Eigenschaften von $\mathcal{P}$ (4)

Satz 3.12.

Sei $\mathcal{A}$ ein Vervollständigungs-Verfahren.
$\mathcal{A}$ breche mit Eingabe $(E, \succ)$ nicht ab.

a) Gilt $u =_E v$ so existiert ein $i \in \mathbb{N}$ mit $u \downarrow v$ in R_i.
Es gilt auch $u \downarrow v$ in R^∞.

b) Es gilt $=_E = \overset{*}{\longleftrightarrow}_{R^\infty}$ und R^∞ ist konvergent.

Beweis-Idee:

a) Es gibt nach Lemma 3.17 eine Kette $B_0 \succeq_\mathcal{P} B_1 \succeq_\mathcal{P} \cdots$ mit B_j minimaler Beweis zu $u = v$ in (E_j, R_j).
Sei B_i minimal unter den B_j. Dies gibt es, da $\succ_\mathcal{P}$ Noethersch ist.
Dann ist nach Lemma 3.18 B_i ein V-Beweis in (E_i, R_i) und in (E^∞, R^∞).

b) $=_E = \overset{*}{\longleftrightarrow}_{R^\infty}$: Es gilt
„$\supseteq$“ wegen $R^\infty \subseteq =_E$
„$\subseteq$“ wegen a)
Termination: Es gilt $R^\infty \subseteq \succ$
Konfluenz: Wegen a) und $=_E = \overset{*}{\longleftrightarrow}_R$ hat $\longrightarrow$ die CR-Eigenschaft. Also ist $\longrightarrow$ konfluent.

3.6.3 Ein effizienter KBV-Algorithmus

Ziel

Algorithmus $\mathcal{A}$ mit

- $\mathcal{A}$ benutzt nur Inferenzregeln aus $\mathcal{P}$.
- $\mathcal{A}$ erzeugt nur faire Ableitungen $(E_i, R_i)_{i \in \mathbb{N}}$

$$E^\infty = \varnothing, \qquad CP(R^\infty) \subseteq \bigcup_{i \geq 0} E_i$$

- Eingabe: $(E, \succ)$ $\qquad (E_0, R_0) = (E, \varnothing)$

Struktur von $\mathcal{A}$: Makro-Schritte:

- R ist stets voll simplifiziert.
  E ist Menge der „wartenden“ Gleichungen.
- Makro-Schritt:
  wähle Gleichung $u = v$ in E und verarbeite sie.
- Stopp, falls E leer.
- Nichtdeterminismus:
  a) Wahl von $u = v$ wichtig!
  b) Reduktionsstrategie fest gewählt.

Makro-Schritt

- **Auswahl von $u = v$ in E (Heuristik)**
  Sei ψ : Term$(F, V) \times$ Term$(F, V) \to \mathbb{N}$ fair.
  Wähle $u = v$ mit minimalem ψ-Wert.

- **Verarbeitung von $u = v$**
  1. Ist $u = v$ noch aktuell?
     Streichen, falls ein Elternteil nicht mehr in R.
  2. Simplifizieren $u = v \rightsquigarrow \hat{u} = \hat{v}$.
  3. Löschen, falls $\hat{u} \equiv \hat{v}$.
  4. Gleichung richtbar?
     Zurückstellen, falls $\hat{u}, \hat{v}$ $\succ$-unvergleichbar, sonst Gleichung richten: $\hat{u} \to \hat{v}$, falls $\hat{u} \succ \hat{v}$.
  5. Einfügen in R $\quad \hat{u} \to \hat{v}$
     a) Bilde kritische Paare R mit $\hat{u} \to \hat{v} : s = t$.
        Simplifiziere $s = t$ zu $\hat{s} = \hat{t}$.
        Lösche, falls $\hat{s} \equiv \hat{t}$.
        Füge $\hat{s} = \hat{t}$ in E ein.
     b) Interreduziere $R \cup \{\hat{u} \to \hat{v}\}$
        (Regeln (P5) und (P6) in $\mathcal{P}$)

- **Abbruch**, falls $E \neq \varnothing$, kein $u = v$ richtbar.

