)[z \leftarrow \theta(r_2)] = y'$
    Perechea critică: (y', y').

Exemplu

Exemplu

- Perechile critice:
  1. $(y' * v', y' * ((y' * v') * v))$,
  2. $(x' * y', (x * (x' * y')) * y')$,
  3. (y', y').

Exemplu

Exemplu

- Perechile critice:
  1. $(y' * v', y' * ((y' * v') * v))$,
  2. $(x' * y', (x * (x' * y')) * y')$,
  3. (y', y').
- Avem
  - $y * ((y * x) * v) > y * x$
  - $(x * (v * y)) * y > v * y$

Exemplu

Exemplu

- Perechile critice:
  1. $(y' * v', y' * ((y' * v') * v))$,
  2. $(x' * y', (x * (x' * y')) * y')$,
  3. (y', y').
- Avem
  - $y * ((y * x) * v) > y * x$
  - $(x * (v * y)) * y > v * y$
- Considerăm
  $T := T \cup \{y * ((y * x) * v) \to y * x,\ (x * (v * y)) * y \to v * y\}$
- T este complet și este completarea lui R_E.

Programare logică ecuațională

Ce am studiat până acum

- (S, Σ) signatură multisortată și Γ mulțime de ecuații condiționate
- G o mulțime de ecuații de forma $(\forall X)t \doteq_s t'$, $t, t' \in T_\Sigma(X)$.

- În cursurile anterioare am răspuns la problema

$$\Gamma \models (\forall X)G.$$

  - $\mathcal{A} \models (\forall X)t \doteq_s t'$ if H:
    pt. or. morfism $h : T_\Sigma(X) \to \mathcal{A}$,
    $h_{s'}(u) = h_{s'}(v)$, or. $u \doteq_{s'} v \in H \Rightarrow h_s(t) = h_s(t')$
  - $\mathcal{A} \models \Gamma$:
    $\mathcal{A} \models (\forall X)t \doteq_s t'$ if H, or. $(\forall X)t \doteq_s t'$ if $H \in \Gamma$
  - $\Gamma \models (\forall X)t \doteq_s t'$:
    or. $\mathcal{A}$ a.î. $\mathcal{A} \models \Gamma$, $\mathcal{A} \models (\forall X)t \doteq_s t'$
  - $\Gamma \models (\forall X)G$:
    or. $\mathcal{A}$ a.î. $\mathcal{A} \models \Gamma$, $\mathcal{A} \models (\forall X)t \doteq_s t'$, or. $(\forall X)t \doteq_s t' \in G$

Problema programării logice ecuaționale

- (S, Σ) signatură multisortată și Γ mulțime de ecuații condiționate
- G o mulțime de ecuații de forma $(\forall X)t \doteq_s t'$, $t, t' \in T_\Sigma(X)$.

- Problema programării logice ecuaționale:

$$\Gamma \models (\exists X)G.$$

- $\Gamma \models (\exists X)G$:
  or. $\mathcal{A}$ a.î. $\mathcal{A} \models \Gamma$, $\mathcal{A} \models (\exists X)G$.

- $\mathcal{A} \models (\exists X)G$:
  există un morfim $h : T_\Sigma(X) \to \mathcal{A}$ a.î. $h_s(t) = h_s(t')$, or. $(\forall X)t \doteq_s t' \in G$.

Teoremele lui Herbrand

- Fundamentale pentru demonstrarea automată.
- Reduce problema satisfacerii în toate modelele, doar la satisfacerea în modelul inițial.

Teorema

Fie G o mulțime de ecuații de forma $(\forall X)t \doteq_s t'$, $t, t' \in T_\Sigma(X)$. Sunt echivalente:

1. $\Gamma \models (\exists X)G$,
2. $T_{\Sigma,\Gamma} \models (\exists X)G$,
3. *există un morfism* $\psi : T_\Sigma(X) \to T_\Sigma$ *a.î.* $\Gamma \models (\forall \emptyset)\psi(G)$.

