# Term Rewriting
# Termersetzungssysteme

Vorlesung an der Fakultät für Mathematik und Informatik der Universität Leipzig [1]

Dr. Rolf Hartwig

seit Wintersemester 1992/93 mehrfach gehalten, zuletzt im WS 1997/98

[1]nach einer vom Autor korrigierten Mitschrift von stud. inf. Dirk Weigenand, letzte Korrektur im Februar 1998

Quelle: http://www.informatik.uni-leipzig.de/∼rhartwig/

Der Autor ist jederzeit für inhaltliche Hinweise zur Verbesserung der Vorlesung als auch für Hinweise zu immer noch verbliebenen Druckfehlern dankbar.

# Inhaltsverzeichnis

**1 Einführung** **1**

1.1 Grundbegriffe: Terme, Termalgebren, Adressen, Lücken . . . . . 1

1.2 Gleichungstheorie, Kongruenzrelationen und Hüllenoperatoren . . . . . 5

**2 Termersetzungssysteme** **13**

**3 Konfluenz und Termination** **17**

**4 Vervollständigung von Termersetzungssystemen** **25**

4.1 Theoretische Grundlagen . . . . . 25

4.2 Der Vervollständigungsalgorithmus . . . . . 34

**5 Termination von Termersetzungssystemen** **43**

5.1 Ordnungsrelationen . . . . . 43

5.2 Einbettungsrelation und Simplifikationsordnungen . . . . . 44

5.3 Eine allgemeine Methode für Terminationsbeweise . . . . . 52

5.4 Die rekursive Pfade–Ordnung . . . . . 55

**Index** **58**

# Kapitel 1

# Einführung

## 1.1 Grundbegriffe: Terme, Termalgebren, Adressen, Lücken

Zur Einführung bringen wir die wichtigsten Grundbegriffe einschließlich der hier gebrauchten Bezeichnungen. Der Student sollte dies zugleich zum Anlaß nehmen, wesentliche Teile des Stoffes der Vorlesung „Algebraische Grundlagen der Informatik“ zu wiederholen.

$\Sigma = (S, \Omega, \alpha)$ sei eine *Signatur*, wobei $S$ die Menge der *Sorten*, $\Omega$ die Menge der *Operatoren* und $\alpha : \Omega \mapsto S^* \times S$ die *Aritätsfunktion* bezeichnen.

Wir bilden im folgenden für $w \in S^*,\ s \in S$ die Menge

$$\Omega_{w,s} = \{\omega \mid \omega \in \Omega \wedge \alpha(\omega) = (w, s)\}$$

und fassen diese Menge von Operatoren gleicher Arität in der Familie

$$\Sigma = (\Sigma_{w,s})_{w \in S^*, s \in S}$$

zusammen und bezeichnen jetzt diese Familie $\Sigma$ als (S–sortige) *Signatur*.

**Beachte:** In der Regel sind die meisten $\Sigma_{w,s}$ leer.

$\omega \in \Sigma$ bedeutet im weiteren also: es existieren ein $w \in S^*$ und ein $s \in S$, mit $\omega \in \Sigma_{w,s}$.

Eine $\Sigma$–*Algebra* ist dann ein Paar

$$\mathcal{A} = ((A_s)_{s \in S}, (f_\sigma)_{\sigma \in \Sigma}),$$

wobei für $\sigma \in \Sigma_{w,s}$

$$f_\sigma : A^w \mapsto A_s,\ \textit{dasheißt}\ f_\sigma : A_{s_1} \times A_{s_2} \times \ldots \times A_{s_n} \mapsto A_s$$

bei $w = s_1 s_2 \ldots s_n$.

Familie der *variablenfreien* oder *Grundterme* $T_\Sigma = (T_{\Sigma,s})_{s \in S}$:

1. wenn $\sigma \in \Sigma_{\epsilon,s}$ („nullstelliger Operator“), so ist auch $\sigma \in T_{\Sigma,s}$ ($\sigma$ ist eine *Konstante*)
2. wenn $\sigma \in \Sigma_{w,s}$ mit $w = s_1 s_2 \ldots s_n$ und $t_i \in T_{\Sigma,s_i}$, $(i = 1, 2, \ldots n)$, so gilt auch $\sigma t_1 t_2 \ldots t_n \in T_{\Sigma,s}$.

Wir betrachten nun *Terme mit Variablen.*

Sei dazu gegeben eine Familie $X = (X_s)_{s \in S}$. Das ist dann eine $S$–sortige Familie von „Variablen“–mengen. Definiere nun Terme wie oben, aber mit dem Zusatz:

$$X_s \subseteq T_{\Sigma,s}$$

Wir bezeichnen diese Termmengenfamilie dann mit $T_\Sigma(X)$.

$\Sigma$–*Termalgebra* über dem Variablensystem $X$ („Standardtermalgebra“):

$$\underline{T_\Sigma(X)} = (T_\Sigma(X), (f_\sigma)_{\sigma \in \Sigma})$$

$$\text{mit } f: \ T_\Sigma(X)^w \mapsto T_\Sigma(X)_s$$

$$\text{vermöge } f_\sigma(t_1, t_2, \ldots, t_n) = \sigma t_1 \ldots t_n.$$

Über der Mengenfamilie $T_\Sigma(X)$ wird eingeführt:

Für $t \in T_\Sigma(X)$ sei

$$var(t) =_{df} \text{ die Menge der in } t \text{ vorkommenden Variablen .}$$

**Übungsaufgabe 1**

Geben Sie eine formale Definition für den inhaltlich klaren Begriff $var(t)$ an!

Ein $\Sigma$–*Homomorphismus* von $\mathcal{A}$ (wie oben) in eine $\Sigma$–Algebra $\mathcal{B} = ((B_s)_{s \in S}, (g_\sigma)_{\sigma \in \Sigma})$ ist eine Abbildungsfamilie $h = (h_s)_{s \in S}$, $h_s : A_s \mapsto B_s$, wobei für alle $\sigma \in \Sigma_{w,s}$ und alle $(a_1, a_2, \ldots, a_n) \in A^w$

$$h_s f_\sigma(a_1, \ldots, a_n) = g_\sigma(h_{s_1} a_1, \ldots, h_{s_n} a_n)$$

gilt, falls $w = s_1 s_2 \ldots s_n$.

$$\begin{tikzcd}
A^w \arrow[r, mapsto, "f_\sigma"] \arrow[d, "h^w"'] \arrow[dr, phantom, "="] & A_s \arrow[d, "h_s"] \\
B^w \arrow[r, mapsto, "g_\sigma"] & B_s
\end{tikzcd}$$

Sei $\mathcal{A}$ eine $\Sigma$–Algebra mit den Bezeichnungen wie oben. Dann ist eine $\mathcal{A}$–*Belegung* des Variablensystems $X$ eine Abbildung $\varphi : X \mapsto A$.

**Satz 1.1**

*Zu jeder $\Sigma$–Algebra $\mathcal{A}$ und jeder $\mathcal{A}$–Belegung $\varphi$ von $X$ gibt es genau einen Homomorphismus $\varphi^*$ von $\underline{T_\Sigma(X)}$ in $\mathcal{A}$, der $\varphi$ fortsetzt.*

$$\begin{tikzcd}
\underline{T_\Sigma(X)} \arrow[dr, "\varphi^*"] & \\
& \mathcal{A} \\
X \arrow[uu, "\subseteq"] \arrow[ur, "\varphi"'] &
\end{tikzcd}$$

das heißt:
$\underline{T_\Sigma(X)}$ ist frei über $X$ in $\underline{\underline{\mathrm{Alg}}}_\Sigma$

Im *Spezialfall* $\mathcal{A} = \underline{T_\Sigma(X)}$ wird definiert:

- $\varphi : \ X \mapsto T_\Sigma(X)$ als *Einsetzung* und
- $\varphi^* : \ T_\Sigma(X) \mapsto T_\Sigma(X)$ als *Substitution* (induziert von der Einsetzung $\varphi$)

**Bemerkung**

1. Manchmal wird statt $\underline{T_\Sigma(X)}$ wie oben, auch eine zu ihr isomorphe Algebra (freie Algebra, PEANO–Algebra, Ausdrucksalgebra) betrachtet und auch deren Elemente *Terme* genannt.

2. Ist $\varphi$ eine Einsetzung und $t$ ein Term, so schreibt man oft einfach $\varphi(t)$ statt $\varphi^*(t)$.

**Definition 1.1**

Eine Einsetzung $\rho : X \mapsto T_\Sigma$ (von Grundtermen für Variablen) heißt *Konkretisierung* (engl. „instantiation“). Für beliebige $t \in T_\Sigma(X)$ und eine gegebene Konkretisierung $\rho$ soll unter $\rho(t)$ wieder $\rho(t) =_{df} \rho^*(t) = sub(t, \rho)$ verstanden werden. $\rho(t)$ heißt dann ein (konkretes) *Beispiel* für $t$ (eine *„Instanz“* von $t$).

**Definition 1.2**

Sei $\underline{T_\Sigma(X)} = (T_\Sigma(X), (f_\sigma)_{\sigma\in\Sigma})$ eine Termalgebra (nicht notwendigerweise die Standardtermalgebra) und $t$ ein beliebiger ihrer Terme: $t \in T_\Sigma(X)_s$.
Die *Adresse* (oder *Stelle*) $a$ eines Teilterms $t'$ von $t$, deren *Sorte* $\mathrm{sort}(a)$ sowie der *Teilterm* $t|a$ werden induktiv wie folgt definiert:

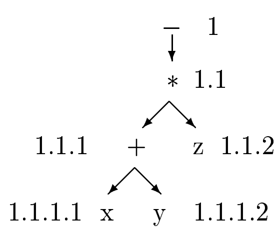
1. Ist $t' = t$, so erhält $t'$ die Adresse 1 und es wird gesetzt: $\mathrm{sort}(1) = s,\ t|1 = t$.

2. Ist $a$ die Adresse eines Teilterms $t'$ von $t$ und ist $t' = f_\sigma(t_1, \ldots, t_n)$ (d.h. $t_1, \ldots, t_n$ sind unmittelbare Teilterme von $t'$) mit $\sigma \in \Sigma_{w,s}$, $w = s_1 s_2 \ldots s_n$, so erhalten die Teilterme $t_1, \ldots, t_n$ von $t$ die Adressen $a.1, a.2, \ldots, a.n$ (in dieser Reihenfolge) und es wird gesetzt: $\mathrm{sort}(a.i) = s_i$ und $t|a.i = t_i$ für $i = 1, \ldots n$.

Die Menge der *Adressen* aller Teilterme von $t$ wird mit $\mathrm{adr}(t)$ bezeichnet.

**Beispiel 1**

$$\boxed{t = -(x + y) * z}$$




einfache Bestimmung der Adresse:
$t$ als Baum darstellen, Knoten „strukturell“ durchnumerieren, Wurzel =1.

$$\mathrm{adr}(t) = \{1, 1.1, 1.1.1, 1.1.2, 1.1.1.1, 1.1.1.2\}.$$

**Bemerkung**

1. Zu jeder Adresse $a \in \mathrm{adr}(t)$ gehört genau eine Sorte $s = \mathrm{sort}(a)$ und ein Teilterm $t|a$ der Sorte $s$, und umgekehrt gehört zu jedem *Vorkommen* eines Teilterms genau eine Adresse. [1]

2. In der Standardtermalgebra (Terme in Präfixschreibweise) genügt es, die Menge der *Positionen* $p$ im Term $(1 \leq p \leq l(t))$ als Adressen zu wählen. Das entspricht der lexikographischen Numerierung der oben definierten Adressen.

**Übungsaufgabe 2**

Formen Sie den im obigen Beispiel angeführten Term $t = -(x + y) * z$ in „Standardtermschreibweise" um und stellen Sie dann die beiden Möglichkeiten der Adressierung gegenüber!

**Definition 1.3 (Ersetzungsoperation)**

Für $t \in T_\Sigma(X),\ a \in \mathrm{adr}(t)$ mit $\mathrm{sort}(a) = s$ und $t' \in T_\Sigma(X)_s$ bezeichnet

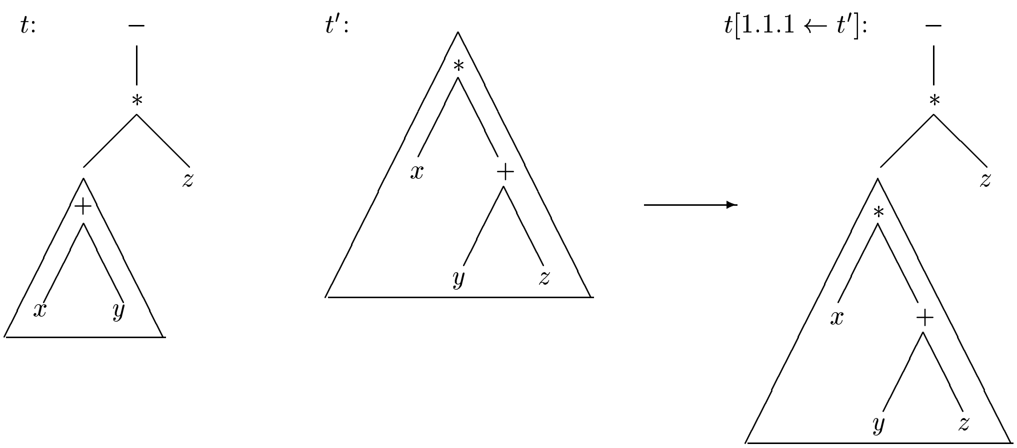
$$t[a \leftarrow\ t']$$

den Term, der entsteht, wenn man den Teilterm von $t$, der an der Stelle $a$ vorkommt, an dieser Stelle streicht und durch den Term $t'$ ersetzt.
[ Anmerkung: In der Regel ändert sich dadurch natürlich die Adressenmenge. ]

**Beispiel 2**

Für $t = -(x + y) * z$, $t' = x * (y + z),\ a = 1.1.1$ folgt dann

$$t[1.1.1 \leftarrow\ t'] \ = \ -((x * (y + z)) * z).$$




Zum Begriff des „Kontext":

Wir führen zu jeder Sorte $s$ eine neue, ausgezeichnete Variable $\boxed{s}$, $\boxed{s} \notin X_s$ (*Lücke*, *Platzhaltervariable*) ein

[1] Formal könnte man *Vorkommen* von $t'$ in $t$ als Paar $(t', a)$ mit $a \in \mathrm{adr}(t)$ definieren.

und bilden:

$$X_s^L = X_s \cup \{\boxed{s}\},\ L = (\{\boxed{s}\})_{s\in S}\ ,$$

also $X^L = X \cup L$. Nun bilden wir $T_\Sigma(X^L)$.

**Definition 1.4**

Jeder Term aus $T_\Sigma(X^L)$, der genau ein Vorkommen einer Platzhaltervariablen aus $L$ enthält, heißt *Kontext* und wird durch

$$C[\ ]$$

bezeichnet. Ist $C[\ ]$ ein Kontext aus $T_\Sigma(X_s^L)_{s'}$, der die Platzhaltervariable $\boxed{s}$ enthält, so heißt $C[\ ]$ ein *Kontext der Sorte $s'$ mit der Lückensorte $s$* und man schreibt auch $C[\ _s]$. $C[\ ] = \boxed{s}$ heißt *leerer Kontext* der Sorte $s$.

**Definition 1.5**

Ist $C[\ _s]$ ein Kontext mit der Lückensorte $s$ und $t \in T_\Sigma(X)_s$, so bezeichnet $C[t]$ den Term aus $T_\Sigma(X)$, den man durch Einsetzung von $t$ für $\boxed{s}$ erhält:

$$t' = C[t] \ \text{ genau dann, wenn } \ t' = sub(C[\ ], \varphi).$$

wobei $\varphi$ die Einsetzung ist mit $\varphi(\boxed{s}) = t$ und $\varphi(x) = x$ für alle $x \in X$ mit $x \neq \boxed{s}$.

## 1.2 Gleichungstheorie, Kongruenzrelationen und Hüllenoperatoren

Zunächst seien ganz knapp einige Begriffe zu Gleichungssystemen zusammengestellt bzw. wiederholt.

$\Sigma$–*Gleichungssystem* über dem Variablensystem $X = (X_s)_{s\in S}$:

$$E = (E_s)_{s\in S} \subseteq (T_\Sigma(X)_s \times T_\Sigma(X_s)_{s\in S})$$

$(t, t') \in E_s$ heißt *Gleichung* der Sorte $s$.

**Bemerkung**

Für die Untersuchung der Gültigkeit von Gleichungen mit Hilfe des üblichen Gleichungskalküls ist es zweckmäßig, nur *nichtleere* Trägermengen bei den beteiligten Algebren vorauszusetzen[2].

Eine $\Sigma$–Algebra $\mathcal{A}$ *erfüllt* eine Gleichung $(t, t')$ : $\mathcal{A} \models t = t'$ genau dann, wenn $\forall \varphi : X \mapsto A$ gilt: $\varphi^*(t) = \varphi^*(t')$.

$\mathcal{A}$ ist *Modell* für $E$ ($E$ *gültig* in $\mathcal{A}$) genau dann, wenn $\forall\ e \in E : \mathcal{A} \models e$.
$\mathcal{A}$ heißt dann $(\Sigma, E)$–*Algebra*.

*Varietät* der $(\Sigma, E)$–Algebren: $\underline{\underline{\text{Alg}}}_{\Sigma,E}$

**Beispiel 3**

$$x + 0 \quad = \quad x$$

[2] vgl. die Vorlesung „Algebraische Grundlagen der Informatik“

$$\begin{array}{rcl} x + (y+1) & = & (x+y)+1 \\ 0 * x & = & 0 \\ (x+1) * y & = & x * y + y \end{array}$$

**Definition 1.6**

Eine zweistellige Relation $R = (R_s)_{s \in S}$ heißt *kompatibel* genau dann, wenn für alle $\sigma \in \Sigma_{w,s}$ mit $w = s_1 s_2 \ldots s_n$ gilt:
wenn $(a_i, b_i) \in R_{s_i}$ $(1 \leq i \leq n)$, so ist auch

$$(f_\sigma(a_1 a_2 \ldots a_n), f_\sigma(b_1 b_2 \ldots b_n)) \in R_s.$$

$R$ ist $\Sigma$–*Kongruenzrelation* über der $\Sigma$–Algebra $\mathcal{A}$ genau dann, wenn $R$ eine *kompatible* Äquivalenzrelation ist.

**Definition 1.7**

Sei $R$ eine Relation über der $\Sigma$–Algebra $\mathcal{A}$. Dann bezeichnet $\mathrm{kom}(R)$ den *kompatiblen Abschluß* von $R$ und wird wie folgt definiert:

1. $R \subseteq \mathrm{kom}(R)$
2. wenn $\sigma \in \Sigma_{w,s}$ mit $w = s_1 s_2 \ldots s_n$ und $(a_i, b_i) \in \mathrm{kom}(R)_{s_i}$ für $1 \leq i \leq n$, so ist $(f_\sigma(a_1, a_2, \ldots, a_n), f_\sigma(b_1, b_2, \ldots, b_n)) \in (\mathrm{kom}(R))_s$
3. $\mathrm{kom}(R)$ ist die bezüglich $\subseteq$ kleinste Relation, für die 1. und 2. gilt.