KBV-Algorithmus

Algorithmus

Eingabe: $(E, \succ)$, $\quad \psi$ Heuristik

Ausgabe: R

1. Setze $R := \varnothing$
2. **<u>begin</u>**
   <u>while</u> $E \neq \varnothing$ **<u>do</u>**
   $\qquad \text{Makro-Schritt}(\psi)$
   <u>od</u>
   <u>end</u>

Mögliche Wahl von ψ

a) $\psi(u = v) = |u| + |v|$

b) $\psi(u = v) = \max(|u|, |v|)$

c) wie a), b) aber
   $\psi_0(u)$ statt $|u|$ mit
   $\psi_0 : F \to \mathbb{N}_+ \qquad \psi_0(x) = \mu$
   ψ_0 : Term$(F, V) \to \mathbb{N}_+$
   $\qquad \psi_0(f(t_1, \ldots, t_n)) = \psi_0(f) + \psi_0(t_1) + \cdots + \psi_0(t_n)$

Beispiel

$E: fff(x) = f(x) \qquad ggg(x) = h(x)$
$ fff(x) = g(x) \qquad hhh(x) = x$

$>_F$: KBO mit $h >_F g >_F f \quad \varphi(f) = \varphi(g) = \varphi(h) = 1$

E	R
$\underline{f^3(x) = f(x)}$ $\quad g^3(x) = h(x)$ $f^3(x) = g(x) \quad h^3(x) = x$	–
$\underline{f^3(x) = g(x)}$ $g^3(x) = h(x)$ $h^3(x) = x$	$f^3(x) \to f(x)$
$\underline{g^3(x) = h(x)}$ $h^3(x) = x$	$f^3(x) \to f(x)$ $g(x) \to f(x)$
$\underline{h^3(x) = x}$	$f^3(x) \to f(x)$ $g(x) \to f(x)$ $h(x) \to f(x)$
–	$g(x) \to x$ $h(x) \to x$ $f(x) \to x$

$R: f(x) \to x, \quad g(x) \to x, \quad h(x) \to x.$

Beispiel: Nilpotentes Monoid

Eingabe: $E: x \cdot 1 = x \qquad (x \cdot y) \cdot z = x \cdot (y \cdot z)$
$ 1 \cdot x = x \qquad x \cdot x = 1$

Frage: Gilt $x \cdot y =_E y \cdot x$?

E	R
$\underline{x \cdot 1 = x} \quad \underline{x \cdot x = 1}$ $\underline{1 \cdot x = x} \quad (x \cdot y) \cdot z = x \cdot (y \cdot z)$	–
$\underline{(x \cdot y) \cdot z = x \cdot (y \cdot z)}$	(1) $x \cdot 1 \to x$ (2) $1 \cdot x \to x$ (3) $x \cdot x \to 1$
$\underline{x \cdot (x \cdot z) = z}$ $\quad (3,4)$ $(x \cdot (y \cdot (x \cdot y))) = 1$ $\quad (3,4)$	(1) bis (3) (4) $(x \cdot y) \cdot z \to x \cdot (y \cdot z)$
$\underline{x \cdot (y \cdot (x \cdot y)) = 1}$ $x \cdot (y \cdot (x \cdot (y \cdot z))) = z$ $\quad (4,5)$	(1) bis (4) (5) $x \cdot (x \cdot y) \to y$
$x \cdot (y \cdot (x \cdot (y \cdot z))) = z$ $\underline{y \cdot (x \cdot y) = x}$ $\quad (5,6)$ ...	(1) bis (5) (6) $x \cdot (y \cdot (x \cdot y)) \to 1$
$x \cdot (y \cdot (x \cdot (y \cdot z))) = z$ $x \cdot y = y \cdot x$ $\quad (5,7)$ ...	(1) bis (5) (7) $y \cdot (x \cdot y) \to x$