Teoremele lui Herbrand

Demonstrație (*)

$1 \Rightarrow 2$: $\Gamma \models (\exists X)G \Rightarrow T_{\Sigma,\Gamma} \models (\exists X)G$

- Știm $\Gamma \models (\exists X)G$: or. $\mathcal{A}$ a.î. $\mathcal{A} \models \Gamma$, $\mathcal{A} \models (\exists X)G$.
- Dar $T_{\Sigma,\Gamma}$ este Γ-algebră inițială, deci $T_{\Sigma,\Gamma} \models \Gamma$.
- În concluzie, $T_{\Sigma,\Gamma} \models (\exists X)G$.

Demonstrație (*) (cont.)

$2 \Rightarrow 3$: $T_{\Sigma,\Gamma} \models (\exists X)G \Rightarrow$ ex. $\psi : T_\Sigma(X) \to T_\Sigma$ a.î. $\Gamma \models (\forall\emptyset)\psi(G)$

- Știm $T_{\Sigma,\Gamma} \models (\exists X)G$: ex. $h : T_\Sigma(X) \to T_{\Sigma,\Gamma}$ a.î. $h_s(t) = h_s(t')$, or. $(\forall X)t \doteq_s t' \in G$.
- $\eta : T_\Sigma \to T_{\Sigma,\Gamma} := T_\Sigma/_{\equiv_{\Gamma,T_\Sigma}}$ morfism surjectiv.
- Obținem că există $\psi : T_\Sigma(X) \to T_\Sigma$ a.î. $\psi;\eta = h$.

$$\begin{tikzcd} T_\Sigma \arrow[r, "\eta"] & T_{\Sigma,\Gamma} \\ & T_\Sigma(X) \arrow[u, "h"'] \arrow[ul, dotted, "\psi"] \end{tikzcd}$$

- Deci $\eta_s(\psi_s(t)) = \eta_s(\psi_s(t'))$, or. $(\forall X)t \doteq_s t' \in G$.

Demonstrație (*) (cont.)

$2 \Rightarrow 3$: $T_{\Sigma,\Gamma} \models (\exists X)G \Rightarrow$ ex. $\psi : T_\Sigma(X) \to T_\Sigma$ a.î. $\Gamma \models (\forall\emptyset)\psi(G)$

- Cum $\eta : T_\Sigma \to T_{\Sigma,\Gamma}$ este morfismul de factorizare, obținem $\psi_s(t) \equiv_{\Gamma,T_\Sigma} \psi_s(t')$, or. $(\forall X)t \doteq_s t' \in G$.
  - Dar $\equiv_{\Gamma,T_\Sigma} := \bigcap\{Ker(g) \mid g : T_\Sigma \to \mathcal{B},\ \mathcal{B} \models \Gamma\}$
  - Deci $\psi_s(t) \equiv_{\Gamma,T_\Sigma} \psi_s(t')$ înseamnă $g_s(\psi_s(t)) = g(\psi_s(t'))$, or. $g : T_\Sigma \to \mathcal{B} \models \Gamma$.
  - Trebuia să arătăm $\Gamma \models (\forall\emptyset)\psi(G)$: or. $\mathcal{B} \models \Gamma$, or. $g : T_\Sigma \to \mathcal{B}$, $g_s(\psi_s(t)) = g(\psi_s(t'))$, or. $(\forall X)t \doteq_s t' \in G$.

Teoremele lui Herbrand

Demonstrație (*) (cont.)

$3 \Rightarrow 1$: ex. $\psi : T_\Sigma(X) \to T_\Sigma$ a.î. $\Gamma \models (\forall\emptyset)\psi(G) \Rightarrow \Gamma \models (\exists X)G$

- Fie $\mathcal{M}$ o Γ-algebră. Arătăm că $\mathcal{M} \models (\exists X)G$.
  - există $h : T_\Sigma(X) \to \mathcal{M}$ a.î. $h_s(t) = h_s(t')$, or. $(\forall X)t \doteq_s t' \in G$.
- Fie $\alpha_\mathcal{M} : T_\Sigma \to \mathcal{M}$ unicul morfism de la T_Σ la $\mathcal{M}$.
- Arătăm că pentru $\psi; \alpha_\mathcal{M} : T_\Sigma(X) \to \mathcal{M}$,
  $(\psi; \alpha_\mathcal{M})_s(t) = (\psi; \alpha_\mathcal{M})_s(t')$, or. $(\forall X)t \doteq_s t' \in G$.
  - Deoarece $\mathcal{M} \models \Gamma$, din ipoteză obținem $\mathcal{M} \models (\forall\emptyset)\psi(G)$.
    - pt. or. $g : T_\Sigma \to \mathcal{M}$, $g_s(\psi_s(t)) = g_s(\psi_s(t'))$, or. $(\forall X)t \doteq_s t' \in G$.
  - Pentru morfism $\alpha_\mathcal{M} : T_\Sigma \to \mathcal{M}$ obțim
    $(\alpha_\mathcal{M})_s(\psi_s(t)) = (\alpha_\mathcal{M})_s(\psi_s(t'))$, or. $(\forall X)t \doteq_s t' \in G$.
- Deci $\mathcal{M} \models (\exists X)G$, or. $\mathcal{M}$ o Γ-algebră. În concluzie, $\Gamma \models (\exists X)G$.