**Folgerung 1.2**

*Der Abschlußoperator* kom *besitzt die Hülleneigenschaften:*

1. $R \subseteq \mathrm{kom}(R)$ *(Einbettung)*
2. $R \subseteq R' \rightarrow \mathrm{kom}(R) \subseteq \mathrm{kom}(R')$ *(Monotonie)*
3. $\mathrm{kom}(\mathrm{kom}(R)) = \mathrm{kom}(R)$ *(Abgeschlossenheit)*

**Satz 1.3**

*$R$ sei ein beliebige zweistellige Relation über $\mathcal{A}$. Dann ist*

$$\mathrm{con}(R) =_{df} \bigcap \{R' \mid R \subseteq R' \ \ \textit{und} \ \ R' \ \ \textit{ist Kongruenzrelation} \ \ \textit{in } \mathcal{A}\}$$

*die von $R$* erzeugte Kongruenz *über $\mathcal{A}$.*

Für eine Relation $R$ bezeichne $R^+$ wie üblich den transitiven Abschluß von $R$, $\ ^{-1}$ die Invertierung und $Id_A$ die Identitätsrelation über $A$. Dann gilt

**Satz 1.4**

$$\mathrm{con}(R) \;=\; (\mathrm{kom}(\; R \;\cup R^{-1} \;\cup Id_A \;))^+$$

Für den Beweis sind zunächst drei Lemmata zu beweisen (am günstigsten induktiv über die Erzeugung des entsprechenden Hüllenoperators).

**Lemma 1.5**

*Für beliebige zweistellige Relationen $Q$ über $\mathcal{A}$ gilt:*

1. *Wenn $Q$ symmetrisch, so auch* $\mathrm{kom}(Q)$ *symmetrisch.*
2. *Wenn $Q$ symmetrisch, so auch $Q^+$ symmetrisch.*
3. *Wenn $Q$ kompatibel, so auch $Q^n$ für beliebige natürliche $n$ und damit auch $Q^+$ kompatibel.*

**Übungsaufgabe 3**

Beweisen Sie die 1. Behauptung des obigen Lemmas 1.5.

**Übungsaufgabe 4**

Beweisen Sie die 2. Behauptung des obigen Lemmas 1.5.

**Übungsaufgabe 5**

Beweisen Sie die 3. Behauptung des obigen Lemmas 1.5.

**Beweis**

(des obigen Satzes):
Sei nun $\hat{R} = (\mathrm{kom}(R \cup R^{-1} \cup Id_A))^+$. Dann gilt:

1. $\hat{R}$ ist Kongruenzrelation über $\mathcal{A}$ mit $R \subseteq \hat{R}$. Die Reflexivität ergibt sich unter Anwendung der Hülleneigenschaften von kom und $+$ in trivialer Weise, die Transitivität von $\hat{R}$ direkt. Zum Beweis der Symmetrie und Kompatiblität sind obige Lemmata heranzuziehen. Also folgt wegen der Definition von $\mathrm{con}(R)$ als $\bigcap$: $\mathrm{con}(R) \subseteq \hat{R}$.

2. Umgekehrt ist $\mathrm{con}(R)$ eine Kongruenzrelation mit $R \subseteq \mathrm{con}(R)$. Da $\mathrm{con}(R)$ Kongruenzrelation ist, muß auch gelten:
$$R^{-1} \subseteq \mathrm{con}(R)$$
und
$$Id_A \subseteq \mathrm{con}(R)$$
(wegen Symmetrie und Reflexivität der Kongruenzrelation). Also
$$R \cup R^{-1} \cup Id_A \subseteq \mathrm{con}(R).$$

Wegen Definition von kom und Kompatibilität von $\text{con}(R)$ auch

$$\text{kom}(R \cup R^{-1} \cup Id_A) \subseteq \text{con}(R)$$

und wegen der Transitivität von $\text{con}(R)$ folgt daraus

$$(\text{kom}(R \cup R^{-1} \cup Id_A))^+ \subseteq \text{con}(R),$$

also $\hat{R} \subseteq \text{con}(R)$.

Aus 1. und 2. ergibt sich die Behauptung. q.e.d.

In Zukunft interessieren uns hauptsächlich Relationen über Termalgebren. Für diese sind zwei Hüllenoperatoren wichtig:

- der *stabile* Abschluß **S** und
- der *invariante* bzw. *kontextuelle* Abschluß inv bzw. **C**

**Definition 1.8**

Eine binäre Relation $R$ über $T_\Sigma(X)$ heißt *stabil* genau dann, wenn aus $(t_1, t_2) \in R$ für eine beliebige Einsetzung $s : X \mapsto T_\Sigma(X)$ stets auch $(s(t_1), s(t_2)) \in R)$ folgt.

$R = (R_s)_{s \in S}$ heißt *invariant* oder auch *monoton* genau dann, wenn aus $(t_1, t_2) \in R_s$ für beliebige Terme $t \in T_\Sigma(X)$, $a \in \text{adr}(t)$ mit $\text{sort}(a) = s$ stets auch $(t[a \leftarrow t_1], t[a \leftarrow t_2]) \in R$ folgt.

**Folgerung 1.6**

*$R$ ist invariant genau dann, wenn aus $(t_1, t_2) \in R_s$ für beliebige Kontexte $C[\ _s] \in T_\Sigma(X^L)$ stets auch $(C[t_1], C[t_2]) \in R$ folgt.*

**Bemerkung**

Eine invariante Kongruenzrelation heißt auch *vollinvariant*.

Entsprechend diesen Eigenschaften definiert man den *stabilen Abschluß* $\mathbf{S}(R)$ einer Relation $R$ [3] und den invarianten Abschluß:

**Definition 1.9**

$R$ sei eine binäre Relation über $T_\Sigma(X)$.

1. $\text{inv}_0(R) =_{df} R$
2. für natürliche Zahlen $n \geq 0$ gilt :
   $\text{inv}_{n+1}(R) =_{df} \text{inv}_n(R) \cup \{(C[t_1], C[t_2]) \mid C[\ _s]$ ist Kontext mit der Lückensorte $s$ und $(t_1, t_2) \in \text{inv}_n(R)_s\}$
3. $\text{inv}(R) =_{df} \bigcup_{n=0}^{\infty} \text{inv}_n(R)$

[3] vgl. dazu die Definition in der Vorlesung „Algebraische Grundlagen der Informatik“

$\mathrm{inv}(R)$ heißt *invarianter Abschluß* bzw. *kontextueller Abschluß* von $R$.

Zum Begriff der *syntaktischen Äquivalenz* $\equiv_E$ in $\underline{\underline{\mathrm{Alg}}}_{\Sigma,E}$ führen wir folgende Definition an:

**Definition 1.10**

$\equiv_E$ ist die von $\mathbf{S}(E)$ erzeugte Kongruenz über $T_\Sigma(X)$.

**Folgerung 1.7**

$$\equiv_E = \mathrm{con}(\mathbf{S}(E))$$

**Satz 1.8**

*Die syntaktische Äquivalenz* $\equiv_E$ *der Varietät* $\underline{\underline{\mathrm{Alg}}}_{\Sigma,E}$ *ergibt sich zu*

$$\equiv_E \, = (\mathrm{kom}(\mathbf{S}(E) \cup \mathbf{S}(E)^{-1} \cup Id_{T_\Sigma(X)}))^+.$$

**Bemerkung**

Die verschiedenen Abschlußoperationen können zwar nicht beliebig vertauscht, aber z.B. ineinander verschachtelt werden wie bei der üblichen Ableitungsrelation der Gleichungslogik:

**Definition 1.11**

$E$ sei ein $\Sigma$–Gleichungssystem über $X$. Dann wird für beliebige $t_1, t_2 \in T_\Sigma(X)$ und beliebige natürliche Zahlen $n$ definiert:

(0) $E \vdash_0 (t_1, t_2)$ genau dann, wenn $t_1 = t_2$ oder $(t_1, t_2) \in \mathbf{S}(E)$.

$E \vdash_{n+1} (t_1, t_2)$ genau dann, wenn einer der folgenden Fälle zutrifft:

(1) $E \vdash_n (t_1, t_2)$ (Stufenhebung)

(2) $E \vdash_n (t_2, t_1)$ (symmetrischer Abschluß)

(3) es existiert ein $t_3 \in T_\Sigma(X)$ mit $E \vdash_n (t_1, t_3)$ und $E \vdash_n (t_3, t_2)$ (transitiver Abschluß)

(4) es existiert ein $\sigma \in \Sigma_{w,s}$ mit $w = s_1 s_2 \ldots s_m$ und $t_{1j}, t_{2j} \in T_\Sigma(X)$ für $j = 1, \ldots, m$ und $E \vdash_n (t_{1j}, t_{2j})$ und es ist $t_1 = \sigma t_{11} t_{12} \ldots t_{1m}$ und $t_2 = \sigma t_{21} t_{22} \ldots t_{2m}$. (kompatibler Abschluß)

Man definiert $E \vdash (t_1, t_2)$ oder $E \vdash t_1 = t_2$ (aus $E$ *ableitbar*) genau dann, wenn ein $n$ existiert mit $E \vdash_n (t_1, t_2)$.

Dann gilt der folgende

**Satz 1.9**

*Für beliebige Terme* $t_1, t_2 \in T_\Sigma(X)$ *gilt*

$$t_1 \equiv_E t_2 \quad \textit{genau dann, wenn} \quad E \vdash t_1 = t_2.$$

Aus Zeitgründen führen wir hier keinen **Beweis**.

Bei der Konstruktion von Kongruenzen über Termalgebren ist die Bildung der kompatiblen Hülle und der kontextuellen Hülle unter gewissen Bedingungen äquivalent. Es gilt

**Lemma 1.10**

*Es sei* $R$ *eine reflexive Relation über* $T_\Sigma(X)$ *und* $C[\ _s]$ *ein Kontext mit Lückensorte* $s$. *Aus* $(t_1, t_2) \in \mathrm{kom}(R)_s$ *folgt dann* $(C[t_1], C[t_2]) \in \mathrm{kom}(R)$.

**Beweis**

(induktiv über den Aufbau des Kontextes $C[\ _s]$:)

(IA) $C[\ _s]$ ist Element des Erzeugendensystems von $T_\Sigma(X^L)$.
Da es ein Kontext sein soll, muß gelten: $C[\ _s] = \boxed{s}$. Also ist $C[t_i] = t_i$ für $i = 1, 2$. Damit folgt die Behauptung aus der Voraussetzung.

(IS) Sei $C[\ _s] = \sigma e_1 \ldots e_n$, $\sigma \in \Sigma_{w,s'}$, $e_i \in T_\Sigma(X^L)$, aber nur in genau einem $e_i$ genau ein Vorkommen von $\boxed{s}$, für dieses $e_j$ sei die Behauptung erfüllt (IV).

$$e_j = C'[\ _s].$$

$C[\ _s] = \sigma e_1 \ldots e_{i-1} C'[\ _s] e_{j+1} \ldots e_n$. Nach (IV) gilt $(C'[t_1], C'[t_2] \in \mathrm{kom}(R)_s$. Wegen der Reflexivität von $R$ gilt nun $(e_i, e_i) \in R \subseteq \mathrm{kom}(R)$ für alle $i \neq j$. Damit ist jetzt nach Definition von „kom“:

$$(\sigma e_1 \ldots e_{j-1} C'[t_1] e_{j+1} \ldots e_n\ ,\ \sigma e_1 \ldots e_{j-1} C'[t_2] e_{j+1} \ldots e_n) \in \mathrm{kom}(R),$$

also $(C[t_1], C[t_2]) \in \mathrm{kom}(R)$.

q.e.d.

**Lemma 1.11**

*Für reflexive Relationen* $R$ *über* $T_\Sigma(X)$ *gilt:*

$$\mathrm{inv}(R) \subseteq \mathrm{kom}(R).$$

**Beweis**

Es wird gezeigt:

$$\forall n \in I\!N :\ inv_n(R) \subseteq \mathrm{kom}(R)$$

(IA) $\mathrm{inv}_0(R) = R \subseteq \mathrm{kom}(R)$.

(IS) Sei $C[\ _s]$ ein Kontext und $(t_1, t_2) \in \mathrm{inv}_n(R)_s$, nach (IV) folgt dann $(t_1, t_2) \in \mathrm{kom}(R)_s$, woraus mit obigem Lemma $(C[t_1], C[t_2]) \in \mathrm{kom}(R)$ folgt.
Damit ist nach Definition von inv auch $\mathrm{inv}_{n+1}(R) \subseteq \mathrm{kom}(R)$.

q.e.d.

**Lemma 1.12**

(Auf dieses Lemma wird in einem später liegenden Abschnitt noch einmal Bezug genommen.)
*Für beliebige binäre Relationen $R$ über $T_\Sigma(X)$ gilt:*

$$\mathrm{kom}(R) \subseteq (\mathrm{inv}(R))^+.$$

**Beweis**

(induktiv über den Aufbau von kom)

(IA) $R \subseteq \mathrm{inv}(R) \subseteq (\mathrm{inv}(R))^+$

(IS) Sei $\sigma \in \Sigma_{w,s}$ mit $w = s_1 \dots s_n$, für $i = 1, \dots n$ sei weiter $(t_i, t_i') \in \mathrm{kom}(R)_{s_i}$, d.h. nach (IV) $(t_i, t_i') \in (\mathrm{inv}(R)_{s_i})^+$.
Betrachte nun für $i = 1, \dots n$ den Kontext

$$C_i[\ _{s_i}] = \sigma t_1' \dots t_{i-1}' \boxed{s_i} t_{i+1} \dots t_n.$$

Da auch $(\mathrm{inv}(R))^+$ invariant abgeschlossen ist (dies ist einfach nachzuweisen), folgt weiter $C_i[t_i], C_i[t_i']) \in (\mathrm{inv}(R)_{s_i})^+$ , also

$$(\sigma t_1' \dots t_{i-1}' t_i t_{i+1} \dots t_n\ ,\ \sigma t_1' \dots t_{i-1}' t_i' t_{i+1} \dots t_n) \in (\mathrm{inv}(R)_s)^+,$$

woraus wegen der Transitivität folgt

$$(\sigma t_1 \dots t_n\ ,\ \sigma t_1' \dots t_n') \in (\mathrm{inv}(R)_s)^+.$$

Damit ist aber

$$\mathrm{kom}(R) \subseteq (\mathrm{inv}(R))^+.$$

q.e.d.

Mit den beiden vorangehenden Lemmata ergibt sich die

**Folgerung 1.13**

$$(\mathrm{kom}(R))^* = (\mathrm{inv}(R))^*.$$

Damit erhalten wir eine weitere Charakterisierung der syntaktische Äquivalenz:

**Satz 1.14**

*Für die syntaktische Äquivalenz* $\equiv_E$ *der Varietät* $\underline{\underline{\mathrm{Alg}}}_{\Sigma,E}$ *gilt:*

$$\equiv_E = \ (\mathrm{inv}(\mathbf{S}(E)) \cup \mathrm{inv}(\mathbf{S}(E))^{-1})^*.$$

Für den **Beweis** ist der entsprechende Satz weiter oben für die syntaktische Äquivalenz, beschrieben durch kom, die Grundlage, wobei zu beachten ist, daß die Symmetrisierung und die Hinzunahme der Identität (reflexiver Abschluß) sich an beliebiger Stelle des Erzeugungsprozesses einfügen läßt.

# Kapitel 2

# Termersetzungssysteme

Durch ein Gleichungssystem $E$ wird wird eine Algebrenklasse $\underline{\underline{\mathrm{Alg}}}_{\Sigma,E}$ und festgelegt und gleichzeitig alle in ihr gültigen Gleichungen $\equiv_E$ . (Man beachte, daß nach dem Vollständigkeitssatz für die Gleichungstheorie die syntaktische Äquivalenz $\equiv_E$ mit der semantischen Äquivalenz $\stackrel{\triangle}{=}_E$ zusammenfällt.)
Die Idee der Auffassung eines Gleichungssystems $E$ als sogenanntes Termersetzungssystem besteht darin, die Gleichungen aus $E$ als Umformungsregeln (von links nach rechts) für Terme anzusehen.

**Definition 2.1**

Ein endliches[1] $\Sigma$–Gleichungssystem $R$ über dem Variablensystem $X$ heißt $\Sigma$–*Reduktionssystem* oder $\Sigma$–*Regelsystem* genau dann, wenn für jedes $(t_1, t_2) \in R$ gilt:

1. $t_1 \notin X$ (linke Seite ist keine Variable)
2. $\mathrm{var}(t_2) \subseteq \mathrm{var}(t_1)$ (rechts kommen keine neuen Variablen hinzu)

Ein Paar $(t_1, t_2) \in R$ heißt *Regel*, geschrieben:

$$t_1 \to t_2,$$

$t_1$ heißt *linke Seite* (LHS) [2], $t_2$ *rechte Seite* der Regel.

**Beispiel 4**

$$\begin{array}{rcll} x+0 & \to & x & \quad (r_1) \\ x+y' & \to & (x+y)' & \quad (r_2) \\ 0*x & \to & 0 & \quad (r_3) \\ x'*y & \to & x*y+y & \quad (r_4) \end{array}$$

**Definition 2.2**

Ein *Termersetzungssystem* (TRS)[3] ist ein Paar $(T, R)$ aus einer $\Sigma$–Termfamilie $T \subseteq T_\Sigma(X)$ und einem $\Sigma$–Reduktionssystem $R$.

[1] Beachte die (für eine allgemeine Theorie nicht notwendige, in der Praxis aber zweckmäßige) Beschränkung auf *endliche* Regelsysteme. Für manche unserer Sätze wird sich diese Einschränkung durchaus als vorteilhaft erweisen.
[2] engl.: left hand side
[3] engl.: term rewriting system

Man spricht von dem Termersetzungssystem $R$ selbst, wenn $(T_\Sigma(X), R)$ gemeint ist. Mit $R^0$ bezeichnet man das Termersetzungssystem $(T_\Sigma, R)$, das sich auf die Grundterme beschränkt.
Durch ein Termersetzungssystem $(T, R)$ werden Relationen festgelegt:
Ist $r : u \to v \in R$ eine Regel und $\varphi : X \mapsto T$ eine Einsetzung, so heißt

$$(\varphi(u), \varphi(v)), \qquad \text{auch } \varphi(u) \to \varphi(v) \text{ geschrieben,}$$

eine *Instanz der Regel* $r$. Die linke Seite $\varphi(u)$ heißt ein *Redex*, deren rechte Seite $\varphi(v)$ das zugehörige *Kontraktum.*
Die *1–Schritt–Reduktionsrelation*

$$\to_R \qquad \subseteq T \times T$$

ist wie folgt definiert:
$t_1 \to_R t_2$ genau dann, wenn eine Regel $r \in R$, Terme $t, u, v \in T$, eine Instanz $u \to v$ von $r$ und eine Adresse $a \in \mathrm{adr}(t)$ existieren, so daß

$$t_1 = t[a \leftarrow u] \qquad \text{und} \qquad t_2 = [a \leftarrow v]$$

gilt. Man schreibt dann auch:

$$t_1 \to_r t_2.$$

Eine möglicherweise unendliche Folge von Termen $t_0, t_1, t_2, \ldots$, für die

$$t_0 \to_R t_1 \to_R t_2 \to_R \ldots$$

gilt, heißt *Reduktionsfolge* oder *Ableitung* . Gilt

$$t_0 \to_R t_1 \to_R \ldots \to_R t_n$$

mit $n \geq 0$, so schreiben wir

$$t_0 \xrightarrow{n}_R t_n,$$

oder allgemein

$$t_0 \xrightarrow{*}_R t_n,$$

und $t_n$ heißt ein *Redukt* von $t_0$.

### Folgerung 2.1

1. *$\to_R$ ist der invariante, stabile Abschluß von $R$ :* $\quad \to_R = \mathrm{inv}(\mathbf{S}(R))$.
2. *$t_1 \to_r t_2$ gilt genau dann, wenn $u, v \in T_\Sigma(X)$, ein Kontext $C[\ ] \in T^L$, ein $\varphi : X \mapsto T$ mit $r = u \to v$ existieren und es ist $t_1 = C[\varphi(u)]$, $t_2 = C[\varphi(v)]$ .*
3. $\xrightarrow{*}_R \; = \; \mathrm{inv}(\mathbf{S}(R)))^*$.