Es gilt

$x \cdot y =_E y \cdot x$

Beispiel: Nilpotentes Monoid (2)

Kritische Paare zu

(4) $(x \cdot y) \cdot z \to x \cdot (y \cdot z)$ (5) $u \cdot (u \cdot v) \to v$

$\rightsquigarrow$ Unifikationsprobleme

$$
\begin{array}{llll}
(\alpha) & (x \cdot y) \cdot z =^? u \cdot (u \cdot v) & \rightsquigarrow u =^? x \cdot y & z =^? (x \cdot y) \cdot v \\
(\beta) & x \cdot y =^? u \cdot (u \cdot v) & \rightsquigarrow x =^? u & y =^? u \cdot v \\
(\gamma) & (x \cdot y) \cdot z =^? u \cdot v & \rightsquigarrow u =^? x \cdot y & v =^? z
\end{array}
$$

$\rightsquigarrow$ Überlappung

(α)

$$
\begin{tikzcd}
 & (x \cdot y) \cdot [(x \cdot y) \cdot v] \arrow[dl, "(4)"'] \arrow[dr, "(5)"] & \\
x \cdot (y \cdot [(x \cdot y) \cdot v]) & & v
\end{tikzcd}
$$

neue Gleichung
(wie (γ))

(β)

$$
\begin{tikzcd}
 & (u \cdot (u \cdot v)) \cdot z \arrow[dl, "(4)"'] \arrow[dr, "(5)"] & \\
u \cdot [(u \cdot v) \cdot z] \arrow[rr, "*"] & & v \cdot z
\end{tikzcd}
$$

zusammenführbar

(γ)

$$
\begin{tikzcd}
 & (x \cdot y) \cdot [(x \cdot y) \cdot z] \arrow[dl, "(5)"'] \arrow[dr, "(4)"] & \\
z & & (x \cdot y) \cdot (x \cdot (y \cdot z)) \arrow[d] \\
 & & x \cdot (y \cdot (x \cdot (y \cdot z)))
\end{tikzcd}
$$

$z = x \cdot (y \cdot (x \cdot (y \cdot z)))$ neue Gleichung

Theorembeweisen

Sei E Gleichungssystem über sig $= (S, F, \tau)$,
$u, v \in$ Term(F, V)

Definition 3.28 Skolemisierung

$u = v \rightsquigarrow \overline{u} = \overline{v}$: Ersetzung der Variablen x_i durch neue Konstanten $\overline{x_i}$ (also $\overline{x_i} \notin F$)

Satz 3.13.

Die KB-Vervollständigung mit

Eingabe: E, $\succ$, ψ, $u = v$

breche nicht ab, sie erzeuge $(E_i, R_i)_{i \geq 0}$.

Sei $\longrightarrow_i = \longrightarrow_{R_i}$. Es gilt:

$$u =_E v \qquad \text{gdw} \qquad \exists i \in \mathbb{N}: \quad \overline{u} \downarrow_i \overline{v}.$$

ohne Beweis

Bemerkung

Man kann den KBV-Algorithmus erweitern zu einem UKBV-Algorithmus (unfailing completion), der den Abbruch bei der Vervollständigung immer vermeidet. Dies liefert einen Semi-Entscheidungsalgorithmus für die E-Gleichheit: Gilt $u =_E v$, so sind $\overline{u}$ und $\overline{v}$ ab einem Zeitpunkt i in (E_i, R_i) zusammenführbar.