$\square$

Soluție

- (S, Σ) signatură multisortată și Γ mulțime de ecuații condiționate
- G o mulțime de ecuații de forma $(\forall X)t \doteq_s t'$, $t, t' \in T_\Sigma(X)$.

Soluție

- (S, Σ) signatură multisortată și Γ mulțime de ecuații condiționate
- G o mulțime de ecuații de forma $(\forall X)t \doteq_s t'$, $t, t' \in T_\Sigma(X)$.

Definitie

Un morfism $f : T_\Sigma(X) \to \mathcal{A}$ este soluție pentru $(\exists X)G$ dacă

$$f(G) \subseteq \equiv_{\Gamma,\mathcal{A}}$$

$$\equiv_{\Gamma,\mathcal{A}} := \bigcap\{Ker(h) \mid h : \mathcal{A} \to \mathcal{B},\ \mathcal{B} \models \Gamma\}.$$

Soluție

- (S, Σ) signatură multisortată și Γ mulțime de ecuații condiționate
- G o mulțime de ecuații de forma $(\forall X)t \doteq_s t'$, $t, t' \in T_\Sigma(X)$.

Definitie

Un morfism $f : T_\Sigma(X) \to \mathcal{A}$ este soluție pentru $(\exists X)G$ dacă

$$f(G) \subseteq \equiv_{\Gamma,\mathcal{A}}$$

$$\equiv_{\Gamma,\mathcal{A}} := \bigcap \{Ker(h) \mid h : \mathcal{A} \to \mathcal{B},\ \mathcal{B} \models \Gamma\}.$$

- Δ este o mulțime de egalități adevărate din $T_\Sigma(X)$.
- Compunerea a două soluții este tot o soluție.

Context extins

- (S, Σ) signatură și X mulțime de variabile
- Un termen $c \in T_\Sigma(X \cup \{z\})_s$ se numește context dacă $nr_z(c) = 1$.
- Pentru un context $c \in T_\Sigma(X \cup \{z\})$, notăm
$$c[z \leftarrow t_0] := \{z \leftarrow t_0\}(c).$$

Context extins

- (S, Σ) signatură și X mulțime de variabile
- Un termen $c \in T_\Sigma(X \cup \{z\})_s$ se numește context dacă $nr_z(c) = 1$.
- Pentru un context $c \in T_\Sigma(X \cup \{z\})$, notăm
$$c[z \leftarrow t_0] := \{z \leftarrow t_0\}(c).$$
- Un context extins este o ecuație de forma
$$c \doteq_s t \text{ sau } t \doteq_s c$$
unde $c \in T_\Sigma(X \cup \{z\})_s$ și $t \in T_\Sigma(X)_s$.
- Notăm un context extins cu C.
- Observăm că $(c \doteq_s t)[z \leftarrow t_0]$ înseamnă $c[z \leftarrow t_0] \doteq_s t$.
- Notăm $C[z \leftarrow t_0]$ cu $C[t_0]$.