### Beispiel 5

(Fortführung von Beispiel 4)
Sei $R$ das Regelsystem wie oben. Wir betrachten dazu das Termersetzungssystem $R^0$, das sich auf die Grundterme bezieht. Es gilt:

$$0'' * 0'' \xrightarrow{*}_R 0'''' \qquad (\hat{=}\, 2 * 2 = 4)$$

Denn wir haben folgende Reduktionsfolge:

$$\begin{aligned} 0'' * 0'' \quad &\to_{r_4} \quad 0' * 0'' + 0'' \to_{r_2} (0' * 0'' + 0')' \to_{r_2} (0' * 0'' + 0)'' \to_{r_1} (0' * 0'')'' \\ &\to_{r_4} \quad (0 * 0'' + 0'')'' \to_{r_3} (0 + 0'')'' \to_{r_2} (0 + 0')''' \to_{r_2} (0 + 0)'''' \to_{r_1} 0'''' \ . \end{aligned}$$

Ein gegebenes $\Sigma$–Reduktionssystem $R$ erzeugt die Relationen:

$\rightarrow_R$ – beschreibt „Rechnungen“ im Gleichungssystem $R$,

$\equiv_R$ – ist die von $R$ erzeugte Kongruenz (d. h. alle in $\underline{\underline{\mathrm{Alg}}}_{\Sigma,R}$ gültigen Gleichungen).

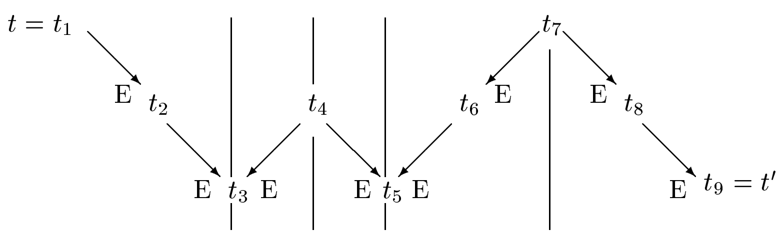
Nach Satz 1.14 ist
$$\equiv_R \; = (\mathrm{inv}(\mathbf{S}(R)) \cup \mathrm{inv}(\mathbf{S}(R))^{-1})^*,$$
woraus sich mit Folgerung 2.1 ergibt:
$$\equiv_R \; = (\rightarrow_R \cup \leftarrow_R)^*$$
oder anders geschrieben
$$\boxed{\equiv_R \; = \; \overset{*}{\underset{R}{\longleftrightarrow}}} \quad .$$

Mit $\overset{*}{\longleftrightarrow}$ bezeichnen wir in naheliegender Weise den reflexiv–transitiv–symmetrischen Abschluß von $\rightarrow_R$. Damit ergibt sich der Versuch, das sogenannte **Wortproblem**
$$t \equiv_E t' \;?$$
(das im allgemeinen unentscheidbar ist) durch Rechnungen in $E$ zu lösen:
$$t \overset{*}{\underset{E}{\longleftrightarrow}} t' \;?$$

Zum allgemein üblichen Sprachgebrauch in diesem Zusammenhang sei bemerkt, daß man vom *Wortproblem im engeren Sinne* spricht, wenn die interessierende Termmenge $T = T_\Sigma$ ist. Wenn man auch Terme mit Variablen einbezieht, also $T = T_\Sigma(X)$ betrachtet, so spricht man vom *Gültigkeitsproblem*.
Betrachtet man einen Beweis von
$$t \overset{*}{\underset{E}{\longleftrightarrow}} t',$$
so ist es i. a. notwendig, desöfteren die „Rechnungsrichtung“ zu wechseln, d. h. Gleichungen von $E$ als Termersetzungsregeln mal von links nach rechts, mal von rechts nach links anzusehen.

**Veranschaulichung:**

Beweis für $t \equiv_E t'$:




$t_1 \equiv_E t_3$ $\quad$ $t_3 \equiv_E t_4 \equiv_E t_5$ $\quad$ $t_5 \equiv_E t_7$ $\quad$ $t_7 \equiv_E t_9$

Wegen der Transitivität folgt dann $t_1 \equiv_E t_9$, was wiederum gleichbedeutend ist mit $t \equiv_E t'$.

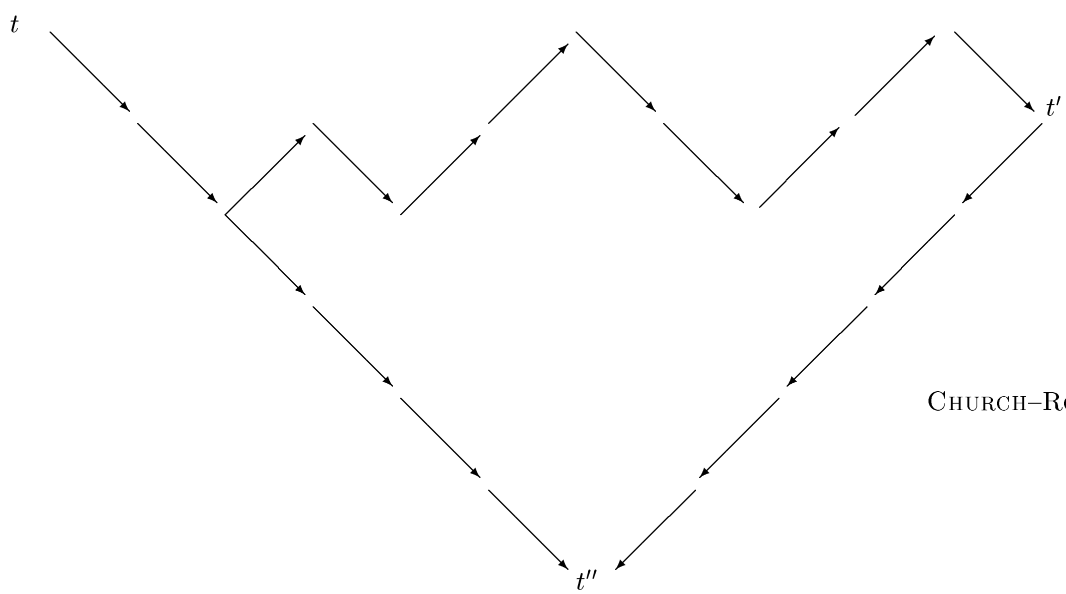
Hier sind also z. B. fünf „Vorwärtsrechnungen“ nötig, das ist sehr unschön, außerdem ist die Strategie für einen solchen „Beweis“im allgemeinen völlig unklar. Es ergibt sich die
**Frage:**
Ist es nicht möglich, sowohl von $t$, als auch von $t'$ aus „vorwärts“ zu rechnen bis zu einem gemeinsamen Ziel $t''$ ? Dies wäre eine klare Strategie. Leider ist dies aber nicht immer möglich. Termersetzungssysteme mit dieser Eigenschaft sind besonders ausgezeichnet. Damit beschäftigen wir uns im nächsten Kapitel.

# Kapitel 3

# Konfluenz und Termination

**Definition 3.1**
Ein Termersetzungssystem $(T, R)$ hat die CHURCH–ROSSER–*Eigenschaft* genau dann, wenn für alle $t, t' \in T$ stets gilt:

$$t \equiv_R t' \text{ genau dann, wenn ein } t'' \in T \text{ existiert mit } t \xrightarrow{*}_R t'' \text{ und } t' \xrightarrow{*}_R t''.$$




CHURCH–ROSSER–Eigenschaft

Ein unter Umständen leichter zu handhabendes Kriterium erweist sich als äquivalent.

**Definition 3.2**

Ein Termersetzungssystem $(T,R)$ heißt *konfluent* genau dann, wenn für alle $t, t_1, t_2 \in T$ gilt:
Wenn $t \xrightarrow{*}_R t_1$ und $t \xrightarrow{*}_R t_2$ gilt, dann existiert ein $t' \in T$ mit $t_1 \xrightarrow{*}_R t'$ und $t_2 \xrightarrow{*}_R t'$.

$$\begin{tikzcd}
 & t \arrow[dl] \arrow[dr] & \\
t_1 \arrow[dr] & & t_2 \arrow[dl] \\
 & t' &
\end{tikzcd}$$

Konfluenz

**Satz 3.1**

*Ein Termersetzungssystem besitzt genau dann die* CHURCH–ROSSER–*Eigenschaft, wenn es konfluent ist.*

**Beweis**

($\rightarrow$) Besitze $(T,R)$ die CHURCH–ROSSER–Eigenschaft. Betrachte beliebige $t, t_1, t_2 \in T$ mit $t \xrightarrow{*}_R t_1$ und $t \xrightarrow{*}_R t_2$. Dann ist $t_1 \equiv_R t_2$ und wegen der CHURCH–ROSSER–Eigenschaft existiert ein $t'$ mit $t_1 \xrightarrow{*}_R t'$ und $t_2 \xrightarrow{*}_R t'$. Also ist $(T,R)$ konfluent.

($\leftarrow$) Sei nun $(T,R)$ konfluent. Zum Nachweis der CHURCH–ROSSER–Eigenschaft ist nur eine Implikationsrichtung zu zeigen, die andere gilt trivialerweise.
Sei also $t \equiv_R t'$. Damit gilt $t \xleftrightarrow[R]{*} t'$. Nach Definition von $\xleftrightarrow[R]{*}$ existieren endlich viele „Zwischenglieder“ $t_i \xleftrightarrow[R]{} t_{i+1}$ (d.h., $t_i \rightarrow_R t_{i+1}$ oder $t_{i+1} \rightarrow_R t_i$). Faßt man diese nach gleicher „Richtung“ zusammen, so ergibt sich: es existiert ein $n$ und $t_1, t_2, \ldots, t_n \in T$ mit

$$t \xleftarrow[R]{*} t_1 \xrightarrow{*}_R t_2 \xleftarrow[R]{*} t_3 \ldots \xrightarrow{*}_R t_{n-1} \xleftarrow[R]{*} t_n \xrightarrow{*}_R t'$$

(beachte: $\xrightarrow{*}_R$ ist reflexiv, deshalb ist obige „Kette“ stets in dieser Weise darstellbar). Der Beweis erfolgt nun induktiv über $n$:

(IA) $n = 1$: $\quad t \xleftarrow[R]{*} t_1 \xrightarrow{*}_R t'$. Die Existenz eines $t''$ mit $t \xrightarrow{*}_R t'' \xleftarrow[R]{*} t'$ ergibt sich damit sofort aus der Konfluenz.

(IS) Betrachte $t_{n-1} \xleftarrow[R]{*} t_n \xrightarrow{*}_R t'$. Aus der Konfluenz folgt die Existenz von $t''' \in T$ mit $t_{n-1} \xrightarrow{*}_R t'''$ und $t' \xrightarrow{*}_R t'''$. Damit hat man

$$t \xleftarrow[R]{*} t_1 \xrightarrow{*}_R t_2 \xleftarrow[R]{*} t_3 \ldots t_{n-2} \xrightarrow{*}_R t_{n-1} \xrightarrow{*}_R t'''$$

beziehungsweise (wegen der Transitivität)

$$t \xleftarrow[R]{*} t_1 \xrightarrow{*}_R t_2 \dots t_{n-2} \xrightarrow{*}_R t''',$$

wobei wegen der geringeren Zahl der Zwischenglieder $(n-2)$ für die letzte Kette die (IV) anwendbar ist, d.h., es existiert ein $t'' \in T$ mit $t \xrightarrow{*}_R t''$ und $t''' \xrightarrow{*}_R t''$. Also gilt wegen $t' \xrightarrow{*}_R t'''$ auch $t' \xrightarrow{*}_R t''$, das heißt, wir haben die behauptete CHURCH–ROSSER–Eigenschaft.

q.e.d.

Leider gilt der

**Satz 3.2**

*Die Eigenschaft der Konfluenz ist für beliebige Termersetzungssysteme unentscheidbar (d. h., es gibt kein allgemeines Verfahren, das für beliebige Termersetzungssysteme angibt, ob es konfluent ist oder nicht).*

Der **Beweis** dieses Satzes wird im allgemeinen geführt durch Zurückführung des Problems z. B. auf die Unentscheidbarkeit des Wortproblems für Halbgruppen.

**Bemerkung**

1. Im Jahre 1984 haben M. DAUCHET und S. TISON bewiesen, daß die Konfluenz für *Grundtermersetzungssysteme*, d.h in den Regeln kommen keine Variablen vor, entscheidbar ist.
   1987 haben sie gemeinsam mit T. HEUILLARD und P. LESCANNE einen Entscheidbarkeitsalgorithmus für die Konfluenz von Grundtermersetzungssystemen vorgelegt.
2. Die Konfluenz ist eine *modulare* Eigenschaft in dem Sinne, daß die *disjunkte* Vereinigung zweier Termersetzungssysteme $R_1$ und $R_2$ (d. h., $R_1$ und $R_2$ haben disjunkte Signaturen, also keine gemeinsamen Operatoren bzw. Konstanten) genau dann konfluent ist, wenn $R_1$ und $R_2$ beide konfluent sind.
   Das wurde durch Y. TOYAMA im Jahre 1987 bewiesen. Diese Eigenschaft ist nützlich wegen der im allgemeinen modularen Definition von abstrakten Datentypen.

Für Termersetzungssysteme mit der CHURCH–ROSSER–Eigenschaft ist es naheliegend, zu versuchen, mit „Normalformen“ zu rechnen.

**Definition 3.3**

Sei $(T, R)$ ein Termersetzungssystem. Ein Term $t \in T$ heißt *irreduzibel*, kurz $t\downarrow$, genau dann, wenn es keinen Term $t' \in T$ mit $t \to_R t'$ gibt. Der Term $t'$ heißt eine *Normalform* von $t$ genau dann, wenn $t'\downarrow$ und $t \xrightarrow{*}_R t'$.

**Bemerkung**

Die Normalform eines Terms muß weder existieren noch eindeutig sein.

Wir sehen uns die Verhältnisse im Zusammenhang mit Normalformen am Beispiel an.

**Beispiel 6**

1. Wenn z. B. die Regel $x+y \to y+x$ in $R$ ist, so besitzt ein beliebiger „Summenterm" (mit Variablen) in $R$ keine Normalform.

2. Betrachte das Termersetzungssystem

$$N \to N|$$
$$N \to |$$

   mit $N$ Variable, $|$ Konstante. Hier besitzt $N$ abzählbar unendlich viele Normalformen.

**Bemerkung**

Die Eigenschaft „besitzt Normalform" ist ein Beispiel dafür, daß es für die Untersuchung von Eigenschaften in Termersetzungssystemen nicht nur auf das Regelsystem $R$ ankommt, sondern die Termmenge $T$ ganz wesentlich ist:
Betrachte etwa die Regeln aus Beispiel 4 (Peano–Algebra) und füge die Regel

$$r_5 : \ x+y \to y+x$$

hinzu. Dann besitzt der Term $x+y$ z. B. keine Normalform. Damit ist das Termersetzungssystem $(T_\Sigma(X), R)$ nicht *schwach normalisierend* ($\leftrightarrow_{df}$ jeder Term besitzt eine Normalform), aber $R_0$, d. h. $(T_\Sigma, R)$ ist schwach normalisierend! Denn jeder Grundterm läßt sich zu einer „Zahl" $0^{\prime\prime\cdots\prime}$ reduzieren.

**Satz 3.3**

*Falls $(T, R)$ konfluent ist und $t_1, t_2$ Normalformen von $t \in T$ sind, so gilt $t_1 = t_2$.*

**Beweis**

$t \xrightarrow{*}_R t_1$ und $t \xrightarrow{*}_R t_2$. Wegen Konfluenz existiert ein $t' \in T$ mit $t_1 \xrightarrow{*}_R t'$ und $t_2 \xrightarrow{*}_R t'$. Da $t_1 \downarrow$, muß sein $t_1 = t'$ ($\xrightarrow{*}_R$ reflexiv !!). Analog $t_2 \downarrow$, also $t_2 = t'$. Damit $t_1 = t_2$. q.e.d.

Also, falls das Termersetzungssystem konfluent ist, so ist die Normalform eindeutig. Für konfluente Termersetzungssysteme wird definiert:

$$t' = \mathrm{NF}(t) : \leftrightarrow_{df} t' \text{ Normalform von } t.$$

$\mathrm{NF} : T \to T$ ist eine partielle Funktion. Eine Normalform existiert, wenn die „Rechnung" abbricht.

**Definition 3.4**

Ein Termersetzungssystem $(T, R)$ heißt *noethersch*[1] oder *terminierend*, falls es keine unendlichen Reduktionsfolgen in $R$

$$t_0 \to_R t_1 \to_R t_2 \to_R \ldots$$

gibt. Man nennt dann auch $R$ bzw. $\to_R$ *noethersch*.

[1] Emmy Noether: deutsche Mathematikerin 1882 – 1935

**Bemerkung**

Für die Existenz einer Normalform ist es *nicht notwendig*, daß es keine unendlich langen Reduktionsfolgen gibt:
Dazu betrachten wir wieder Beispiel 4 mit den Regeln $r_1 - r_4$ und das Kommutativgesetz (Regel $r_5$) dazu als Regelsystem $R$. Für das entsprechende Termersetzungssystem $R_0$ stellen wir fest, daß dann unendliche Reduktionsfolgen möglich sind und trotzdem eine Normalform existiert:

$$\rightarrow_R 0 + 0' \rightarrow_{r_2} (0+0)' \rightarrow_{r_5} \ldots \rightarrow_{r_5} (0+0)' \rightarrow_{r_1} 0'$$

$$\rightarrow_R 0 + 0' \leftrightarrow_{r_5} \ldots \leftrightarrow_{r_5} 0' + 0 \rightarrow_{r_1} 0'$$

$0'$ ist hier also Normalform von $0 + 0'$.
Aber selbstverständlich ist die Unmöglichkeit unendlicher Reduktionsfolgen *hinreichend* für die Existenz von Normalformen.

Es gibt sogar schon Ein–Regel–Termersetzungssysteme, die *schwach normalisierend* sind wie obiges $R_0$ (d. h., jeder Term besitzt eine Normalform), aber nicht noethersch sind:

$$R: \qquad F(a,(F(x,y)) \rightarrow F(x,F(x,F(b,b)))$$

ist *nicht noethersch*. Zum Beispiel haben wir

$$F(a,F(a,F(b,b))) \rightarrow_R F(a,F(a,F(b,b))) \rightarrow_R \ldots,$$

aber dieser Term hat eine Normalform:

$$F(a,\underline{F(a,F(b,b))}) \rightarrow_R F(a,F(\overbrace{\ b\ }^{x},\overbrace{F(b,F(b,b))}^{y})) \rightarrow_R F(b,F(b,F(b,b))) \downarrow .$$

**Übungsaufgabe 6**

Zeigen Sie allgemein, daß jeder Term (mit den Operatoren, Konstanten und Variablen $F, a, b, x, y$) in $R$ eine Normalform besitzt.

Für die Existenz von Normalformen ist die Eigenschaft der Termination (die „Noetherizität“) also hinreichend.
Leider gilt auch hier der

**Satz 3.4** (G. Huet; D. S. Lankford 1978)
*Es ist unentscheidbar, ob ein beliebig gegebenes Termersetzungssystem noethersch ist.*

Die Verhältnisse sind zum Teil noch komplizierter als bei der Konfluenz. Es wurde beispielsweise bewiesen:

**Satz 3.5** (N. Dershowitz 1985/87)
*Es ist unentscheidbar, ob ein beliebig gegebenes Termersetzungssystem, das nur zwei Regeln besitzt, noethersch ist.*

Als weitere Verschärfung wurde die Unentscheidbarkeit der Termination für sogar nur Ein–Regel–Termersetzungssysteme durch M. Dauchet 1987 gezeigt.