Waldmeister-Beweis

(siehe http://www-avenhaus.informatik.uni-kl.de/waldmeister/)

```
NAME          BoolRing

SORTS         S

SIGNATURE     +: S S -> S
              *: S S -> S
              -: S -> S
              0: -> S
              1: -> S
              a: -> S
              b: -> S
              c: -> S
              not: S -> S
              and: S S -> S
              or: S S -> S

ORDERING      LPO
              not > and > or > - > * > + > 1 > 0 > c > b > a

EQUATIONS     +(+(x,y),z) = +(x,+(y,z))
              +(x,y) = +(y,x)
              +(x,0) = x
              +(x,x) = 0
              +(x,-(x)) = 0
              *(x,0) = 0
              *(x,1) = x
              *(x,x) = x
              *(x,+(y,z)) = +(*(x,y),*(x,z))
              *(*(x,y),z) = *(x,*(y,z))
              *(x,y) = *(y,x)
              not(x) = +(1,-(x))
              and(x,y) = *(x,y)
              or(x,y) = +(+(*(x,y),x),y)
```


Waldmeister-Beweis (Forts.)

```
CONCLUSION    -(a) = a                                      % ( 6)
              *(a,a) = a                                    % ( 1)
              not(and(a,b)) = or(not(a),not(b))             % ( 8)
              not(or(a,b)) = and(not(a),not(b))             % ( 9)
              and(or(a,b),c) = or(and(a,c),and(b,c))        % (12)
              or(and(a,b),c) = and(or(a,c),or(b,c))         % (11)
              and(and(a,b),c) = and(a,and(b,c))             % (10)
              or(or(a,b),c) = or(a,or(b,c))                 % (17)
              and(a,b) = and(b,a)                           % (15)
              or(a,b) = or(b,a)                             % (19)
              and(a,or(a,b))= a                             % (14)
              or(a,and(a,b)) = a                            % (18)
              not(not(a)) = a                               % ( 7)
              not(1) = 0                                    % ( 2)
              not(0) = 1                                    % ( 5)
              and(a,0) = 0                                  % ( 3)
              and(a,1) = a                                  % (13)
              and(a,not(a)) = 0                             % ( 4)
              or(not(or(a,not(b))),not(or(not(a),not(b)))) = b  % (16)
```

Waldmeister beweist dies in 0,32 Sekunden

und erzeugt dabei

- 394 kritische Paare,
- 26 Regeln,
- 6 unrichtbare Gleichungen.

3.7 Reduktionsordnungen

Ziel: Konstruktion von Reduktionsordnungen $\succ$.

dann: $R \subseteq \succ \quad \rightsquigarrow \quad R$ terminierend.

2 Ansätze

a) Semantische Termordnungen
- $(\mathcal{A}, \succ_0)$ sig-Algebra und Noethersche Partialordnung.
- φ interpretiert die $f \in F$ als monotone Funktionen $\varphi(f)$ auf $\mathcal{A}$.
- $s \succ t \text{ gdw } \varphi(s) \succ_0 \varphi(t)$ auf $\mathcal{A}$.

b) Syntaktische Termordnungen
- direkt auf der syntaktischen Struktur der Terme definiert.

Voraussetzung ab jetzt

- Es gibt nur eine Sorte in sig.
- Es gibt $f \in F$ mit variabler Stelligkeit, z. B. $+(t_1, \ldots, t_n)$ statt $(\cdots(t_1 + t_2) + \cdots) + t_n$.
- F ist endlich.

3.7.1 Simplifikationsordnungen

Idee für Konstruktion von $\succeq$

- „Noethersch" ist i. Allg. schwer nachweisbar,
- suche hinreichendes syntaktisches Kriterium.

$\Rightarrow$ Simplifikationsordnungen.

Ansatz: Vermeidung offensichtlicher Nichttermination.

Sei $\succeq$ Ordnung auf Term(F, V).

Definition 3.29

R **erlaubt $\succeq$-Schleifen** gdw

$$\exists s, t : s \xrightarrow{+}_R t \wedge t \succeq s$$

Beispiel

$\succeq \, = \, \geq_{TT} \quad t \geq_{TT} s \text{ gdw } \exists p : t/p \equiv s.$

Also $s \xrightarrow{+}_R t \equiv t[s] \rightsquigarrow R$ nicht terminierend.