Reguli de deducție

Regula Morfismului $\boxed{\dfrac{G}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$

Reguli de deducție

Regula Morfismului $\boxed{\dfrac{G}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$

Regula Reflexiei extinse $\boxed{\dfrac{G \cup \{l \doteq_s r\}}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$ a.î. $\theta_s(l) = \theta_s(r)$

Reguli de deducție

Regula Morfismului $\boxed{\dfrac{G}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$

Regula Reflexiei extinse $\boxed{\dfrac{G \cup \{l \doteq_s r\}}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$ a.î. $\theta_s(l) = \theta_s(r)$

Regula Reflexiei $\boxed{\dfrac{G \cup \{l \doteq_s r\}}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$ a.î. $\theta = cgu(l, r)$

Reguli de deducție

Regula Morfismului $\boxed{\dfrac{G}{\theta(G)}}$

G mulțime de ecuații,
$\theta : T_\Sigma(X) \to T_\Sigma(Y)$

Regula Reflexiei extinse $\boxed{\dfrac{G \cup \{l \doteq_s r\}}{\theta(G)}}$

G mulțime de ecuații,
$\theta : T_\Sigma(X) \to T_\Sigma(Y)$ a.î.
$\theta_s(l) = \theta_s(r)$

Regula Reflexiei $\boxed{\dfrac{G \cup \{l \doteq_s r\}}{\theta(G)}}$

G mulțime de ecuații,
$\theta : T_\Sigma(X) \to T_\Sigma(Y)$ a.î.
$\theta = cgu(l, r)$

Regula Pararescrierii $\boxed{\dfrac{G \cup \{C[\theta_s(l)]\}}{G \cup \theta(H) \cup \{C[\theta_s(r)]\}}}$

G mulțime de ecuații,
$(\forall Y) l \doteq_s r$ if $H \in \Gamma$,
$\theta : T_\Sigma(Y) \to T_\Sigma(X)$
C context extins

Reguli de deducție

Regula Paramodulației extinse

$$\boxed{\frac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$$

G mulțime de ecuații,
$(\forall Y) l \doteq_s r$ if $H \in \Gamma$,
$X \cap Y = \emptyset$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta_s(l) = \theta_s(a)$, $a \in T_\Sigma(X)_s$
C context extins

Reguli de deducție

Regula Paramodulației extinse

$$\boxed{\frac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$$

G mulțime de ecuații,
$(\forall Y) l \doteq_s r$ if $H \in \Gamma$,
$X \cap Y = \emptyset$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta_s(l) = \theta_s(a)$, $a \in T_\Sigma(X)_s$
C context extins

Regula Paramodulației

$$\boxed{\frac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$$

G mulțime de ecuații,
$(\forall Y) l \doteq_s r$ if $H \in \Gamma$,
$X \cap Y = \emptyset$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta = cgu(l, a)$, $a \in T_\Sigma(X)_s$
C context extins

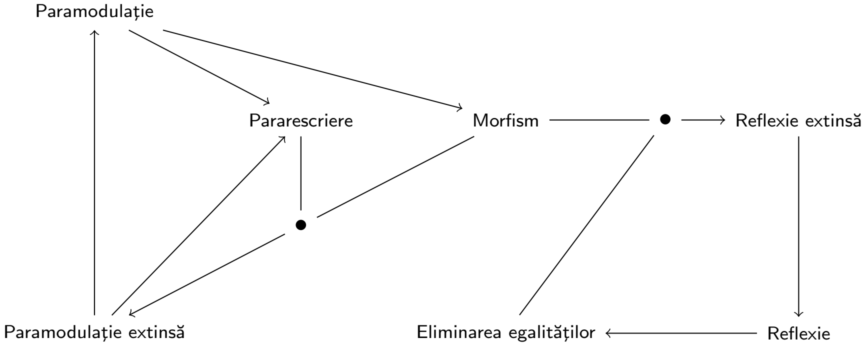
Regula narrowing

Regula Narrowing $\boxed{\dfrac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$

G mulțime de ecuații,
$(\forall Y) l \doteq_s r$ if $H \in \Gamma$,
l nu este variabilă,
$X \cap Y = \emptyset$,
$a \in T_\Sigma(X)_s$, $a \notin X$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta = cgu(l, a)$,
C context extins

- Caz particular de Paramodulație.