Im Gegensatz zur Konfluenz ist das Noetherschsein *nicht* einmal eine *modulare* Eigenschaft, wie das folgende Gegenbeispiel von Y. Toyama (1987) zeigt:

$$\begin{array}{ll} R_1: & \text{if } x \text{ then } 1 \text{ else } 0 \text{ fi } \rightarrow \text{if } x \text{ then } x \text{ else } x \text{ fi} \\ R_2: & x \vee y \rightarrow x \\ & x \vee y \rightarrow y \end{array}$$

Es ist leicht einzusehen, daß $R_1$ und $R_2$ beide noethersch sind. Zum Beispiel kann man anführen, daß sich in den Termen der Reduktionsfolgen bei Anwendung von $R_1$ die Anzahl der Vorkommen der Teilzeichenreihe „1 else 0" reduziert, beziehungsweise bei Anwendung von $R_2$ die Zahl der Vorkommen von „$\vee$", was zum Abbruch führen muß.
Aber in $R_1 \cup R_2$ ist folgende unendliche Reduktionsfolge möglich:

$$\begin{array}{l} \text{if } 0 \vee 1 \text{ then } 0 \vee 1 \text{ else } 0 \vee 1 \text{ fi } \rightarrow_{R_2} \text{if } 0 \vee 1 \text{ then } 0 \vee 1 \text{ else } 0 \text{ fi } \rightarrow_{R_2} \\ \text{if } 0 \vee 1 \text{ then } 1 \text{ else } 0 \text{ fi } \rightarrow_{R_1} \text{if } 0 \vee 1 \text{ then } 0 \vee 1 \text{ else } 0 \vee 1 \text{ fi } \rightarrow \ldots \end{array}$$

Allerdings gilt auch hier (analog zur Konfluenz):
Für Grundtermersetzungssysteme (in den Regeln kommen keine Variablen vor) ist die Termination entscheidbar (Huet / Lankford 1978).

Wegen der Unentscheidbarkeit der Termination sind also auch hierfür hinreichende Kriterien für gewisse Klassen von Termersetzungssystemen zu suchen, dafür sind bereits zahlreiche Verfahren entwickelt worden. Vgl. das spätere Kapitel 5 „Termination".

Um ein schwaches Gefühl für die Kompliziertheit der Frage der Termination zu geben, führen wir jetzt einige Beispiele an.

**Beispiel 7**

$$- - x \rightarrow x$$

ist noethersch (Zeichenfolge „$--$" wird reduziert).

**Beispiel 8**

$$- - x \rightarrow - - - x$$

ist trivialerweise nicht noethersch.

**Beispiel 9**

Als Holländisches Nationalflaggenspiel bekannt:

$$\begin{array}{rcl} wei\ss, rot & \rightarrow & rot, wei\ss \\ blau, rot & \rightarrow & rot, blau \\ blau, wei\ss & \rightarrow & wei\ss, blau \end{array}$$

Terme sind hier beliebige Folgen von Konstanten. Die Folgenbildung muß man hier als "beliebigstelligen" Operator auffassen. Unsere Konstanten sind $wei\ss, rot, blau$.

**Übungsaufgabe 7**

Ist das „Holländische Nationalflaggenspiel" noethersch?

**Beispiel 10**

$$\leftharpoondown (x \circ y) \to (\leftharpoondown\leftharpoondown x \circ y) \circ y$$

**Übungsaufgabe 8**

Ist dieses Ein–Regel–System noethersch ?

Wir kommen in Abschnitt 5.2 auf dieses Beispiel zurück.

**Beispiel 11**

$$\begin{array}{rcl}
\sim\sim p & \to & p \\
\sim (p \wedge q) & \to & (\sim p \vee \sim q) \\
\sim (p \vee q) & \to & (\sim p \wedge \sim q) \\
p \wedge (q \vee r) & \to & (p \wedge q) \vee (p \wedge r) \\
(p \vee q) \wedge r & \to & (p \wedge r) \vee (q \wedge r)
\end{array}$$

Dieses Regelsystem ist sicher sehr schwierig zu behandeln: es gibt sowohl „termverkürzende", als auch „termverlängernde" Reduktionsfolgen. Außerdem ändert sich ständig die Termstruktur. Betrachte zum Beispiel:

$$\sim (F \wedge (\sim W \vee \sim W)) \to \ldots \to \sim F \vee (W \wedge W)$$

$$\sim (F \wedge (W \vee W)) \to \ldots \to (((\sim F \wedge \sim F) \vee (\sim F \wedge \sim W)) \vee ((\sim W \wedge \sim F) \vee (\sim W \wedge \sim W)))$$

**Übungsaufgabe 9**

Führen Sie obige Ableitungen ausführlich aus. Welche Regeln werden angewendet?

Wir kommen auf dieses Beispiel auf Seite 57 zurück.

Zusammenfassend haben wir bisher:
Konfluenz führt zur Eindeutigkeit der Normalform, das Noetherschsein zur Existenz einer Normalform.

**Definition 3.5**

Ein Termersetzungssystem $(T, R)$ heißt *vollständig* (oder *kanonisch*, manchmal auch *konvergent*) genau dann, wenn es konfluent und noethersch ist.

**Folgerung 3.6**

*Für ein vollständiges Termersetzungssystem $(T, R)$ gibt es zu jedem $t \in T$ genau eine Normalform:*

$$t \xrightarrow{*}_R \mathrm{NF}(t)\downarrow .$$

# Kapitel 4

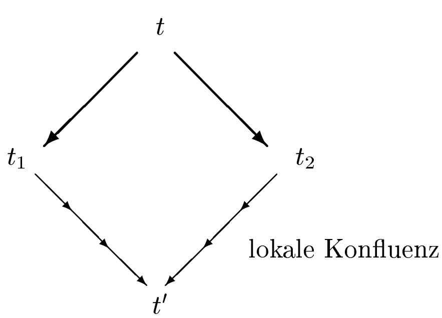
# Vervollständigung von Termersetzungssystemen

## 4.1 Theoretische Grundlagen

Bei noetherschen Termersetzungssystemen gibt es ein einfacheres Kriterium für die Konfluenz:

**Definition 4.1**

Ein Termersetzungssystem $(T, R)$ heißt *lokal konfluent* genau dann, wenn für alle $t, t_1, t_2 \in T$ gilt:
Wenn $t \to_R t_1$ und $t \to_R t_2$, so existiert ein $t' \in T$ mit $t_1 \xrightarrow{*}_R t'$ und $t_2 \xrightarrow{*}_R t'$.




**Folgerung 4.1**
*Aus der Konfluenz folgt stets die lokale Konfluenz.*

**Beweis**
Sei $(T, R)$ konfluent. Betrachte die Terme $t, t_1, t_2 \in T$ mit $t \to_R t_1$ und $t \to_R t_2$, so nach Definition: $t \xrightarrow{*}_R t_1$ und $t \xrightarrow{*}_R t_2$, woraus wegen der Konfluenz die Existenz von $t' \in T$ mit $t_1 \xrightarrow{*}_R t'$ und $t_2 \xrightarrow{*}_R < t'$ folgt. q.e.d.

Hier nun unser einfacheres **Konfluenzkriterium**:

**Satz 4.2** (NEWMAN**'s Lemma** [1942])
*Ist das Termersetzungssystem $(T, R)$ noethersch, so ist es konfluent genau dann, wenn es lokal konfluent ist.*

**Bemerkung**
Die Voraussetzung „noethersch“ ist wesentlich. Ohne sie kann man *nicht* von der lokalen Konfluenz auf die Konfluenz schließen.

*Gegenbeispiel:*
Sei $R = \{A \to B, A \to C, B \to A, B \to D\}$. $R$ ist lokal konfluent. Denn zu $B \leftarrow A \to C$ haben wir $B \to^* C$ und $C \to^* C$. Und zu $D \leftarrow B \to A$ haben wir $D \to^* D$ und $A \to^* A$.
Aber $R$ ist *nicht* konfluent, so hat $A$ zum Beispiel zwei Normalformen:

$$A \to^* C \downarrow \text{ und } A \to^* D \downarrow .$$

Zum **Beweis** von NEWMAN'S Lemma benötigen wir zunächst einen Hilfssatz:

**Satz 4.3**

*Sei $(T, R)$ ein endliches, noethersches Termersetzungssystem. Dann ist für einen beliebigen Term $t \in T$ die Menge*

$$\Delta(t) = \{u \mid t \xrightarrow{*}_R u\}$$

*stets endlich. ($\to_R$ ist „global endlich“.)*

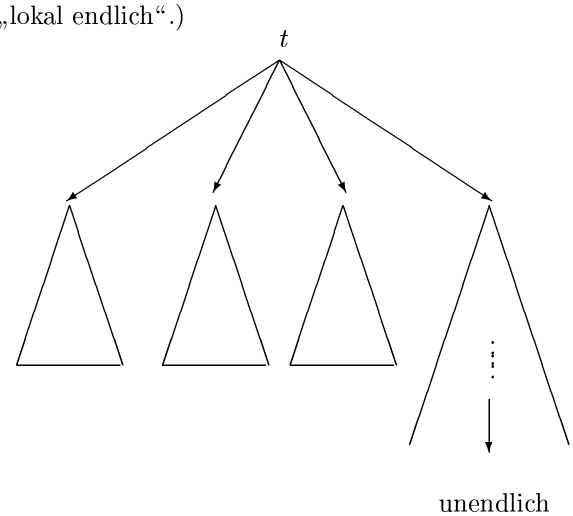
**Beweis**

Das Regelsystem $R$ ist endlich und die Anzahl der Vorkommen von Teiltermen in jedem Term ebenfalls. Dann ist also für jeden Term die Zahl der Redexe, die er enthält, endlich. Also ist für alle $w \in T$

$$\Delta_1(w) = \{v \mid w \to_R v\}$$

endlich. ($\to_R$ heißt dann „endlich verzweigend“ oder „lokal endlich“.)

Angenommen, für ein $t \in T$ sei $\Delta(t)$ unendlich. Sei nun $\Delta_1(t) = \{u_1, \dots, u_n\}$ (siehe oben: $\Delta_1(t)$ muß endlich sein). Dann gibt es also mindestens ein $u_i$ mit $1 \le i \le n$, für das $\Delta(u_i)$ unendlich ist. Wähle ein solches $u_i$ und bezeichne es als $t_1$. Allgemein gibt es für jedes $t_k$ mit $\Delta(t_k)$ unendlich mindestens ein $t_{k+1} \in \Delta(t_k)$, für das auch $\Delta(t_{k+1})$ unendlich ist.




Damit existierten $t_1, t_2, \dots, t_k, t_{k+1}, \dots$ mit $t \to_R t_1 \to_R t_2 \to_R \dots \to_R t_k \to_R t_{k+1} \to_R \dots$ im Widerspruch dazu, daß $R$ noethersch ist. Also ist die Annahme falsch und die Behauptung richtig. q.e.d.

**Folgerung 4.4**

*Zu jedem $t \in T$ gibt es unter obigen Voraussetzungen eine maximale Ableitungstiefe (maximale Länge der Reduktionsfolgen):*
*Sei $\Delta(t) = \{u_1, \dots, u_n\}$. Also ist $t \xrightarrow{*}_R u_i$ und es gibt ein $k_i \in I\!N$ mit $t \xrightarrow{k_i} u_i$. Nun sei*

$$\max(t) =_{df} \max\{k_1, \dots, k_n\}.$$

Damit läßt sich NEWMAN's Lemma nun induktiv beweisen:

**Beweis**

Sei $(T, R)$ noethersch. Es ist nur zu zeigen, daß aus der lokalen Konfluenz die Konfluenz folgt. Wir beweisen induktiv für alle $n \in I\!N$:

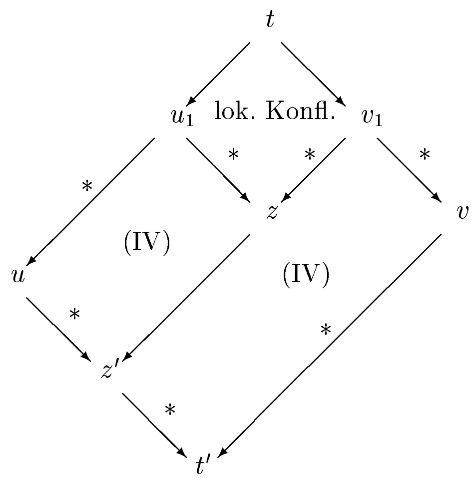
(IB) Für beliebige $t, u, v \in T$ folgt aus $t \xrightarrow{*}_R u$ und $t \xrightarrow{*}_R v$ und $\max(t) \leq n$ stets die Existenz eines $t' \in T$ mit $u \xrightarrow{*}_R t'$ und $v \xrightarrow{*}_R t'$.

(IA) $n = 0$. Dann folgt $t = u = v$. Setze $t' = t$. Daraus folgt die Behauptung.

(IS)

1. $t = u$ oder $t = v$. Für $t = u$ setze $t' = v$. Dann folgt $t \to^* v = t'$ nach Voraussetzung also $u \to^* t'$ und $v \to^* t'$ trivialerweise. Falls $t = v$ setze $t' = u$. Dann folgt die Behauptung analog.
2. $t \neq u$ und $t \neq v$.

   Dann existieren $u_1, v_1$ mit $t \to_R u_1 \xrightarrow{*}_R u$ und $t \to_R v_1 \xrightarrow{*}_R v$. Wegen $\max(t) \leq n$ folgt $\max(u_1) \leq n-1$ und $\max(v_1) \leq n-1$.
   Da $R$ lokal konfluent ist, existiert ein $z \in T$ mit $u_1 \xrightarrow{*}_R z$ und $v_1 \xrightarrow{*}_R z$. Zu $u_1 \xrightarrow{*}_R z$ und $u_1 \xrightarrow{*}_R u$ existiert nach Induktionsvoraussetzung wegen $\max(u_1) \leq n-1$ ein $z' \in T$ mit $z \xrightarrow{*}_R z'$ und $u \xrightarrow{*}_R z'$. Damit ist $v_1 \xrightarrow{*}_R z'$. Außerdem $v_1 \xrightarrow{*}_R v$ und $\max(v_1) \leq n-1$. Nach Induktionsvoraussetzung existiert also $t' \in T$ mit $z' \xrightarrow{*}_R t'$ und $v \xrightarrow{*}_R t'$. Daraus folgt aber $u \xrightarrow{*}_R t'$ und $v \xrightarrow{*}_R t'$.




q.e.d.

Zum Rechnen mit Normalformen sind deren Existenz und Eindeutigkeit anzustreben. Dafür sind Termination ($R$ noethersch) und Konfluenz hinreichend. Angenommen, ein gegebenes Termersetzungssystem $(T, R)$ sei noethersch. Welche Situationen sind dann für die Konfluenz kritisch ?

**Beispiel 12**

1. $x + 0 \to x$
2. $x * (y + z) \to x * y + x * z$

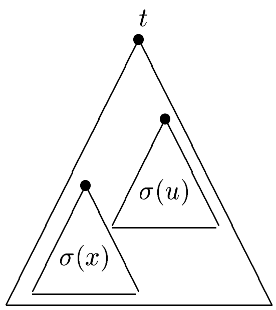
$$\begin{array}{ccc} & & -3 * 1 \\ & \nearrow (1) & \\ -3 * (1+0) & & \\ & \searrow (2) & \\ & & -(3 * 1 + 3 * 0) \end{array}$$

Eine *kritische Situation* kann dann entstehen, wenn es einen Term $t$ gibt, auf den zwei Regeln $u \to v$ und $x \to y$ anwendbar sind, die zu verschiedenen Resultaten führen.

**Allgemein:**

$$\begin{tikzcd} & t_1 \\ t \arrow[ur, "{(u,v)}"] \arrow[dr, "{(x,y)}"'] & \\ & t_2 \end{tikzcd}$$

das heißt, in $t$ muß sowohl ein Redex zu $u$, als auch zu $x$ vorhanden sein. O.B.d.A. kann man voraussetzen, daß $\mathrm{var}(u) \cap \mathrm{var}(x) = \emptyset$ ist. (Man kann dann mit „einheitlicher“ Einsetzung arbeiten.)
Eine kritische Situation *kann* also entstehen, falls es eine Einsetzung $\sigma$ gibt, so daß sowohl $\sigma(u)$ als auch $\sigma(x)$ Teilterme von $t$ sind.

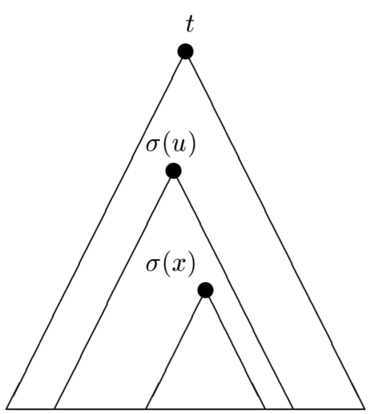



Solche Situationen aber sind *unkritisch*, denn die lokale Konfluenz ist sofort gesichert.
Die Regeln $u \to v$ und $x \to y$ lassen sich hier in beliebiger Reihenfolge nacheinander anwenden. Man beachte dabei, daß sich Teilterme nicht überlappen können.

$\sigma(u), \sigma(x)$ als Teilterme von $t$ führen also nur dann zu einer kritischen Situation, falls sie nicht „disjunkt“sind. Wegen der Termstruktur muß also einer Teilterm des anderen sein. O.B.d.A. sei

$$\sigma(x) \text{ Teilterm von } \sigma(u).$$

**Kritische Situation:**




Ein Anwenden der Regel $u \to v$ verhindert im allgemeinen die Anwendung der Regel $x \to y$ und umgekehrt.
Falls $\sigma(x)$ Teilterm von $\sigma(u)$ ist, so gibt es stets ein solches $t$, Die *kritische Situation* ist also unabhängig von $t$, nur abhängig vom Regelsystem.
(Man beachte: $u, x$ als LHS von Regeln sind *keine* Variablen.)

Daraus ergibt sich die allgemeine

**Definition 4.2**

Sei $(T, R)$ ein Termersetzungssystem mit $T \subseteq T_\Sigma(X)$. Zwei Regeln $(t, t'), (s, s') \in R$ heißen *überlappend* genau dann, wenn Einsetzungen $\varphi, \psi$ existieren und eine Adresse $a \in \mathrm{adr}(t)$ mit

$$t|a \notin X \text{ und } \varphi(t|a) = \psi(s).$$

O.B.d.A. kann man $R$ so wählen, daß die LHS von $R$ paarweise verschiedene Variablen enthalten: für $t \to t'$, $s \to s' \in R$:

$$\mathrm{var}(t) \cap \mathrm{var}(s) = \emptyset.$$

Dann lassen sich obige Einsetzungen $\varphi$ und $\psi$ zu einer Einsetzung $\sigma$ zusammenfassen:

$$\sigma(t|a) = \sigma(s)$$

Außerdem kann man $\sigma$ so finden, daß die Substitution möglichst „minimal" ist:

**Definition 4.3**

Eine Einsetzung $\tau$ heißt *allgemeinster Unifikator* mgu (most general unifier) für die Terme $t$ und $s$ genau dann, wenn

$$\tau(t) = \tau(s) \quad \text{und} \quad \forall \sigma(\sigma(t) = \sigma(s) \to \exists \varphi(\sigma = \varphi \circ \tau)).$$

**Beispiel 13**

Vergleiche dazu obiges Beispiel 12.

$$\begin{aligned} t &= x * (y + z) \\ s &= v + 0 \end{aligned}$$

*Eine* Einsetzung, die die oben angeführten Regeln als überlappend ausweist, ist die oben angewendete:

$$\sigma(x) = 3,\ \sigma(y) = 1,\ \sigma(z) = 0,\ \sigma(v) = 1\,.$$

Ein mgu dafür ist $\tau$ mit $\tau(y) = v,\ \tau(z) = 0$.
Für $\varphi$ entsprechend obiger Definition gilt dann $\varphi(x) = 3,\ \varphi(v) = 1$.