Lemma 3.19

R erlaubt $\geq_{TT}$-Schleifen $\rightsquigarrow R$ nicht terminierend.

Motivation: Konstruktion von Reduktionsordnungen

Frage:

Lässt sich Lemma 3.19 umkehren?

1. Versuch

Erlaubt R keine $\geq_{TT}$-Schleifen, so ist R terminierend.

$\Rightarrow$ Falsch!

2. Versuch

$\succeq = \trianglerighteq =$homöomorphe Einbettung.

Erlaubt R keine $\trianglerighteq$-Schleifen, so ist R terminierend.

$\Rightarrow$ Richtig! Es gilt $\geq_{TT} \subseteq \trianglerighteq$.

Ziele:

1. Definition von $\trianglerighteq$ auf $\mathsf{Term}(F, V)$ mit:
   Gilt $R \subseteq \trianglerighteq$, so ist R terminierend.
2. Verschärfung dieses Ansatzes.

Homöomorphe Einbettung $\trianglerighteq$

Definition 3.30

$s \equiv x \trianglelefteq t \text{ gdw } x \in \mathsf{Var}(t)$.

$s \equiv f(s_1, \ldots, s_n) \trianglelefteq t \equiv g(t_1, \ldots, t_m) \text{ gdw}$

$f = g$, es gibt $1 \leq j_1 < j_2 < \cdots < j_n \leq m$ mit $s_i \trianglelefteq t_{j_i}$ (α)

oder

es gibt $1 \leq j \leq m$: $s \trianglelefteq t_j$ (β)

Beachte

f habe feste Stelligkeit. Dann lautet (α).

$f = g \wedge s_i \trianglelefteq t_i$ für alle $1 \leq i \leq n$.

Beispiel

$f(a + b) \trianglelefteq f(a + f(b + a))$

denn $a + b \trianglelefteq a + f(b + a)$,

denn $a \trianglelefteq a$ und $b \trianglelefteq f(b + a)$,

denn $b <_{TT} f(b + a)$.

Homöomorphe Einbettung $\trianglerighteq$ (Forts.)

Lemma 3.20

a) Gilt $s \geq_{TT} t$, so ist $s \trianglerighteq t$.

b) $\trianglerighteq$ ist Noethersche Partialordnung.

Beweis:

a) Induktion nach $|s|$.

b) $\triangleright$ ist Partialordnung (klar).
Gilt $s \triangleright t$, so ist $|s| > |t|$, also ist $\trianglerighteq$ Noethersch.
(Induktion nach $|s|$.)

Regelsystem für $\trianglerighteq$

R bestehe aus folgenden Regeln

$f(x_1, \ldots, x_n) \to x_i \qquad f \in F \qquad 1 \leq i \leq n$

$f(x_1, \ldots, x_i, \ldots, x_n) \to f(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$,
falls f variable Stelligkeit hat.

Beispiel

$t \equiv f(a+f(b+a)) \longrightarrow_R f(a+f(b)) \longrightarrow_R f(a+b) \longrightarrow_R a+b$,
also $t \xrightarrow{*}_R a+b$.

Lemma 3.21

Es gilt $s \trianglerighteq t$ gdw $s \xrightarrow{*}_R t$.

Also ist $\trianglerighteq$ eine Reduktionsordnung.

Es gilt $\geq_{TT} \subseteq \trianglerighteq$.

Simplifikationsordnung

Definition 3.31

Eine Partialordnung $\succ$ auf Term(F, V) heißt **Simplifikationsordnung**, falls

a) $s \succ t \ \rightsquigarrow f(\cdots, s, \cdots) \succ f(\cdots, t, \cdots)$
b) $f(\cdots, t, \cdots) \succ t$
c) $f(\cdots, t_{i-1}, t_i, t_{i+1}, \cdots) \succ f(\cdots, t_{i-1}, t_{i+1}, \cdots)$

Bezeichnungen

a) Verträglichkeit mit Termstruktur.
b) Teilterm-Eigenschaft.
c) Lösch-Eigenschaft (nur sinnvoll, falls f variable Stelligkeit hat).