Legături între regulile de deducție



Exemplu

Exemplu

- $S = \{nat, nlist, list\}$
- $\Sigma = \{0 :\to nat, s : nat \to nat, nil :\to list,$
  $\_,\_ : list\ list \to list, \_,\_ : nlist\ nlist \to nlist,$
  $head : nlist \to nat, cdr : nlist \to list, \# : list \to nat\}$
- $\Gamma = \{(\forall\{E, L\})head(E, L) \doteq E,$
  $(\forall\{E, L\})cdr(E, L) \doteq L,$
  $(\forall\emptyset)\#(nil) \doteq 0,$
  $(\forall\{E, L\})\#(E, L) \doteq s(\#(L))\}$

Căutăm o soluție pentru problema:

$$(\exists L)\{\#(L) \doteq s(s(0)), head(L) \doteq 0\}.$$

Exemplu

Exemplu

- $\{\#(L) \doteq s(s(0)), head(L) \doteq 0\}$

Regula Paramodulației

$$\boxed{\frac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$$

G mulțime de ecuații,
$(\forall Y)l \doteq_s r$ if $H \in \Gamma$,
$X \cap Y = \emptyset$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta = cgu(l, a)$, $a \in T_\Sigma(X)_s$
C context extins

  - $(\forall\{E1, L1\})\#(E1, L1) \doteq s(\#(L1)) \in \Gamma$
  - $C : \bullet = s(s(0))$
  - $a : \#(L)$
  - θ cgu pt $\#(L)$ și $\#(E1, L1)$: $\theta(L) = E1, L1$

- $\{s(\#(L1)) \doteq s(s(0)), head(E1, L1) \doteq 0\}$ cu morfismul $h_1 : T_\Sigma(\{L\}) \to T_\Sigma(\{E1, L1\})$, $h(L) = E1, L1$

Exemplu

Exemplu

- $\{s(\#(L1)) \doteq s(s(0)), head(E1, L1) \doteq 0\}$

Regula Pararescrierii $\boxed{\dfrac{G \cup \{C[\theta_s(l)]\}}{G \cup \theta(H) \cup \{C[\theta_s(r)]\}}}$ G mulțime de ecuații, $(\forall Y) l \doteq_s r$ if $H \in \Gamma$, $\theta : T_\Sigma(Y) \to T_\Sigma(X)$ C context extins

  - $\forall\{E, L\}) head(E, L) \doteq E \in \Gamma$
  - $C : \bullet = 0$
  - $\theta : T_\Sigma(\{E, L\}) \to T_\Sigma(\{E1, L1\})$, $\theta(E) = E1$ și $\theta(L) = L1$

- $\{s(\#(L1)) \doteq s(s(0)), E1 \doteq 0\}$ cu morfismul $h_2 : T_\Sigma(\{E1, L1\}) \to T_\Sigma(\{E1, L1\})$ identitatea

Exemplu

Exemplu

- $\{s(\#(L1)) \doteq s(s(0)), E1 \doteq 0\}$

Regula Reflexiei $\boxed{\dfrac{G \cup \{l \doteq_s r\}}{\theta(G)}}$ G mulțime de ecuații, $\theta : T_\Sigma(X) \to T_\Sigma(Y)$ a.î. $\theta = cgu(l, r)$

  - $\theta : T_\Sigma(\{E1, L1\}) \to T_\Sigma(\{L1\})$, $\theta(E1) = 0$ este cgu pt $E1$ și 0

- $\{s(\#(L1)) \doteq s(s(0))\}$ cu morfismul $h_3 : T_\Sigma(\{E1, L1\}) \to T_\Sigma(\{L1\})$, $h_3(E1) = 0$

Exemplu

Exemplu

- $\{s(\#(L1)) \doteq s(s(0))\}$

Regula Paramodulației

$$\boxed{\frac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$$

G mulțime de ecuații,
$(\forall Y)l \doteq_s r$ if $H \in \Gamma$,
$X \cap Y = \emptyset$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta = cgu(l, a)$, $a \in T_\Sigma(X)_s$
C context extins

  - $(\forall\{E, L\})\#(E, L) \doteq s(\#(L)) \in \Gamma$
  - $C : s(\bullet) = s(s(0))$
  - $a : \#(L1)$
  - $\theta(L1) = E, L$ este cgu pt $\#(E, L)$ si $\#(L1)$

- $\{s(s(\#(L))) \doteq s(s(0))\}$ cu morfismul $h_4 : T_\Sigma(\{L1\}) \to T_\Sigma(\{E, L\})$, $h_4(L1) = E, L$

Exemplu

Exemplu

- $\{s(s(\#(L))) \doteq s(s(0))\}$

Regula Paramodulației $\boxed{\dfrac{G \cup \{C[a]\}}{\theta(G \cup H \cup \{C[r]\})}}$