**Definition 4.4 („Superposition")**

Sind $(t, t'),\ (s, s') \in R$ zwei überlappende, o.B.d.A. variablendisjunkte Regeln, so daß $t|a \notin X$ und $s$ mit dem mgu $\tau$ unifizierbar sind, d.h. $\tau(t|a) = \tau(s)$, so heißt die Zweiermenge von Termen

$$\{\, \tau(t[a \leftarrow s']),\, \tau(t')\, \}$$

ein *kritisches Paar* von $R$. Die Menge aller kritischen Paare von $R$ bezeichnen wir mit $\mathrm{CP}(R)$ .

**Bemerkung**

(zur Superposition)

1. Um $\mathrm{CP}(R)$ zu berechnen, ist für alle möglichen Regelpaare in allgemeinster Weise die LHS der ersten Regel mit den nichtvariablen Teiltermen der LHS der zweiten Regel zu unifizieren, falls möglich.

   Ein kritisches Paar resultiert dann aus den zwei möglichen Wegen der Reduktion der gemeinsamen „Instanz":
   Unter den Voraussetzungen wie in der Definition kann man einerseits ableiten $\tau(t) \xrightarrow{*} \tau(t')$ und andererseits aber auch $\tau(s) \xrightarrow{*} \tau(s')$. Damit erhalten wir

$$\begin{array}{ccc} & & \tau(t') \\ & \nearrow & \\ \tau(t) & & \\ & \searrow & \\ & & \tau(t[a \leftarrow s']) \end{array} .$$

2. Die Menge $\mathrm{CP}(R)$ ist für endliche Reduktionssysteme stets endlich. Die Unifikation zweier Terme ist stets entscheidbar und ein mgu stets berechenbar[1] Damit ist auch die Menge $\mathrm{CP}(R)$ stets berechenbar.

**Satz 4.5** (**Huet–Theorem**)

(von Huet 1980; 1970 unter der Voraussetzung „noethersch" von Knuth und Bendix)
*Ein Reduktionssystem $R$ ist genau dann lokal konfluent, wenn für alle $\{p,q\} \in \mathrm{CP}(R)$ gilt:*

$$\text{es gibt ein } w \text{ mit } p \xrightarrow{*} w \text{ und } q \xrightarrow{*} w.$$

**Beweis**

Der Beweisgang ist die exakte Fassung unserer obigen Vorüberlegungen.

($\rightarrow$) Sei $R$ lokal konfluent. Sei $\{p,q\} \in \mathrm{CP}(R)$ zu den überlappenden Regeln $(t,t'),\ (s,s') \in R$ mit den Bezeichnungen wie in der Definition:

$$p = \tau(t[a \leftarrow s']),\ q = \tau(t')\,.$$

Dann gilt

$$\tau(t) \xrightarrow[R]{} p,\ \tau(t) \xrightarrow[R]{} q\,,$$

woraus wegen der lokalen Konfluenz die Behauptung folgt.

($\leftarrow$) Sei für alle $\{p,q\} \in \mathrm{CP}(R)$:

$$\text{es gibt ein } w:\ p \xrightarrow{*} w,\ q \xrightarrow{*} w\,.$$

$R$ sei o.B.d.A. als variablendisjunkt vorausgesetzt. Sei $u, v_1, v_2 \in T$ mit

$$u \xrightarrow[R]{} v_1,\ u \xrightarrow[R]{} v_2,\ v_1 \neq v_2,$$

das heißt, es existieren zwei Adressen $a_1, a_2 \in \mathrm{adr}(u)$ und Regeln $(t,t'),(s,s') \in R$ und es existieren Einsetzungen $\sigma_1, \sigma_2$, so daß

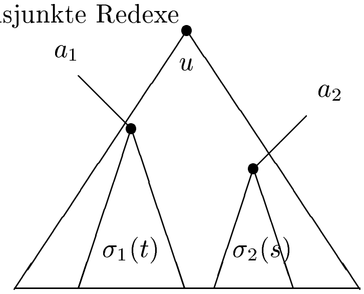
$$\sigma_1(t) = u|a_1,\ \sigma_2(s) = u|a_2$$

und

$$v_1 = u[a_1 \leftarrow \sigma_1(t')],\ v_2 = u[a_2 \leftarrow \sigma_2(s')]\,.$$

Folgende Fälle der Vorkommen der Redexe in $u$ sind möglich:

1. Fall: disjunkte Redexe



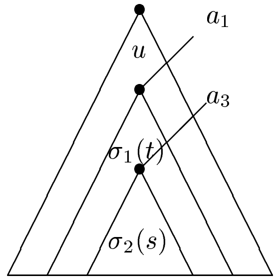

Nach Regelanwendung ändern sich nur die Adressen und Teilterme, die zu den ersetzten Teiltermen gehören. Deshalb ist $v_1|a_2 = \sigma_2(s)$ und $v_2|a_1 = \sigma_1(t)$ und damit auch

$$v_1[a_2 \leftarrow \sigma_2(s')] = v_2[a_1 \leftarrow \sigma_1(t')] \quad \text{setze dies} \quad = v_3\,.$$

Also existiert ein $v_3$ mit $v_1 \xrightarrow[R]{} v_3$ und $v_2 \xrightarrow[R]{} v_3$.

[1] In der sogenannten „Unifikationstheorie" werden u.a. Algorithmen für derartige Unifikationsprobleme untersucht.

2. Fall: nichtdisjunkte Redexe, das heißt, ein Redex ist Teilterm des anderen. O.B.d.A. sei $\sigma_2(s)$ Teilterm von $\sigma_1(t)$.



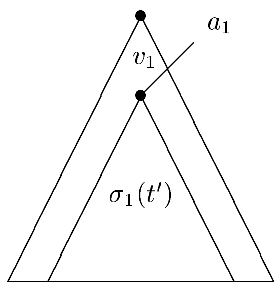

Sei $a_3 \in \mathrm{adr}(\sigma_1(t))$, so daß $\sigma_1(t)|a_3 = \sigma_2(s) = u|a_2$. Dann gilt $v_2|a_1 = \sigma_1(t)[a_3 \leftarrow \sigma_2(s')]$.

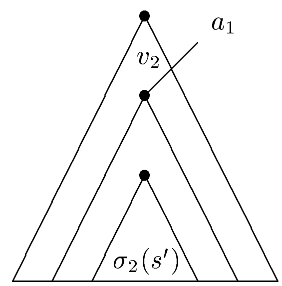
Wir stellen nun die Zwischenbehauptung $(Z)$ auf, die es uns erlaubt, unseren Beweisgang schnell zu Ende zu führen, und die wir dann anschließend beweisen:

$$(Z): \quad \text{Es existiert ein } z \text{ mit } \sigma_1(t') \xrightarrow{*} z \text{ und } v_2|a_1 \xrightarrow{*} z\,.$$

Weil $a_2$ die Adresse eines Teilterms von $u|a_1$ ist, hat man hier

$$v_1 \;=\; u[a_1 \leftarrow \sigma_1(t')] \quad \text{und} \quad v_2 \;=\; u[a_1 \leftarrow v_2|a_1]\,.$$



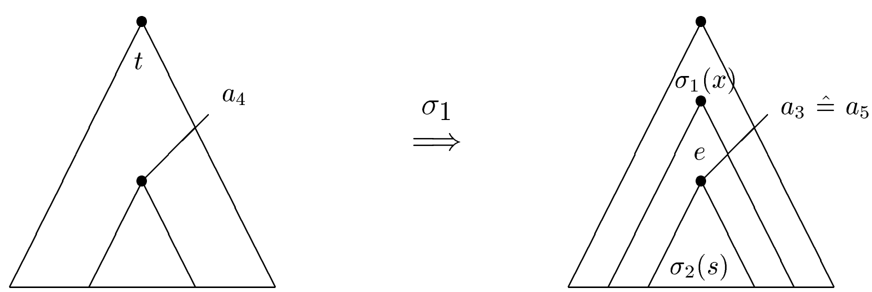

Wegen des invarianten Abschlusses von $\xrightarrow{*}_R$ ist unter obigen Voraussetzungen für

$$v_3 = u[a_1 \leftarrow z]$$

die Behauptung $v_1 \xrightarrow{*} v_3$ und $v_2 \xrightarrow{*} v_3$ erfüllt.

**Beweis (von $Z$):**

Wir betrachten $\sigma_1(t)$. Es ist $\sigma_1(t)|a_3 = \sigma_2(s)$.



Entsprechend der Ausführung der Substitutionen muß es in $t$ eine Adresse $a_4 \in \mathrm{adr}(t)$ geben, die „Ursprung" der Adresse $a_3$ in $\sigma_1(t)$ ist.

Es sind wieder **2 Fälle** möglich:

a) $t|a_4 = x \in X$ und es gibt einen Term $e$ mit $\sigma_1(x) = e$ und $\sigma_2(s)$ ist Teilterm von $e$ an der Adresse $a_5 \in \mathrm{adr}(e)$, wobei $e|a_5 = \sigma_2(s) = \sigma_1(t)|a_3$.
Betrachte die Substitution $\sigma$ mit

$$\sigma(x) = e[a_5 \leftarrow \sigma_2(s')] \quad \text{und} \quad \sigma(y) = \sigma_1(y) \text{ für alle } y \neq x, y \in X.$$

Setze $z = \sigma(t')$.
Es ist $\sigma_1(x) = e \xrightarrow[R]{} \sigma(x)$ vermöge Regel $(s, s') \in R$. Falls nun die Variable $x$ in $t'$ vorkommt, folgt wegen der Kompatibilität von $\xrightarrow{*}_R$

$$\sigma_1(t') \xrightarrow{*}_R \sigma(t'),$$

kommt $x$ nicht in $t'$ vor, ist sowieso $\sigma_1(t') = \sigma(t')$, also gilt stets $\sigma_1(t') \xrightarrow{*}_R z$. Weiter ist wegen

$$\sigma_1(x) = e \xrightarrow[R]{} \sigma(x) = e[a_5 \leftarrow \sigma_2(s')] \quad \text{und} \quad \sigma_1(t)|a_3 = \sigma_2(s) = e|a_5$$

und der Kompatiblität auch

$$\underbrace{\sigma_1(t)[a_3 \leftarrow \sigma_2(s')]}_{v_2|a_1} \xrightarrow{*} \sigma(t)\,.$$

Wegen $(t, t') \in R$ ist auch $\sigma(t) \xrightarrow[R]{} \sigma(t')$, womit sich der zweite Teil der Behauptung

$$v_2|a_1 \xrightarrow{*}_R \sigma(t') = z$$

ergibt.

b) $t|a_4 \notin X$ und $\sigma_2(s) = \sigma_1(t|a_4)$.
Das heißt, die Regeln $(t, t')$ und $(s, s')$ sind nach Definition überlappend. Wegen der Voraussetzung der Variablendisjunktheit der Regeln existiert eine Substitution $\sigma$, so daß also $\sigma(s) = \sigma(t|a_4)\,, \sigma(t) = \sigma_1(t)\,, \sigma(t') = \sigma_1(t')\,, \sigma(s) = \sigma_2(s)$ und $\sigma(s') = \sigma_2(s')$ ($\sigma_1, \sigma_2$ zu $\sigma$ zusammenfassen). Entsprechend der Definition existiert ein kritisches Paar $\{p, q\}$ mit

$$p = \tau(t[a_4 \leftarrow s'])\ ,\ q = \tau(t')$$

und $\tau$ ist mgu für $t|a_4$ und $s$. Also existiert eine Einsetzung $\varphi$ mit $\sigma = \varphi \circ \tau$, d.h.:

$$\begin{aligned} \varphi(p) &= \overbrace{\varphi \circ \tau}^{\sigma}\,(t[a_4 \leftarrow s']) \\ &= \sigma(t)[a_3 \leftarrow \sigma(s')] \\ &= \sigma_1(t)[a_3 \leftarrow \sigma_2(s')] \\ &= v_2|a_1 \end{aligned}$$

und $\varphi(q) = \varphi \circ \tau(t') = \sigma(t') = \sigma_1(t')$. Nach Voraussetzung existiert ein $w$ mit $p \xrightarrow{*} w$ und $q \xrightarrow{*} w$, also auch

$$\varphi(p) \xrightarrow{*} \varphi(w) \quad \text{und} \quad \varphi(q) \xrightarrow{*} \varphi(w).$$

Setze nun $z = \phi(w)$, dann ist $(Z)$ auch in diesem Fall b) erfüllt.

q.e.d.

**Bemerkung**

Der obige Beweis zeigt deutlich, daß zu einer absolut klaren, exakten Beweisführung weitere Hilfssätze, insbesondere zu Adressen und deren Veränderung bei Substitution, nötig wären. Diese wurden im Beweis durch die „Anschauung" ersetzt.

**Beispiel 14**

zur Anwendung des HUET–Theorems: Sei $R$ wie folgt gegeben:

1. $f(x) \to a$
2. $f(x) \to g(f(x))$
3. $g(f(x)) \to f(h(x))$

| CP($R$) | Resultat der Superposition von | gemeinsames Redukt | Begründung |
|---|---|---|---|
| $\{a,\, g(f(x))\}$ | 1./2. | $a$ | $g(f(x)) \xrightarrow[3]{} f(h(x)) \xrightarrow[1]{} a$ |
| $\{g(a),\, f(h(x))\}$ | 1./3. | $g(a)$ | $f(h(x)) \xrightarrow[2]{} g(f(h(x))) \xrightarrow[1]{} g(a)$ |
| $\{g(g(f(x))),\, f(h(x))\}$ | 2./3. | $g(a)$ | obige beide Ableitungen |

Also ist $R$ lokal konfluent. Trotzdem ist $R$ nicht konfluent. ($R$ ist nicht noethersch!)

$$\begin{array}{ccc} & g(f(x)) \longrightarrow g(a)\downarrow & \\ & \nearrow & \\ f(x) & & , \\ & \searrow & \\ & a\downarrow & \end{array}$$

aber $g(a)$ und $a$ haben kein gemeinsames Redukt.

Als Folgerung aus dem NEWMAN–Lemma und dem HUET–Theorem ergibt sich das folgende bereits 1970 von KNUTH und BENDIX bewiesene Theorem

**Satz 4.6** (KNUTH–BENDIX–**Theorem)**

*Ist das Termersetzungssystem $(T, R)$ noethersch, so ist es konfluent genau dann, wenn jedes kritische Paar $\{p, q\} \in \mathrm{CP}(R)$ ein gemeinsames Redukt besitzt.*

**Übungsaufgabe 10**

Führen Sie den **Beweis** des KNUTH–BENDIX–Theorems m.H. des NEWMAN–Lemmas und des HUET–Theorems aus.

**Folgerung 4.7**

*Für noethersche Termersetzungssysteme ist die Konfluenz entscheidbar.*

**Beweis**

Da $R$ endlich ist, ist auch $\mathrm{CP}(R)$ endlich. Bilde für $\{p, q\} \in \mathrm{CP}(R)$ jeweils $NF(p)$ und $NF(q)$. Da $R$ noethersch ist, führt jede Reduktionsfolge in endlich vielen Schritten zu einer Normalfornm. Gilt stets $NF(p) = NF(q)$, so folgt mit dem KNUTH–BENDIX–Theorem die Konfluenz. Ist aber für ein $\{p, q\}$ $NF(p) \neq NF(q)$, so kann $R$ nicht konfluent sein, da sonst die $NF$ eindeutig wäre, und außerdem nach obigem Theorem gleich. q.e.d.

## 4.2 Der Vervollständigungsalgorithmus

Vollständige Termersetzungssysteme (noethersch und konfluent) sind wichtig für die Lösung des Wortproblems:

$$t_1 \equiv_E t_2$$

für Gleichungsspezifikationen $(T, E)$:
Betrachte $E$ als Regelsystem $R$, dann

$$t_1 \equiv t_2 \quad \text{genau dann, wenn} \quad t_1 \overset{*}{\underset{R}{\longleftrightarrow}} t_2\,.$$

Falls $(T, R)$ vollständig ist, so existiert stets eine eindeutige Normalform und ist in endlich vielen Schritten berechenbar:

$$t_1 \overset{*}{\underset{R}{\longleftrightarrow}} t_2 \quad \text{genau dann, wenn} \quad NF(t_1) = NF(t_2)\,.$$

Deswegen steht die Frage nach der Existenz bzw. der Suche nach vollständigen Termersetzungssystemen $(T, R)$, die äquivalent zu gegebenem Regel– oder Gleichungssystem sind („Vervollständigung" – engl. „Completion").

**Definition 4.5**

Zwei Reduktionssysteme $R_1$ und $R_2$ heißen *äquivalent*, kurz

$$R_1 \sim R_2,$$

genau dann, wenn die von ihnen erzeugte Kongruenz dieselbe ist:

$$\overset{*}{\underset{R_1}{\longleftrightarrow}} \;=\; \overset{*}{\underset{R_2}{\longleftrightarrow}}\,.$$

Die Aufgabe der Vervollständigung eines gegebenen Reduktionssystems $R_1$ löst der folgende auf Knuth und Bendix (1970) zurückgehende **Algorithmus**. Er wird oft auch als "Critical–Pair / Completion" – Prozedur (CP/C–Prozedur) bezeichnet.

Wir beschreiben ihn zunächst nur prinzipiell und weisen danach auf naheliegende Ergänzungen hin, die ihn effizienter machen:

Vorausgesetzt seien zwei Prozeduren:

| | | |
|---|---|---|
| $NF(t, R)$: | Funktionsprozedur mit | |
| | Eingangsparametern: | $t \in T$ , Reduktionssystem $R$ |
| | Resultat : | (eine) Normalform von $t$ bezüglich des Termersetzungssystems $(T, R)$: $t \overset{*}{\longrightarrow}_R NF(t, R)\downarrow$. |
| CP$(r, R)$: | Funktionsprozedur mit | |
| | Eingangsparametern: | Reduktionssystem $R$, Regel $r \in R$ |
| | Resultat: | die Menge aller kritischen Paare, die aus der Superposition von $r$ mit allen Regeln von $R$ entstehen. |

Die allgemeine Form des Knuth–Bendix–Vervollständigungsalgorithmus sei nun als Prozedur

$$\text{KB}(R; R')$$

mit dem Eingangsparameter $R$ und dem Ausgang $R'$ ($R$ , $R'$ beides Reduktionssysteme, $R$ noethersch vorausgesetzt) beschrieben:

```
procedure KB(R;R');
          R' := R; C := ∅;
          for all r ∈ R' do C := c ∪ CP(r,R) od ;
                              /* C Menge aller kritischen Paare von R */
1 : while C ≠ ∅ do
          {t,s} :=∈ C;        /* Paar ∈ C nichtdet., aber fair ausgewählt */
          C := C \ {{t,s}};
          t := NF(t,R'), s := NF(s,R');
          if t=s then goto 1
                              /* Beachte HUET--Theorem zu lokaler Konfluenz */
          else  2: choose one of A or B;
                              /* nichtdet. Auswahl, Wahl der Orientierung */
                A: if var(s) ⊆ var(t) then r := t → s; goto 3 fi ;
                B: if var(t) ⊆ var(s) then r := s → t; goto 3 fi ;
                stop with failure
          fi ;
3:        R':= R' ∪ {r};
          if R' nicht noethersch then stop with failure
          else C := C ∪ CP(r,R')
          fi ;
                od ;
stop                          /* mit Erfolg : R' vollständig */
```

Aus den vorangegangenen Sätzen ergibt sich der Beweis des **Korrektheitssatzes** für die KB–Prozedur:

**Satz 4.8**

*Wenn der* KNUTH–BENDIX–*Algorithmus* KB$(R, R')$ *nach endlich vielen Schritten anhält, so entweder „with failure“ oder das erzeugte Reduktionssystem $R'$ ist konfluent und noethersch und es gilt*

$$R' \sim R.$$

**Bemerkung**

Die KB–Prozedur muß nicht terminieren. Falls sie „with failure“ stoppt, kann sie bei anderer Wahl der Orientierungen (Schritt 2 / d.h. bei anderer Termordnung) eventuell erfolgreich sein.