Lemma 3.22

Sei $\succ$ Simplifikations-Ordnung.

Dann gilt $\rhd \subseteq \succ$.

Beweis: Induktion nach $|s|$ liefert

$$s \trianglerighteq t \quad \rightsquigarrow s \succeq t$$

Das Baum-Theorem

Voraussetzung F, V_0 sind endlich.

Satz 3.14. (Kruskal, Higman)

Zu jeder unendlichen Folge $(t_i)_{i \in \mathbb{N}}, t_i \in$ Term(F, V_0) gibt es $j < k$ mit $t_j \trianglelefteq t_k$.

ohne Beweis.

Satz 3.15.

Es gebe kein $s, t \in$ Term(F, V_0) mit $s \xrightarrow{+}_R t$ und $s \trianglelefteq t$. Dann ist R terminierend.

Beweis: Dies folgt aus Satz 3.14.

Satz 3.16.

Seien $\succ, \succ_1$ Partialordnungen auf Term(F, V_0) mit $\succ \subseteq \succ_1$ und $\rhd \subseteq \succ_1$. Dann ist $\succ$ Noethersch.

Beweis: Sonst sei $t_0 \succ t_1 \succ t_2 \succ \cdots$ eine unendliche Kette.

$$\text{Dann gilt} \quad \left.\begin{array}{ll} t_j \succ_1 t_k & \text{für alle } j, k \text{ mit } j < k \\ t_j \trianglelefteq t_k & \text{für ein } j, k \text{ mit } j < k \\ \rightsquigarrow t_j \preceq_1 t_k & \text{für ein } j, k \text{ mit } j < k \end{array}\right\} \lightning$$

Ergebnisse

Satz 3.17.

Jede Simplifikationsordnung auf Term(F, V_0) ist Noethersch.

Beweis: Wähle $\succ_1 = \succ$ in Satz 3.16. Dies ist möglich wegen Lemma 3.22.

Satz 3.18.

Sei $\succ$ eine Simplifikationsordnung auf Term(F, V), die mit Substitutionen verträglich ist. Dann ist $\succ$ eine Reduktionsordnung.

Beweis: Dies folgt aus Satz 3.17.

Beispiel

R: $x - 0 \to x$
$s(x) - s(y) \to x - y$
$0 - y \to 0$

Es gilt $R \subseteq \trianglerighteq$, also ist R terminierend.

Übung

a) Jede KBO ist eine Simplifikationsordnung (also Noethersch!).
b) $\trianglerighteq$ ist eine Reduktionsordnung.

3.7.2 Semantische Termordnungen

Definition 3.32

Sei $(M, \succ_0)$ eine Menge mit Partialordnung.

$\alpha : M^n \to M$ heißt **stark monoton**, falls

- $x \succ_0 y \leadsto \alpha(\cdots, x, \cdots) \succ_0 \alpha(\cdots, y, \cdots)$.
- $\alpha(\cdots, x_{i-1}, x_i, x_{i+1}, \cdots) \succ_0 \alpha(\cdots, x_{i-1}, x_{i+1}, \cdots)$, falls α variable Stelligkeit hat.
- $\alpha(\cdots, x, \cdots) \succ_0 x$.

Sei $(M, \succ_0)$ wie oben,

φ eine Interpretation von F auf $(M, \succ_0)$ mit
$\varphi(f)$ stark monotone Funktion.

Setze φ : Term$(F) \to M$

$$\varphi(f(t_1, \ldots, t_n)) = \varphi(f)(\varphi(t_1), \ldots, \varphi(t_n))$$
$$s \succ t \text{ gdw } \varphi(s) \succ_0 \varphi(t)$$

Dann ist $\succ$ eine **semantische Termordnung 1. Art** auf Term(F).