G mulțime de ecuații,
$(\forall Y)l \doteq_s r$ if $H \in \Gamma$,
$X \cap Y = \emptyset$,
$\theta : T_\Sigma(X \cup Y) \to T_\Sigma(Z)$ a.î.
$\theta = cgu(l, a)$, $a \in T_\Sigma(X)_s$
C context extins

  - $(\forall\emptyset)\#(nil) \doteq 0 \in \Gamma$
  - $C : s(s(\bullet)) = s(s(0))$
  - $a : \#(L)$
  - $\theta(L) = nil$ este cgu pt $\#(nil)$ și $\#(L)$

- $\{s(s(0)) \doteq s(s(0))\}$ cu morfismul $h_5 : T_\Sigma(\{E, L\}) \to T_\Sigma(\{E\})$, $h_5(L) = nil$

Exemplu

Exemplu

- Un morfism $f : T_\Sigma(X) \to \mathcal{A}$ este soluție pentru $(\exists X)G$ dacă

$$f(G) \subseteq \equiv_{\Gamma,\mathcal{A}}$$

- Soluția cautată este: $h_1; h_2; h_3; h_4; h_5 : T_\Sigma(\{L\}) \to T_\Sigma(\{E\})$
  - $h_1 : T_\Sigma(\{L\}) \to T_\Sigma(\{E1, L1\})$, $h(L) = E1, L1$
  - $h_2 : T_\Sigma(\{E1, L1\}) \to T_\Sigma(\{E1, L1\})$
  - $h_3 : T_\Sigma(\{E1, L1\}) \to T_\Sigma(\{L1\})$, $h_3(E1) = 0$
  - $h_4 : T_\Sigma(\{L1\}) \to T_\Sigma(\{E, L\})$, $h_4(L1) = E, L$
  - $h_5 : T_\Sigma(\{E, L\}) \to T_\Sigma(\{E\})$, $h_5(L) = nil$

$$(h_1; h_2; h_3; h_4; h_5)(\#(L)) = (h_1; h_2; h_3; h_4; h_5)(s(s(0)))$$
$$(h_1; h_2; h_3; h_4; h_5)(head(L)) = (h_1; h_2; h_3; h_4; h_5)(0)$$

Concluzii

Teorema

În cadrul ecuațional, rezoluția se poate obține din narrowing și eliminarea egalităților adevărate.

Rezoluție = Narrowing = Paramodulație

Recapitulare

Recapitulare

- Programare Logică - cazul logicii clauzelor definite propoziționale
  - Logica propozițională
  - Sistem de deducție pentru clauze definite

Recapitulare

- Programare Logică - cazul logicii clauzelor definite propoziționale
  - Logica propozițională
  - Sistem de deducție pentru clauze definite
- Programare Logică - cazul logicii Horn
  - Logica de ordinul I (calculul cu predicate)
  - Algoritmul de unificare
  - Sistem de deducție backchain pentru logica Horn (clauze definite)
  - Rezoluție SLD

Recapitulare

- Algebre multisortate
  - Signaturi multisortate. Mulțimi și funcții multisortate
  - Algebre multisortate
  - Morfisme de algebre multisortate
  - Izomorfisme de algebre multisortate
  - Tipuri abstracte de date
  - Termeni. Algebre de termeni
  - Algebre inițiale
  - Algebre libere
  - Congruențe
  - Ecuații. Relația de satisfacere
  - Γ-algebre
  - Specificații algebrice

Recapitulare

- Logica ecuațională
  - Deducție ecuațională - cazul necondiționat
  - Deducție ecuațională - cazul condiționat
  - Corectitudinea logicii ecuaționale
  - Completitudinea logicii ecuaționale

Recapitulare

- Rescrierea termenilor
  - Contexte
  - Sistem de rescriere
  - Logica ecuațională și rescrierea termenilor
  - Sisteme de rescriere abstracte. Diverse proprietăți
  - Terminarea sistemelor de rescriere
  - Confluență și perechi critice
  - Algoritmul Knuth-Bendix

Recapitulare

Tipuri de exerciții la seminar:

1. Algoritmul de unificare
2. Deducții ecuaționale (cazul necondiționat)
3. Specificații algebrice
4. Rezoluție SLD
5. Sisteme de rescriere. Confluență

Baftă la examen!