## Varianten und Verbesserungen

1. **Zur Wahl der Orientierung**

   (a) Einbau von Backtracking:
   Wenn sich im 3. Schritt $R'$ als nicht noethersch erweist, so gehe zu einem zuvor liegenden Schritt 2 der Orientierungswahl zurück, wähle anders und prüfe erneut. Man beachte, daß Noetherschsein keine modulare Eigenschaft ist. Erst wenn keine andere Orientierungswahl mehr möglich ist, **stop** `with failure`.

(b) Einführung neuer Operatoren:
Auch wenn weder `A` noch `B` in Schritt 2 anwendbar sind, ist es noch nicht unbedingt erforderlich, erfolglos anzuhalten. Man kann *neue Operatoren* einführen und mit jedem zwei neue Gleichungen (d.h., die Signatur erweitern und eine sogenannte „konservative Erweiterung“ der Gleichungstheorie vornehmen).
Sei $\mathrm{var}(s) \cap \mathrm{var}(t) = \{x_1, \ldots, x_n\}$ mit $\mathrm{sort}(x_i) = s_i$. Sei $z = \mathrm{sort}(t) = \mathrm{sort}(s)$. Dann führe einen neuen Operator $h$ ein mit $\alpha(h) = (s_1, \ldots, s_n, z)$ und zwei neue Regeln:

$$s \to h(x_1, \ldots, x_n) \qquad t \to h(x_1, \ldots x_n)$$

und setze mit diesen fort. In der Literatur sind Fälle dokumentiert, wo dann ein erfolgreicher Abschluß möglich ist.

2. Zur **Normalisierung** der „alten“ Regeln mit den neu gewonnenen:

(a) im 3. Schritt *vor* Bildung der neuen kritischen Paare:
für jede Regel $u \to v \in R'$ erst

$$v := NF(v, R')$$

ausführen.

(b) im 3. Schritt *nach* Bildung der neuen kritischen Paare:
Streichen aller Regeln aus $R'$ (außer $r$), deren LHS eine Instanz von $r$ enthält:
Das sind nämlich die Regeln, deren Superposition mit $r$ zu einem kritischen Paar geführt haben und damit *später* zu einer neuen Regel (mit der äquivalenten Wirkung) Anlaß geben.

$$r \;=\; t \to s \qquad\qquad \begin{tikzcd} & C[t] \arrow[dl, "r"'] \arrow[ddr, "r'"] & \\ C[s] & & \\ & & t' \end{tikzcd}$$

Regel $r'$ streichen: damit wäre $C[s] \equiv_R t'$ nicht mehr gesichert, aber durch die Regel

$$C[s] \to t' \quad \text{oder} \quad t' \to C[s]\,,$$

die im 2. Schritt später entsteht, ist diese Äquivalenz wieder ableitbar.

3. Zur **Verallgemeinerung des Inputs**

(a) Statt von einem noetherschen Reduktionssystem $R$ kann man von einem endlichen Gleichungssystem $E$ ausgehen und ein dazu äquivalentes vollständiges Reduktionssystem $R'$ suchen:

$$\mathrm{KB1}(E;\, R')$$

Im Initialschritt setze man

$$R' \;:=\; \emptyset;\; C \;:=\; E;$$

und fahre sonst genauso fort. Man beachte, daß kritische Paare abgeleitete Gleichungen sind, die zu Regeln gemacht werden müssen.

(b) Man kann beides kombinieren und sowohl von einem Gleichungssystem $E$ und einem dazu konsistenten, noetherschen Regelsystem $R$ ausgehen (d.h. als Gleichungssystem ist $E \cup R$ widerspruchsfrei, es besitzt also nichttriviale Modelle):

$$\mathrm{KB2}(E, R;\, R')$$

Initial wäre dann zu setzen:

$$R' \;:=\; R;\; C \;:=\; E;$$

Man erhält in $R'$ Regeln, die ebenfalls zu $E$ konsistent sind.

4. Zum **Test auf Noetherschsein (Termination)**
   Das Hauptproblem des Algorithmus ist der in jedem Schritt (d.h., nach Gewinnung einer neuen Regel $r$) erforderliche *Beweis der Termination* von $R'$. Damit der Algorithmus korrekt ist, d.h., bei „Stop mit Erfolg“ ein vollständiges $R'$ liefert, muß jedes Zwischenresultat $R'$ noethersch sein !! (Beachte das Knuth–Bendix–Theorem.) Um dies abzusichern, gibt es verschiedene Methoden. Die am häufigsten benutzte ist die Nutzung einer partiellen Ordnung (Halbordnung) $\succ$ über $T$, die als weiterer Eingangsparameter in die KB–Prozedur eingeht, diese sichert bei jeder Wahl von $r = t \to s$ durch $t \succ s$, daß das resultierende $R'$ wieder noethersch ist. Dazu vergleiche Kapitel 5 der Vorlesung. Man erhält

   $$\mathrm{KB3}(R, \succ; R'),$$

   indem in $\mathrm{KB}(R, R')$ die Schritte 2 und 3 ersetzt werden durch:

```
            else 2: if t ≻ s then r := t → s;
                    else if s ≻ t then r := s → t
                    else stop with failure
                                 /* t, s sind bezüglich ≻ nicht vergleichbar */
                    fi ;
            fi ;
            R' := R' ∪ {r};
            C := C ∪ CP(r,R')
                    od ;
stop                             /* mit Erfolg : R' vollständig */
```

## Vervollständigung der Gruppenaxiome

**Beispiel 15**

für die Ausführung des Knuth–Bendix–Algorithmus:

Gegeben seien die bekannten **Gruppenaxiome** als Gleichungssystem $E$:

$$\begin{array}{rcl} e \circ x & = & x \\ x^{-1} \circ x & = & e \\ x \circ (y \circ z) & = & (x \circ y) \circ z \end{array}$$

Als Regelsystem bietet sich folgende Orientierung an:

$$\begin{array}{rllcl} R: & 1. & e \circ x & \to & x \\ & 2. & x^{-1} \circ x & \to & e \\ & 3. & (x \circ y) \circ z & \to & x \circ (y \circ z) \end{array}$$

$R$ ist noethersch (vergleiche auch nächstes Kapitel für den Beweis).

Wir führen nun die KB–Prozedur $\mathrm{KB}(R, R')$ „per Hand“ aus.

Die Menge $C$ aller kritischen Paare wird dabei durch Superposition der Regeln berechnet. Dazu betrachten wir jeweils die Superposition der Regel $m$ mit der Regel $n$, im folgenden angedeutet durch $m./n.$ .

- Superposition 1./1., 2./2., 3./3., 1./2. : –

- Superposition 1./3. :

$$\left.\begin{array}{lll|l} (e \circ y) \circ z & \to_{1.} & y \circ z & \to y \circ z \\ & \to_{3.} & e \circ (y \circ z) & \to_{1.}\ y \circ z \end{array}\right\} \text{Hier sind die Normalformen gleich.}$$

$\{y \circ z, e \circ (y \circ z)\}\ \ \to : -$

- Superposition 2./3. :

$$\left.\begin{array}{lll|l} (x^{-1} \circ x) \circ z & \to_{2.} & e \circ z & \to z \\ & \to_{3.} & x^{-1} \circ (x \circ z) & \to x^{-1} \circ (x \circ z) \end{array}\right\} \text{Hier wird eine neue Regel gebraucht.}$$

$\{e \circ z, x^{-1} \circ (x \circ z)\}$ ist das kritische Paar, das durch die Superposition von Regel 2. mit Regel 3. entsteht. Die neue Regel wird vom Komplizierten zum Einfachen gewählt:

$$4.\ x^{-1} \circ (x \circ z) \to z$$

Es ergibt sich somit das neue Regelsystem:

$$\begin{array}{llllll} R': & 1. & e \circ x & \to & x \\ & 2. & x^{-1} \circ x & \to & e \\ & 3. & (x \circ y) \circ z & \to & x \circ (y \circ z) \\ & 4. & x^{-1} \circ (x \circ z) & \to & z \end{array}$$

- Superposition 1./4. :

$$\left.\begin{array}{lll|l} (e^{-1} \circ (e \circ z)) & \to_{1.} & e^{-1} \circ z & \to e^{-1} \circ z \\ & \to_{4.} & z & \to z \end{array}\right\} \text{kritisches Paar}$$

Es entsteht das kritische Paar $\{e^{-1} \circ z, z\}$.

- Superposition 2./4. :

$$\left.\begin{array}{lll|l} (y^{-1})^{-1} \circ (y^{-1} \circ y) & \to_{2.} & (y^{-1})^{-1} \circ e & (y^{-1})^{-1} \circ e \\ & \to_{4.} & y & y \end{array}\right\} \text{kritisches Paar}$$

Als kritisches Paar ergibt sich $\{(y^{-1})^{-1} \circ e, y\}$.

- Superposition 3./4., 4./4. : –
- Wir nehmen die bei der vorangegangenen Unifikation entstandenen kritischen Paare als Regeln zu unserem bisherigen Regelsystem $R'$ hinzu und bekommen:

$$\begin{array}{llllll} R': & 1. & e \circ x & \to & x \\ & 2. & x^{-1} \circ x & \to & e \\ & 3. & (x \circ y) \circ z & \to & x \circ (y \circ z) \\ & 4. & x^{-1} \circ (x \circ z) & \to & z \\ & 5. & e^{-1} \circ z & \to & z \\ & 6. & (y^{-1})^{-1} \circ e & \to & y \end{array}$$

- Superposition 1./5. : –
- Superposition 2./5. :

$$\left.\begin{array}{lll|l} e^{-1} \circ e & \to_{2.} & e & \to e \\ & \to_{5.} & e & \to e \end{array}\right\} \text{Normalform ist gleich, also kein kritisches Paar.}$$

- Superposition 3./5. :

$$\left.\begin{array}{lll|l} (e^{-1} \circ y) \circ z & \to_{3.} & e^{-1} \circ (y \circ z) & \to y \circ z \\ & \to_{5.} & y \circ z & \to y \circ z \end{array}\right\} \text{kein kritisches Paar}$$

- Superposition 4./5. :

$$\left.\begin{array}{lll|l} (e^{-1})^{-1}\circ(e^{-1}\circ z) & \to_{4.} & z & \to z \\ & \to_{5.} & (e^{-1})^{-1}\circ z & \to z \end{array}\right\}\text{kritisches Paar}$$

$\{(e^{-1})^{-1}\circ z, z\}$

- Superposition 1./6., 2./6. : –
- Superposition 3./6. :

$$\left.\begin{array}{lll|l} ((y^{-1})^{-1}\circ e)\circ z & \to_{3.} & (y^{-1})^{-1}\circ(e\circ z) & \to_{1.} (y^{-1})^{-1}z \\ & \to_{6.} & y\circ z & \to y\circ z \end{array}\right\}\text{kritisches Paar}$$

$\{(y^{-1})^{-1}z, y\circ z\}$

- Superposition 4./6. :

$$\left.\begin{array}{lll|l} (((y^{-1})^{-1})^{-1}\circ((y^{-1})^{-1}\circ e) & \to_{4.} & e & \to e \\ & \to_{6.} & (((y^{-1})^{-1})^{-1}\circ y & \to (((y^{-1})^{-1})^{-1}\circ y \end{array}\right\}\text{kritisches Paar}$$

$\{(((y^{-1})^{-1})^{-1}\circ y, e\}$

- Superposition 5./6., 6./6. : –
- Die Wahl des nächsten Paares als Regel kann nichtdeterministisch erfolgen (d.h. nicht zwangsläufig von oben). Wähle nun als neue Regel

$$7.\ \ (y^{-1})^{-1}\circ z \to y\circ z\ .$$

Damit ergibt sich als neues Regelsystem:

$$\begin{array}{rllcl} R': & 1. & e\circ x & \to & x \\ & 2. & x^{-1}\circ x & \to & e \\ & 3. & (x\circ y)\circ z & \to & x\circ(y\circ z) \\ & 4. & x^{-1}\circ(x\circ z) & \to & z \\ & 5. & e^{-1}\circ z & \to & z \\ & 6. & (y^{-1})^{-1}\circ e & \to & y \\ & 7. & (y^{-1})^{-1}\circ z & \to & y\circ z \end{array}$$

- Superposition 1./7. : –
- Superposition 2./7. :

$$\left.\begin{array}{lll|l} (y^{-1})^{-1}\circ y^{-1} & \to_{2.} & e & \to e \\ & \to_{7.} & y\circ y^{-1} & \to y\circ y^{-1} \end{array}\right\}\text{kritisches Paar}$$

$\{y\circ y^{-1}, e\}$

- Superposition 3./7. :

$$\left.\begin{array}{lll|l} ((y^{-1})^{-1}\circ z)\circ x & \to_{3.} & (y^{-1})^{-1}\circ(z\circ x) & \to_{7.} y\circ(z\circ x) \\ & \to_{7.} & (y\circ z)\circ x & \to_{3.} y\circ(z\circ x) \end{array}\right\}\text{kein kritisches Paar}$$

- Superposition 4./7. :

$$\left.\begin{array}{lllll|l} ((y^{-1})^{-1})^{-1}\circ((y^{-1})^{-1}\circ z) & \to_{4.} & z & & & \to z \\ & \to_{7.} & ((y^{-1})^{-1})^{-1}\circ(y\circ z) & \to_{4.} & y^{-1}\circ(y\circ z) & \to_{4.} z \end{array}\right\}$$

Kein kritisches Paar.

Man beachte, daß bei der Superposition zweier Regeln die Unifikation an mehreren Stellen möglich sein kann bzw. mal die eine, mal die andere Regel als erste genommen werden muß. Mit den Regeln 4. und 7. haben wir hierfür ein konkretes Beispiel! Neben der oben angeführten Unifikation, die zu keinem kritischen Paar führt, finden wir noch:

$$\left.\begin{array}{lll|l} (y^{-1})^{-1} \circ (y^{-1} \circ z) & \to_{4.} & z & \to z \\ & \to_{7.} & (y \circ (y^{-1} \circ z) & \to (y \circ (y^{-1} \circ z) \end{array}\right\} \text{kritisches Paar}$$

$\{(y \circ (y^{-1} \circ z), z\}$

- Superposition 5./7. : –
- Superposition 6./7. :

$$\left.\begin{array}{lll|l} (y^{-1})^{-1} \circ e & \to_{6.} & y & \to y \\ & \to_{7.} & y \circ e & \to y \circ e \end{array}\right\} \text{kritisches Paar}$$

$\{y \circ e, y\}$

- Superposition 7./7. : –
- Als neue Regel nehmen wir nun $y \circ e \to y$ in $R'$ auf.

$$\begin{array}{llllll} R': & 1. & e \circ x & \to & x \\ & 2. & x^{-1} \circ x & \to & e \\ & 3. & (x \circ y) \circ z & \to & x \circ (y \circ z) \\ & 4. & x^{-1} \circ (x \circ z) & \to & z \\ & 5. & e^{-1} \circ z & \to & z \\ & 6. & (y^{-1})^{-1} \circ e & \to & y \\ & 7. & (y^{-1})^{-1} \circ z & \to & y \circ z \\ & 8. & y \circ e & \to & y \end{array}$$

- Superposition 1./8., 2./8. : –
- Superposition 3./8. :

$$\left.\begin{array}{lll|l} (x \circ e) \circ z & \to_{8.} & x \circ z & \to x \circ z \\ & \to_{3.} & x \circ (e \circ z) & \to_{1.} x \circ z \end{array}\right\} \text{kein kritisches Paar}$$

- Superposition 4./8. :

$$\left.\begin{array}{lll|l} x^{-1} \circ (x \circ e) & \to_{4.} & e & \to e \\ & \to_{8.} & x^{-1} \circ x & \to_{2.} e \end{array}\right\} \text{kein kritisches Paar}$$

- Superposition 5./8. :

$$\left.\begin{array}{lll|l} e^{-1} \circ e & \to_{5.} & e & \to e \\ & \to_{8.} & e^{-1} & \to e^{-1} \end{array}\right\} \text{kritisches Paar}$$

$\{e^{-1}, e\}$

- Superposition 6./.8 :

$$\left.\begin{array}{lll|l} (y^{-1})^{-1} \circ e & \to_{6.} & y & \to y \\ & \to_{8.} & (y^{-1})^{-1} & \to (y^{-1})^{-1} \end{array}\right\} \text{kritisches Paar}$$

$\{(y^{-1})^{-1}, y\}$

Die Regel 8. macht die Regel 6. überflüssig (vgl. dazu auch die Bemerkungen zu Varianten der KB–Prozedur, speziell zur Normalisierung Teil b).

- Superposition 7./8. :

$$\left.\begin{array}{llll} (y^{-1})^{-1} \circ e & \rightarrow_{7.} & y \circ e & \Big| \rightarrow y \\ & \rightarrow_{8.} & (y^{-1})^{-1} & \Big| \rightarrow (y^{-1})^{-1} \end{array}\right\} \text{kritisches Paar ist schon vorhanden}$$

- Superposition 8./8. : –

**Übungsaufgabe 11**

Führen Sie die Berechnungen des Beispiels 15 zur Vervollständigung der Gruppenaxiome bis zum erfolgreichen Abschluß weiter.
(Zum Vergleich geben wir im folgenden ein vollständiges Regelsystem für Gruppen an.)

Das folgende aus 10 Regeln bestehende Reduktionssystem ist zu dem Gruppenaxiomen– Regelsystem $R$ äquivalent und ist vollständig. Wenn man entsprechend der KB–Prozedur die obigen Berechnungen fortführt, kommt man nach endlich vielen Schritten und nach eventueller Umbenennung der Regeln und eventuell auch der Variablen zu demselben Resultat.

Es ist das ’ klassische‘ bereits von Knuth und Bendix angeführte Beispiel:

$$\begin{array}{lllll} R': & 1. & e \circ x & \rightarrow & x \\ & 2. & x^{-1} \circ x & \rightarrow & e \\ & 3. & (x \circ y) \circ z & \rightarrow & x \circ (y \circ z) \\ & 4. & x^{-1} \circ (x \circ z) & \rightarrow & z \\ & 5. & y \circ e & \rightarrow & y \\ & 6. & e^{-1} & \rightarrow & e \\ & 7. & (y^{-1})^{-1} & \rightarrow & y \\ & 8. & y \circ y^{-1} & \rightarrow & e \\ & 9. & y \circ (y^{-1} \circ x) & \rightarrow & x \\ & 10. & (x \circ y)^{-1} & \rightarrow & y^{-1} \circ x^{-1} \end{array}$$

## Anwendungen des Knuth–Bendix–Algorithmus (Ausblick)

1. Vervollständigung von Gleichungs–Axiomensystemen zur **Lösung des Wortproblems** der betreffenden Gleichungstheorie:

   Gilt $t_1 = t_2$?

   In der Vergangenheit wurden viele Axiomensysteme vervollständigt.

   Es existieren aber auch wichtige Algebren, für die bisher keine vollständige Axiomensysteme bekannt sind, z.B. modulare Verbände, Lie–Algebren, Körper. Für letztere ist das klar, da die Körper keine gleichungsdefinierbare Klasse bilden.