Ergebnisse

Lemma 3.23

Jede semantische Termordnung 1. Art ist eine Simplifikationsordnung auf Term(F), also Noethersch.

Beachte

$\succ_0$ muss nicht Noethersch sein.

Beispiel: $(M, \succ_0) = (\mathbb{R}, >)$ ist erlaubt.

Semantische Termordnung 2. Art

Definition 3.33

Sei $(M, \succ_0)$ wie eben, $\succ_0$ Noethersch.

$\alpha : M^n \to M$ heißt **monoton**, falls

$$x \succ_0 y \rightsquigarrow \alpha(\cdots, x, \cdots) \succ_0 \alpha(\cdots, y, \cdots)$$

Sei φ Interpretation von F, die $\varphi(f)$ seien monoton.

Setze φ : Term$(F) \to M$ wie eben.

Dann heißt $\succ$

$$s \succ t \text{ gdw } \varphi(s) \succ_0 \varphi(t)$$

eine **semantische Termordnung 2. Art** auf Term(F).

Lemma 3.24

Sei $\succ$ semantische Termordnung 2. Art. Dann ist $\succ$ Noethersch und verträglich mit der Termstruktur.

Semantische Termordnungen + Termination

Lemma 3.25

Sei R Regelsystem über sig $= (S, F, \tau)$

$\succ$ semantische Termordnung 1. oder 2. Art auf Term(F).

Gilt $\sigma(l) \succ \sigma(r)$ für alle $l \to r$ in R, σ Grundsubstitution,

dann ist R terminierend.

Beweis:

Sei $s \longrightarrow_R t$, $s, t \in$ Term(F). Dann ist $s \succ t$, da $\succ$ verträglich mit der Termstruktur ist.

Es gibt also keine Folge $(t_i)_{i \in \mathbb{N}}$ mit $t_i \longrightarrow_R t_{i+1}$, $t_i \in$ Term(F).

Also ist R terminierend auf Term(F), also auch auf Term(F, V).

Definition 3.34

Eine semantische Termordnung heißt eine **Polynomordnung**, falls $\varphi(f)$ ein Polynom ist für alle $f \in F$.

Semantische Termordnung auf Term(F, V)

Sei

- $(M, \succ_0)$ wie oben
- $\succ_1$ die Ordnung auf Funktionen $\alpha : M^n \to M$
- $\alpha_1 \succ_1 \alpha_2$ gdw
  $\forall x_1, \ldots, x_n \in M : \alpha_1(x_1, \ldots, x_n) \succ_0 \alpha_2(x_1, \ldots, x_n)$

Definition 3.35

Sei φ eine Interpretation von F auf $(M, \succ_0)$ wie oben.

Setze φ : Term$(F, V) \to$ Menge der $\alpha : M^n \to M$ mit

$\varphi(x) = x$ für $x \in V$.

$\varphi(f(t_1, \ldots, t_n)) = \varphi(f)(\varphi(t_1), \ldots, \varphi(t_n))$

Sei $\succ$ auf Term(F, V):

$$s \succ t \text{ gdw } \varphi(s) \succ_1 \varphi(t)$$

Dann heißt φ eine **semantische Termordnung auf Term**(F, V) 1. bzw. 2. Art.

Ergebnisse

Satz 3.19.

Jede semantische Termordnung ist eine Reduktionsordnung auf $\mathsf{Term}(F, V)$.

Also:

- $\succ$ semantische Termordnung.
- $R \subseteq\ \succ$

$\rightsquigarrow$ R ist terminierend.