2. **Abstrakte Datentypen**
   Gleichungsspezifizierte abstrakte Datentypen können „direkt“ implementiert werden, wenn für das zugehörige Wortproblem ein Entscheidungsalgorithmus existiert. Das heißt, wenn sich die algebraische Spezifikation eines abstrakten Datentyps vervollständigen läßt, so ist eine direkte Implementation möglich. Man kann dann in dem abstakten Datentyp „automatisch“ rechnen lassen. Deshalb spricht man in diesem Zusammenhang auch von der *operationalen Semantik* der abstrakten Datentypen.

3. **Induktionslose Induktion**

Die Methode der sogenannten induktionslosen Induktion geht auf GOGUEN (1980) zurück. Es ist eine Methode, die zum Beweis der Gültigkeit einer Gleichung

$$t_1 = t_2$$

in dem initialen Modell eines Gleichungssystems $E$ entwickelt wurde. Man wendet den KNUTH–BENDIX–Algorithmus auf $E \cup \{t_1 = t_2\}$ an. Unter gewissen Voraussetzungen an die Signatur $\Sigma$ und an die Gleichungen ist $t_1 = t_2$ gültig in $\underline{T_{\Sigma,E}}$ (initiale Algebra von $\underline{\underline{\mathrm{Alg}}}_{\Sigma,E}$ genau dann, wenn der Algorithmus stoppt, ohne einen Widerspruch erzeugt zu haben (z.B. **true** = **false**).

Die Gleichung $t_1 = t_2$ gehört in diesem Fall zur sogenannten „induktiven Hülle" von $E$.

Beispielsweise gehören die Gleichungen

$$\begin{array}{lcl} x + y & = & y + x \\ x + (y + z) & = & (x + y) + z \end{array}$$

zur induktiven Hülle von Beispiel 4 auf Seite 13, d.h. sie gelten in dem initialen Modell (natürliche Zahlen mit $+, *, 0, '$). Die Gleichungen gehören aber nicht zur Folgerungshülle von $E$, d.h. sie sind auch nicht ableitbar aus $r_1, r_2, r_3, r_4$. Üblicherweise beweist man ihre Gültigkeit im initialen Modell durch vollständige Induktion[2] !

[2] vgl. dazu auch die entsprechenden Ausführungen in der Vorlesung „Algebraische Spezifikation" des Autors

# Kapitel 5

# Termination von Termersetzungssystemen

Im vorigen Kapitel haben wir die Schlüsselrolle der Eigenschaft des „Noetherschseins“, der Termination von Termersetzungssystemen gesehen, u.a. die Notwendigkeit eines „noethersch?“–Tests in der CP/C–Prozedur.

Andererseits ist die generelle Unentscheidbarkeit dieser Frage bekannt. Daraus ergibt sich die Suche nach möglichst umfassenden hinreichenden Kriterien für die Termination von Termersetzungssystemen.

Dabei spielen Ordnungsrelationen eine besondere Rolle, deshalb im folgenden zunächst eine Zusammenstellung der in diesem Zusammenhang gebrauchten Begriffe.

## 5.1 Ordnungsrelationen

Eine binäre Relation $\succ$ über einer Menge $M$ ($\succ \subseteq M \times M$) heißt (irreflexive) *partielle Ordnung*, *Halbordnung* oder kurz *Ordnung* in $M$ genau dann, wenn sie irreflexiv und transitiv ist. (Man beachte, daß aus der Irreflexivität und der Transitivität auch die Antisymmetrie folgt.)

Ist $\succ$ eine Ordnung, so bezeichnet $\prec$ deren inverse Relation

$$\prec \; = \; \succ^{-1} \; ,$$

sie ist ebenfalls Ordnung.

$\succeq$ bezeichnet die Relation über $M$ mit

$$x \succeq y \quad \text{genau dann, wenn} \quad x \succ y \text{ oder } x = y \; .$$

$\succeq$ ist dann reflexiv, antisymmetrisch und transitiv und heißt eine *reflexive* (Halb)*Ordnung*.

Eine (partielle) Ordnung heißt *totale Ordnung* in $M$ genau dann, wenn für alle $x, y \in M$ gilt

$$x \succ y \text{ oder } y \succ x \text{ oder } x = y \; .$$

Eine (partielle) Ordnung $\succ$ heißt *noethersch* (in der englischsprachigen Literatur: “well–founded“) in $M$ genau dann, wenn es keine unendlichen absteigenden Folgen

$$x_1 \succ x_2 \succ x_3 \succ \ldots$$

von Elementen aus $M$ gibt.

**Beispiel 16**

$>$ in $I\!N$ ist totale Ordnung und noethersch, die inverse Relation $<$ allerdings nicht.

Aber $>$ in $Z\!\!Z$ ist nicht noethersch : $-3 > -4 > -5 > \dots$

Auch in $R^+$ (Menge der positiven rationalen Zahlen) ist $>$ nicht noethersch.

## 5.2 Einbettungsrelation und Simplifikationsordnungen

**Satz 5.1**

*Sei $R$ ein Reduktionssystem. Dann ist $R$ noethersch genau dann, wenn es keinen Grundterm $t_0 \in T_\Sigma$ gibt, der eine unendliche Reduktionsfolge besitzt.*

**Beweis**

($\rightarrow$) $R$ noethersch. Dann existieren keine unendlichen Reduktionsfolgen

$$t_0 \rightarrow t_1 \rightarrow t_2 \rightarrow \dots,$$

also auch kein solches $t_0 \in T_\Sigma$.

($\leftarrow$) Ist $R$ nicht noethersch, so existiert eine unendliche Folge

$$t_0' \rightarrow t_1' \rightarrow t_2' \rightarrow \dots$$

mit $t_i' \in T_\Sigma(X)$. Nun betrachten wir eine beliebige Konkretisierung $\rho : X \mapsto T_\Sigma$ und setzen $t_i = \rho(t_i')$. Nach Definition der Reduktionsrelation $\underset{R}{\longrightarrow}$ gilt dann auch

$$t_0 \underset{R}{\longrightarrow} t_1 \underset{R}{\longrightarrow} t_2 \underset{R}{\longrightarrow} \dots,$$

es gibt also einen solchen Grundterm $t_0$.

q.e.d.

**Bemerkung**

Bei Terminationsfragen genügt es also, sich auf Grundterme zu beschränken. Manchmal erscheint es aber auch zweckmäßiger, die Variablen aus der Betrachtung nicht auszuschließen.

Sicher gilt auch der folgende

**Satz 5.2**

*Ein Termersetzungssystem $(T, R)$ ist noethersch genau dann, wenn alle Ableitungen von Redexen von $R$ (Instanzen der linken Seiten von $R$) terminieren.*

**Übungsaufgabe 12**

**Beweisen** Sie den obigen Satz 5.2 !

**Bemerkung**

Für Terminationsuntersuchungen ist es also hinlänglich, die variablenfreien Instanzen der linken Seiten aller Regeln auf deren Verhalten bei Reduktionen zu untersuchen.

Statt nach Terminationskriterien zu suchen, kann man zunächst auch nach Gründen für die *Nichttermination* von Termersetzungssystemen suchen.

Mit Sicherheit ist $R$ nicht noethersch, wenn es Reduktionsfolgen gibt, in denen sich ein Term (auch als Teilterm) wiederholt.

**Definition 5.1**

Eine Reduktionsfolge

$$t_0 \underset{R}{\longrightarrow} t_1 \underset{R}{\longrightarrow} t_2 \underset{R}{\longrightarrow} \ldots \underset{R}{\longrightarrow} t_j \underset{R}{\longrightarrow} \ldots \underset{R}{\longrightarrow} t_k \underset{R}{\longrightarrow}$$

*besitzt Schleifen* (engl.: "loops") genau dann, wenn natürliche Zahlen $j, k$ existieren mit $j < k$ und $t_j$ ist Teilterm von $t_k$.
Gilt speziell $t_j = t_k$, so sagt man, die Reduktionsfolge *besitzt Zyklen* (engl. "cycles").
Man sagt auch, das Termersetzungssystem $(T, R)$ besitzt Schleifen bzw. Zyklen, wenn es eine Reduktionsfolge in $R$ gibt, die Schleifen bzw. Zyklen besitzt.

**Folgerung 5.3**

*Termersetzungssysteme mit Schleifen sind nicht noethersch.*

**Beispiel 17**

1. $-x \to ---x$ besitzt Schleifen.

2. Vergleiche auch das Gegenbeispiel von Toyama auf Seite 22, daß Noetherschsein keine modulare Eigenschaft ist. Die Ableitung dort besitzt Zyklen.

3. Vergleiche das Beispiel 14 zum Huet–Theorem auf Seite 33

$$f(x) \to g(f(x)) \to g(g(f(x))) \to \ldots$$

besitzt Schleifen.

4. Betrachte das Beispiel 10 aus Kapitel 2 (Übung 8).

$$\neg(x \circ y) \to (\neg\neg x \circ y) \circ y$$

Es gilt

$$
\begin{aligned}
\neg(\underline{(\neg\neg x \circ y)} \circ y) & \underset{R}{\longrightarrow} (\neg\underline{\neg(\overbrace{\neg\neg x}^{x} \circ y)} \circ y) \circ y \\
& \underset{R}{\longrightarrow} (\neg(\overbrace{(\neg\neg\neg\neg x \circ y)}^{x} \circ y) \circ y) \circ y \\
& \underset{R}{\longrightarrow} (((\neg\neg(\neg\neg\neg\neg x \circ y) \circ y) \circ y) \circ y) \circ y \\
& \underset{R}{\longrightarrow} \ldots
\end{aligned}
$$

Das ist eine unendliche Reduktionsfolge. Die Regel ist auf die „negierte“ innere Klammer immer wieder anwendbar, diese Reduktionsfolge besitzt aber keine Schleifen ! (Vergleiche die Termstruktur, speziell „$\neg$“ und „(“, die Struktur ändert sich jeweils, so daß kein Term als Teilterm wieder vorkommen kann.)

Also ist ein schwächerer Begriff als der der Schleife, der sich auf den Begriff des Teilterms stützt, notwendig, um die Nichttermination zu charakterisieren.

**Definition 5.2 (homöomorphe Einbettung)**

Seien $s, t \in T$. Dann wird rekursiv definiert:

$$s \trianglelefteq t$$

genau dann, wenn

1. $s = t$ (Beachte: Beide Terme können Variablen oder Konstanten sein.)

oder 2. $t = \sigma t_1 \ldots t_n \quad \wedge \quad \exists i (1 \le i \le n \wedge s \trianglelefteq t_i)$

oder 3. $s = \sigma s_1 \ldots s_n \quad \wedge \quad t = \sigma t_1 \ldots t_n \quad \wedge \quad \forall i (1 \le i \le n \rightarrow s_i \trianglelefteq t_i)$.

Die zugehörige irreflexive Relation bezeichnen wir mit $\triangleleft$: $s \triangleleft t$ genau dann, wenn $s \trianglelefteq t$ und $s \ne t$.

**Bemerkung**

$s \trianglelefteq t$ genau dann, wenn $s$ aus $t$ durch Streichen geeigneter Operatoren und Teilterme hervorgeht.

**Beispiel 18**

$f(a,b) \trianglelefteq g(f(h(a), g(a,b)), c)$

An Hand der Definition überprüfen:

- mit 2.: $f(a,b) \trianglelefteq f(h(a), q(a,b))$
- mit 3.: $a \trianglelefteq h(a)$, $b \trianglelefteq g(a,b)$
- mit 1.: $a \trianglelefteq a$, $b \trianglelefteq b$

Es gilt nun der grundlegende

**Satz 5.4** (KRUSKAL's **Baum–Theorem,** KRUSKAL 1954/60, HIGMAN 1952)

*Sei die gegebene Signatur $\Sigma$ endlich (d.h. nur endlich viele Operatoren). Dann gibt es für jede unendliche Folge von Termen aus $T_\Sigma$ (Grundterme)*

$$t_1, t_2, t_3, \ldots$$

*zwei Terme $t_i$ und $t_k$ mit $i < k$, so daß*

$$t_i \trianglelefteq t_k.$$

Der **Beweis** beruht auf der Theorie der sogenannten Wohlquasiordnungen und dem Beweis, daß die Relation $\trianglelefteq$ eine solche Wohlquasiordnung ist, und ist im Original recht langwierig und nicht als trivial anzusehen.

**Definition 5.3**

Ein Termersetzungssystem $(T, R)$ heißt *selbsteinbettend* genau dann, wenn es eine Reduktionsfolge in $R$ gibt, so daß

$$t_1 \to_R t_2 \to_R \ldots \to_R t_i \to_R \ldots \to_R t_k$$

mit $t_i \trianglelefteq t_k$ für $i < k$ gilt.

**Satz 5.5** (DERSHOWITZ 1982 für Grundterme, PLAISTED 1986 Erweiterung)

*Ein nichtterminierendes Termersetzungssystem ist stets selbsteinbettend.*

**Beweis**

Sei $(T, R)$ nicht noethersch, d.h. es existiert eine unendliche Folge

$$t_1 \to_R t_2 \to_R \ldots$$

mit $t_i \in T$. Falls $T \subseteq T_\Sigma$, folgt daraus mit KRUSKAL's Theorem bereits die Behauptung, weil in der Folge $t_1 \to_R t_2 \to_R \ldots$ nur endlich viele Operatoren vorkommen können, nämlich die Operatoren aus $t_1$ und aus $R$ – beides endlich, so daß alle $t_i$ als Terme über einer endlichen Signatur aufgefaßt werden können.
Weiter gilt wegen der Variablenbedingung an die Regeln von $R$: $\mathrm{var}(t_{j+1}) \subseteq \mathrm{var}(t_j)$, also $\mathrm{var}(t_i) \subseteq t_1$ für alle $i$. Also ist auch die Menge der vorkommenden Variablen endlich. Für die Anwendung des KRUSKAL–Theorems können sie als Konstanten aufgefaßt werden, so daß auch für $T \subseteq T_\Sigma(X)$ die Behauptung folgt.
q.e.d.

**Folgerung 5.6**

*Ist ein Termersetzungssystem nicht selbsteinbettend, so ist es noethersch.*

Leider gilt auch für dieses hinreichende Kriterium:

**Satz 5.7** (PLAISTED 1985)

*Die Eigenschaft der Selbsteinbettung für Termersetzungssysteme ist unentscheidbar.*

Dieses hinreichende Kriterium ist leider aber auch nicht notwendig:

**Beispiel 19**

$$f(f(x)) \to f(g(f(x)))$$

ist selbsteinbettend und trotzdem terminierend!

Die homöomorphe Einbettungsrelation $\trianglelefteq$ spielt beim Beweis der Termination von Termersetzungssystemen eine Schlüsselrolle. Sie soll deshalb noch näher charakterisiert werden.
Im folgenden bezeichne dabei

$$\sqsubseteq \qquad \text{die } \textit{Teiltermrelation} \text{ über } T_\Sigma(X) \ :$$

$t_1 \sqsubseteq t_2$ genau dann, wenn $t_1$ Teilterm von $t_2$ ist.
Wie üblich sei dann $\sqsubset$ die *echte* Teiltermrelation.

**Folgerung 5.8**

*$t_1 \sqsubseteq t_2$ genau dann, wenn ein $C[\ ] \in T_\Sigma(X^L)$ existiert mit $t_2 = C[t_1]$.*

**Lemma 5.9**

*Die Relation $\trianglelefteq$ (homöomorphe Einbettungsrelation) über $T_\Sigma(X)$ ist eine transitive, kompatible Relation, die die Teiltermrelation $\sqsubseteq$ umfaßt.*

**Beweis**

1. $\sqsubseteq \subseteq \ \trianglelefteq$:
   Sei $t' \sqsubseteq t$. Beweis der Behauptung $t' \trianglelefteq t$ induktiv über den Aufbau von $t$:
   - (IA) Für $t \in X$ oder $t \in K$ folgt wegen $t' = t$ sofort $t' \trianglelefteq t$ (wegen Punkt (1) der Definition).
   - (IS) Sei $t = \sigma t_1 \dots t_n$. Für $t' = t$ folgt sofort aus der Definition $t' \trianglelefteq t$. Sei nun $t' \sqsubset t$, d.h. es existiert ein $i$ mit $1 < i < n$, so daß $t' \sqsubseteq t_i$. Nach (IV) folgt also $t' \trianglelefteq t_i$ und damit nach Punkt (2) der Definition auch $t' \trianglelefteq t$.
2. $\trianglelefteq$ ist kompatible Relation:
   Sei $\sigma \in \Sigma_{w,s}$ mit $w = s_1 \dots s_n$ und für $i = 1, \dots, n$ gelte $t'_i, t_i \in T_\Sigma(X)_{s_i}$ und $t'_i \trianglelefteq t_i$. Damit gilt wegen Punkt (3) der Definition $\sigma t'_1 \dots t'_n \trianglelefteq \sigma t_1 \dots t_n$.
3. $\trianglelefteq$ ist transitive Relation: $t'' \trianglelefteq t' \wedge t' \trianglelefteq t \to t'' \trianglelefteq t$:
   - (IA) $t \in X \cup K$. Wegen $t' \trianglelefteq t$ folgt auch $t' \in X \cup K$ und $t' = t$. Analog wegen $t'' \trianglelefteq t'$ auch $t'' = t'$. Also $t'' = t$, d.h. $t'' \trianglelefteq t$.
   - (IS) Sei $t = \sigma t_1 \dots t_n$. Ist $t' = t$, so ist die Behauptung trivial. Für $t' \triangleleft t$ gibt es zwei Fälle:
     - (a) $t' \trianglelefteq t_i$ für ein $i$ mit $1 \leq i \leq n$. Nach Voraussetzung ist $t'' \trianglelefteq t'$. Nach (IV) ist damit $t'' \trianglelefteq t - i$ und nach Definition Punkt (2) folgt $t'' \trianglelefteq t$.
     - (b) $t' = \sigma t'_1 \dots t'_n$ und $\forall i (1 \leq i \leq n \to t'_i \trianglelefteq t_i)$. Nach Voraussetzung ist $t'' \trianglelefteq t'$. Für $t'' = t'$ folgt sofort die Behauptung. Gilt $t'' \triangleleft t$, sind zwei Fälle zu unterscheiden:
       - (b1) $t'' \trianglelefteq t'_j$ für ein $j$ mit $(1 \leq j \leq n)$. Damit ist $t'' \trianglelefteq t'_j \wedge t'_j \trianglelefteq t_j$, woraus mit (IV) $t'' \trianglelefteq t_j$ folgt und damit die Behauptung nach Definition Punkt (2).

(b2) $t'' = \sigma t''_1 \dots t''_n$ und $\forall k(1 \leq k \leq n \rightarrow t''_k \trianglelefteq t'_k)$. Damit ist $\forall i(1 \leq i \leq n \rightarrow t''_i \trianglelefteq t'_i \wedge t'_i \trianglelefteq t_i)$ und mit (IV) wieder $\forall i(1 \leq i \leq n \rightarrow t''_i \trianglelefteq t_i)$, woraus nach Definition Punkt (3) auch $t'' \trianglelefteq t$ folgt.

q.e.d.

**Folgerung 5.10**

*Die Relation $\trianglelefteq$ ist eine reflexive Ordnung über $T_\Sigma(X)$.*

**Lemma 5.11**

*Es gilt*

$$\trianglelefteq \;\subseteq\; (\mathrm{kom}(\sqsubseteq))^+ \ .$$

**Übungsaufgabe 13**

**Beweisen** Sie das Lemma 5.11.

**Satz 5.12**

$$\trianglelefteq = (\mathrm{kom}(\sqsubset))^+$$

*(Die homöomorphe Einbettung ist die kleinste kompatible Ordnung, die die Teiltermordnung umfaßt.)*

**Beweis**

$\sqsubseteq \subseteq \trianglelefteq$ (folgt aus obigem Lemma 5.9). Aus der Monotonieeigenschaft von kom folgt: $\mathrm{kom}(\sqsubseteq) \subseteq \mathrm{kom}(\trianglelefteq) = \trianglelefteq$ (letztere Gleichung folgt ebenfalls aus Lemma 5.9), also gilt wegen der Hülleneigenschaft und der Transitivität (Lemma 5.9) $(\mathrm{kom}(\sqsubseteq))^+ \subseteq \trianglelefteq^+ = \trianglelefteq$, woraus mit obigem Lemma 5.11 die Behauptung folgt.
q.e.d.