Polynomordnung: Beispiel 1

Sei

R: $\mathsf{not}(\mathsf{and}(x, y)) \to \mathsf{or}(\mathsf{not}(x), \mathsf{not}(y))$ (1)
$\mathsf{not}(\mathsf{or}(x, y)) \to \mathsf{and}(\mathsf{not}(x), \mathsf{not}(y))$ (2)

Wähle

- $M = \mathbb{N}_2 = \{i \in \mathbb{N} \mid i \geq 2\}$ $\succ$ auf $\mathbb{N}$
- $\varphi(\mathsf{not})(x) := x^2$
- $\varphi(\mathsf{or})(x, y) := x + y$ $\varphi(\mathsf{and})(x, y) := x + y$.

Es gilt

$l \succ r$ für alle $l \to r$ in R:

zu (1): Es gilt

$\varphi(l) = (x + y)^2 \succ_1 x^2 + y^2 = \varphi(r)$, also $l \succ r$.

zu (2): Auch hier gilt

$\varphi(l) = (x + y)^2 \succ_1 x^2 + y^2 = \varphi(r)$, also $l \succ r$.

Also:

R ist terminierend.

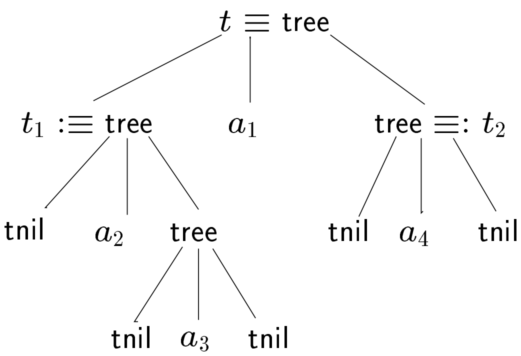
Beachte

Es gibt keine KBO $\succeq_{\mathsf{kbo}}$ mit $R \subseteq\ \succ_{\mathsf{kbo}}$.

Beispiel 2

Ziel Baum in Liste transformieren.

Beispiel:



$t \xrightarrow{*}_R a_2.a_3.a_1.a_4.$nil

Modellierung

Signatur: sig $= (S, F, \tau)$ mit

Sorten $S = \{$NAT, LIST, TREE$\}$

Operatoren F:

nil :	$\to$ LIST	tnil :	$\to$ TREE
. :	NAT, LIST $\to$ LIST	tree :	TREE, NAT, TREE $\to$ TREE
app :	LIST, LIST $\to$ LIST	flat :	TREE $\to$ LIST

Beispiel 2 (Forts.)

Sei

$R:$ (1) app(nil, l_2) $\to l_2$
(2) app($x.l_1, l_2$) $\to x.$app(l_1, l_2)
(3) flat(tnil) $\to$ nil
(4) flat(tree(t_1, x, t_2)) $\to$ app(flat(t_1), $x.$flat(t_2))

Beispiel wie oben:

flat(t) $\longrightarrow_R$ app(flat(t_1), $a_1.$flat(t_2))
$\xrightarrow{*}_R$ app($a_2.a_3.$nil, $a_1.a_4.$nil) $\xrightarrow{*}_R a_2.a_3.a_1.a_4.$nil

Wähle $M = \mathbb{N}_2$.

$\varphi(\text{nil}) = \varphi(\text{tnil}) = 2$ $\quad$ $\varphi(\text{flat})(t) = t^2$
$\varphi(.)(x, l) = x + l$ $\quad$ $\varphi(\text{app})(l_1, l_2) = 2l_1 + l_2$
$\varphi(\text{tree})(t_1, x, t_2) = 2t_1 + x + t_2$

Es gilt $l \succ r$ für alle $l \to r$ in R:

zu (2): $\varphi(l) = 2(x + l_1) + l_2 \succ_1 x + 2l_1 + l_2 = \varphi(r)$,
also $l \succ r$.

zu (4): $\varphi(l) = (2t_1 + x + t_2)^2 \succ_1 2t_1^2 + x + t_2^2 = \varphi(r)$,
also $l \succ r$.

zu (1) und (3): Übung.

Also: R ist terminierend.