Auf der Suche nach *Kriterien für die Termination* von Termersetzungssystemenentdeckt man schnell den relativ trivialen

**Satz 5.13** (Manna / Ness 1969)

*Ein Termersetzungssystem $(T, R)$ ist terminierend genau dann, wenn eine noethersche Ordnung $\succ$ über $T$ existiert, so daß aus*

$$t \xrightarrow[R]{} u \quad \textit{stets} \quad t \succ u$$

*für alle $t, u \in T$ folgt.*

**Beweis**

(1) Sei $(T,R)$ terminierend.
Dann $\xrightarrow[R]{}$ noethersch, also $\xrightarrow[R]{}$ irreflexiv (sonst niemals noethersch), d.h. $\xrightarrow[R]{}^{+}$ ist irreflexiv und transitiv, also eine Ordnung, und ebenfalls noethersch.

(2) Gebe es eine Ordnung $\succ$ wie im Satz.
Dann ist auch $\xrightarrow[R]{}$ noethersch, denn hätte man eine unendliche Kette

$$t_0 \xrightarrow[R]{} t_1 \xrightarrow[R]{} t_2 \xrightarrow[R]{} \dots \quad ,$$

hätte man nach Voraussetzung auch eine unendliche Kette

$$t_0 \succ t_1 \succ t_2 \succ \dots \quad ,$$

was im Widerspruch zu $\succ$ noethersch steht. Also ist $(T,R)$ terminierend.

q.e.d.

Man kann die Forderung der Noetherizität von $\succ$ auch noch abschwächen:
Die Ordnungsrelation $\succ$ braucht nicht als Relation über $T$ noethersch zusein, sondern man muß nur Terme in Betracht ziehen, die in der Reduktionsrelationstehen. Deshalb folgende

**Definition 5.4**

Für ein gegebenes Termersetzungssystem $(T,R)$ heißt eine Ordnung $\succ$über $T$ *$R$–noethersch* (*reduktions-noethersch*) genau dann, wenn $\succ \cap \xrightarrow[R]{}^{*}$ noethersch ist.

Ein Kriterium für die $R$–Noetherizität liefert folgender

**Satz 5.14**

*Eine Ordnung $\succ$ über $T$ ist $R$–noethersch, wenn sie die irreflexivehomöomorphe Einbettungsrelation $\rhd$ über $T$ umfaßt:*

$$\rhd \subseteq \succ \quad .$$

**Beweis**

Sei $\rhd \subseteq \succ$, d.h. $\lhd \subseteq \prec$ bzw. $\unrhd \subseteq \preceq$. Wäre nun $\succ$ nicht $R$–noethersch, dann gäbe es eine unendliche Folge von Termen aus $T$ mit

$$t_0 \succ t_1 \succ t_2 \succ \dots$$

und

$$t_0 \xrightarrow[R]{}^{*} t_1 \xrightarrow[R]{}^{*} t_2 \xrightarrow[R]{}^{*} \dots$$

Nach Selbsteinbettungssatz (Satz 5.5 auf Seite 47)gibt es dann in dieserunendlichen Folge zwei Terme $t_i, t_j$ mit $i < j$, für die

$$t_i \unlhd t_j .$$

ist. Nach Voraussetzung folgt dann $t_i \preceq t_j$, was im Widerspruch zu $t_i \succ t_j$ (gilt wegen obiger Kette und der Transitivität von $\succ$) steht. q.e.d.

Um den Satz 5.13 von Manna / Ness anzuwenden, sind also Ordnungen $\succ$ mit $\rhd \subseteq \succ$ zweckmäßig.

**Definition 5.5**

Eine Ordnung $\succ$ über $T$ heißt *Simplifikationsordnung* , wenn gilt:

(1) $C[t] \ \succ \ t$ (*Teiltermeigenschaft*)

und

(2) $t \succ u$ impliziert $C[t] \succ C[u]$ (*Invarianz, Monotonie*)

für alle $t, u \in T$ und alle echten Kontexte $C[\,]$ aus $T^L \setminus X^L$.

Damit ergibt sich der

**Satz 5.15**

*Eine Ordnung $\succ$ über $T$ ist genau dann eine Simplifikationsordnung, wenn sie eine invariante Erweiterung der homöomorphen Einbettungsrelation $\rhd$ ist:*

$$\rhd \ \subseteq \succ \ .$$

**Beweis**

Nach Satz 5.12 ist

$$\unlhd = (\mathrm{kom}(\sqsubseteq))^+ \ ,$$

also ist $\lhd = (\mathrm{kom}(\sqsubset))^+$ bzw. $\rhd = (\mathrm{kom}(\sqsupset))^+$.

($\rightarrow$) Sei nun $\succ$ invariante Ordnung mit $\rhd \subseteq \succ$. Wegen $\rhd = (\mathrm{kom}(\sqsupset))^+$ und Hülleneigenschaft folgt $\sqsupset \subseteq \succ$ (1). Eigenschaft (2) ist vorausgesetzt, also ist $\succ$ Simplifikationsordnung.

($\leftarrow$) $\succ$ sei Simplifikationsordnung. Dann gilt wegen (1) $\sqsupset \subseteq \succ$. Wegen (2) und den Hülleneigenschaft folgt damit

$$\mathrm{inv}(\sqsupseteq) \ \subseteq \ \mathrm{inv}(\succ) \ = \succ$$

und wegen der Transitivität der Ordnung

$$(\mathrm{inv}(\sqsupset))^+ \ \subseteq \succ^+ = \succ$$

und damit nach Lemma 1.12 auf Seite 11

$$\mathrm{kom}(\sqsupset) \ \subseteq \ (\mathrm{inv}(\sqsupset))^+$$

und weiter

$$(\mathrm{kom}(\sqsupset))^+ \ \subseteq \ (\mathrm{inv}(\sqsupset))^+ \ \subseteq \succ \ ,$$

also

$$\rhd \ \subseteq \succ \ .$$

q.e.d.

Aus den vorangegngenen beiden Sätzen ergibt sich als Folgerung:

**Satz 5.16** (Dershowitz 1982)

*Jede Simplifikationsordnung ist reduktionsnoethersch.*

Weiter gilt (in gewissem Maße umgekehrt):

**Satz 5.17**

*Jede* totale *invariante Ordnung* $\succ$ *über* $T$*, die noethersch ist, ist eineSimplifikationsordnung.*

**Beweis**

Die Invarianz ist vorausgesetzt, also ist nur die Teiltermeigenschaft (1) zu zeigen.
Sei (1) nicht erfüllt, d.h. es existiert ein Term $t \in T$ und ein Kontext$C[\ ] \in T^L \setminus X^L$ mit $C[t] \prec t$ (wegen der Totalität anders nicht möglich), also $t \succ C[t]$. Wegen der Invarianz (2) gilt dann

$$C[t] \succ C[C[t]] \quad \text{usw.},$$

d.h.

$$t \succ C[t] \succ C[C[t]] \succ \ldots$$

im Widerspruch zu $\succ$ noethersch. q.e.d.

**Bemerkung**

Leider aber genügen totale, invariante Ordnungen *nicht* für alle Terminationsbeweise entsprechend dem Satz von Manna/Ness:

*Gegenbeispiel:*
$R = \{f(a) \to f(b),\ g(b) \to g(a)\}$, $(a, b, f, g \in \Sigma$ – keine Variablen).
$R$ ist dann trivialerweise terminierend.
Wenn die Ordnung $\succ$ total ist, so gilt entweder $a \succ b$ oder $b \succ a$. Aus $a \succ b$ folgt wegen der Invarianz $g(a) \succ g(b)$ im Widerspruch zur 2. Regel, $b \succ a$ liefert einen Widerspruch zur 1.Regel.

## 5.3 Eine allgemeine Methode für Terminationsbeweise

Ausgangspunkt ist der Satz von Manna/Ness:
Die Termination von $(T, R)$ ist gesichert, falls bekannt:

$\succ$ – $R$–noethersche Ordnung über $T$ mit:

$t \xrightarrow[R]{} u$ impliziert $t \succ u$.

**Beispiel 20**

„Holländisches Nationalflaggen–Spiel“ (vgl. dazu Beispiel 9 auf Seite 22)
Betrachte als $\succ$ die lexikographische Ordnung der Farbenfolgen (eine lexikographische Ordnung ist stets noethersch!) mit *blau* $\succ$ *weiß* $\succ$ *rot*. Damit ist die Termination bewiesen.

Durch „Umformulierung“ erhält man den leichter zu handhabenden

**Satz 5.18** (Kamin / Lévy 1980)

*Ein Termersetzungssystem* $(T, R)$ *ist terminierend genau dann, wenn eine noetherscheOrdnung* $\succ$ *über* $T$ *existiert mit:*

1. $\varphi(l) \succ \varphi(r)$ *für jede Regel* $l \to r \in R$ *und jede Einsetzung* $\varphi : X \mapsto T$
2. $t \underset{R}{\longrightarrow} u$ *und* $t \succ u$ *implizieren* $C[t] \succ C[u]$ *für alle* $t, u \in T$ *und alle Kontexte* $C[\ ] \in T^L$.[1]

**Beweis**

1. Falls eine solche Ordnung $\succ$ existiert, so folgt aus $t \to_R u$ (vgl. die Definition von $\underset{R}{\longrightarrow}$) nach Voraussetzung sofort $t \succ u$ und damit nach Satz von Manna / Ness die Termination.
2. Sei $(T, R)$ terminierend. Dann existiert nach dem Satz von Manna / Ness eine noethersche Ordnung $\succ$ über $T$ mit $t \underset{R}{\longrightarrow} u$ impliziert $t \succ u$.
   $t \underset{R}{\longrightarrow} u$ gilt aber genau dann, wenn $l \to r \in R, C[\ ] \in T^L, \varphi : X \mapsto T$ existieren mit $t = C[\varphi(l)], u = C[\varphi(r)]$.
   Wenn $C[\ ]$ der leere Kontext (der passenden Sorte) ist, ergibt sich aus der Forderung $t \succ u$ die obige Bedingung 1.
   Sei nun $t \underset{R}{\longrightarrow} u$, so nach Definition von $\underset{R}{\longrightarrow}$ auch $C[t] \underset{R}{\longrightarrow} C[u]$ ($\underset{R}{\longrightarrow}$ ist invariant abgeschlossen), was nach Voraussetzung $C[t] \succ C[u]$ nach sich zieht. Damit ist auch die 2. Bedingung richtig.

q.e.d.

**Beispiel 21**

$$f(f(x)) \to f(g(f(x)))$$

Man definiere eine Ordnung $\succ$ über den Termen mit Hilfe der „direkt benachbarten" $f$.Damit ist die 1. Bedingung erfüllt. Allerdings ist diese Ordnung *nicht* invariant, denn z.B. ist

$$g(f(f(a))) \succ f(a),$$

aber

$$f(g(f(f(a)))) \not\succ f(f(a)).$$

Trotzdem ist der Satz anwendbar, da die 2. Bedingung erfüllt ist, weil

$$g(f(f(a))) \underset{R}{\not\longrightarrow} f(a)\ .$$

Das 2. Kriterium im obigen Satz ist immer noch relativ schwer zu überprüfen, da unendlich viele Ableitungen $t \underset{R}{\longrightarrow} u$ zu überblicken sind, aber mit Blick auf die bereits bekannten zweckmäßigen Eigenschaften von Ordnungsrelationen kann man dies erneut leicht umformulieren. Man kommt auf diese Weise zu dem folgenden

**Satz 5.19 (Lankford 1977, lokale Methode)**

*Ein Termersetzungssystem* $(T, R)$ *ist terminierend genau dann, wenn eine invariante noethersche Ordnung* $\succ$ *über* $T$ *existiert mit*

$$\varphi(l) \ \succ \ \varphi(r)$$

[1] vgl. die Eigenschaft „invariant": diese Bedingung ist weniger als „invariant"; die Idee ist dieselbe wie beim Begriff „$R$–noethersch".

*für jede Regel $l \to r \in R$ und jede Einsetzung $\varphi : X \mapsto T$.*

**Beweis**

1. Falls $\succ$ wie im Satz existiert, so ist die 1. Bedingung des Satzes von KAMIN / LÉVY laut Voraussetzung erfüllt. Die 2. Bedingung ergibt sich aus der Invarianz. Damit ist die Termination gesichert.

2. Sei $(T, R)$ terminierend. Dann ist $\underset{R}{\longrightarrow}^{+}$ noethersche Ordnung (vergleiche den Beweis Teil (1) des Satzes von MANNA/NESS). Nach Definition von $\underset{R}{\longrightarrow}$ ist $\underset{R}{\longrightarrow}$ invariant und damit auch $\underset{R}{\longrightarrow}^{+}$, also ist $\overset{+}{\underset{R}{\longrightarrow}}$ invariante noethersche Ordnung über $T$.

q.e.d.

**Folgerung 5.20**

*Wenn für das Termersetzungssystem $(T, R)$ eine stabile, invariantereduktionsnoethersche Ordnung $\succ$ über $T$ existiert, für die jeweils*

$$l \succ r$$

*für jede Regel $l \to r \in R$ gilt, so ist $(T, R)$ terminierend.*

**Bemerkung**

In dieser Form wird der Satz von LANKFORD am häufigsten angewandt, wobei die meisten dabei verwendeten Ordnungen Simplifikationsordnungen (diese sind reduktionsnoethersch!) sind.

**Beispiel 22**

$$f(g(x)) \;\to\; g(f(x))$$

dazu betrachte Ordnung $\succ$ mit

$$\begin{array}{lllll} s \succ t & \text{genau dann, wenn} & & l(s) > l(t) & \\ & & \text{oder} & l(s) = l(t) \quad \text{und} & \mathrm{HVF}(s) \overset{\circ}{\succ} \mathrm{HVF}(t) \\ & & & \qquad\qquad \text{oder} & \mathrm{HVF}(s) = \mathrm{HVF}(t) \wedge \arg(s) \succ \arg(t). \end{array}$$

Dabei sind:

- HVF der Hauptverknüpfungsfunktor,
- arg die Argument–, bzw. Operandenliste,
- $\overset{\circ}{\succ}$ eine gegebene Ordnung der Operatoren (Funktoren), hier $f \overset{\circ}{\succ} g$.

Es ist relativ leicht zu überprüfen, daß damit $\succ$ stabil, invariant und noethersch ist. Daraus folgt die Termination für das Beispiel.

## 5.4 Die rekursive Pfade–Ordnung

Die bekanntesten und gegenwärtig wohl gebräuchlichsten Simplifikationsordnungen sind:

- the Path of Subterm Ordering (PSO), von PLAISTED 1978 definiert,
- the Recursive Path Ordering (RPO), DERSHOWITZ 1979,
- the Recursive Decomposition Ordering (RDO), JOUANNAUD et al. 1982,
- the Path Ordering (KNS), KAPUR et al. 1985

und Modifikationen von diesen.

Alle diese vier aufgeführten Typen von Termordnungen basieren auf einer vorzugebenden Halbordnung der Operatoren, genannt *precedence* (Vorrang, Stützordnung) . Sie ziehen zum Vergleich der Terme sogenannte „Multimengen“ von Teiltermen heran, die (rekursiv) wieder verglichen werden müssen. Sie werden deshalb manchmal auch als *Multimengenordnungen* bezeichnet.

Zum Verständnis dieser Ordnungen ist der Begriff der „Multimenge“ und die Übertragung einer gegebenen Ordnung über $E$ auf eine Ordnung von Multimengen über $E$ erforderlich.

**Definition 5.6**

Sei $E$ eine endliche Menge. Eine *Multimenge* $M$ über $E$ ist eine eindeutige Abbildung

$$M \; : \; E \; \mapsto \; \mathbb{N}.$$

Für $x \in E$ heißt $M(x)$ die *Zahl der Vorkommen* von $x$ in $M$.
Die Menge aller Multimengen über $E$ bezeichnen wir mit $\mathcal{M}(E)$.

**Schreibweise:**
Z.B.:

$$E \; = \; \{1,2,3,4\} \qquad M \; = \; \{\!\![2,3,2,1,1,2]\!\!\} \; .$$

Hier ist

$$M(1) = 2, \; M(2) = 3, \; M(3) = 1, \; M(4) = 0 \; .$$

**Definition 5.7**

Sei $\prec$ eine Ordnung über $E$, dann bezeichne $\prec\!\!\prec$ die wie folgt festgelegte Ordnung über $\mathcal{M}(E)$:

$$M_1 \prec\!\!\prec M_2$$

genau dann, wenn $\quad M_1 \neq M_2$ und
$\forall x (x \in E \to (M_1(x) > M_2(x) \to \exists y (y \in E \land M_1(y) < M_2(y) \land x \prec y))).$

Mit anderen Worten:
Für jedes Element (mit entsprechender Vielfachheit zählen) aus $M_1$ muß es in $M_2$ ein dominierendes Element (bzgl. der Häufigkeit bzw. bzgl. $\prec$) geben. Für diejenigen Elemente, die in $M_1$ weniger oft als in $M_2$ vorkommen, sind das sie selbst. Interessant ist also nur der Fall (siehe Definition), wo ein Element in $M_1$ öfter als in $M_2$ vorkommt. Dafür muß es ein größeres Element in $M_2$ geben, das nach dem Vergleich mit seinesgleichen noch übrig ist (also in $M_2$ häufiger als in $M_1$ vorkommt).

**Also:** Zum Vergleich erst einmal in $M_1$ und $M_2$ alle Elemente so oft streichen, wie sie sowohl in $M_1$ als auch in $M_2$ vorkommen (Streichen des verallgemeinerten Durchschnittes), Rest mit $\prec$ vergleichen.

**Beispiel 23**

Sei $E \subset I\!N$ und $\prec$ die Kleiner–Relation $<$ in den natürlichen Zahlen $I\!N$.

1. $\emptyset \ll \{\![2,2,2]\!\}$
2. $\{\![3,2,2,1,1,1,4]\!\} \ll \{\![3,3,4]\!\}$, denn $\{\![2,2,1,1,1]\!\} \ll \{\![3]\!\}$.
3. $\{\![3,4]\!\} \ll \{\![3,3,4]\!\}$
4. $\{\![3,3,3,3,2,2]\!\} \ll \{\![3,3,4,0]\!\}$

**Definition 5.8 (Rekursive Pfade–Ordnung** [DERSHOWITZ 1982]**)**

Es sei $<$ eine Ordnung über der Menge der Operatoren von $\Sigma$, die sogenannte *Stützordnung (precedence)* gegeben. Die von ihr induzierte rekursive Pfade–Ordnung $<_{\text{rpo}}$ ist dann rekursiv wie folgt definiert:
Seien $s, t \in T \subseteq T_\Sigma(X)$. Dann ist

$$s <_{\text{rpo}} t$$

genau dann, wenn:

a) $s \neq t$ und $s \in \text{var}(t)$ oder $s, t \in \Sigma$ mit $s < t$ (Konstanten!)

oder

b) $s = \sigma s_1 \dots s_m$ und $t = \omega t_1 \dots t_n$, $m, n \geq 1$, und

(rpo1): $\sigma = \omega \quad \wedge \quad \{\![s_1, s_2, \dots, s_m]\!\} \ll_{\text{rpo}} \{\![t_1, \dots, t_n]\!\}$
oder
(rpo2): $\sigma < \omega \quad \wedge \quad \forall i (1 \leq i \leq m \;\rightarrow\; s_i <_{\text{rpo}} t)$
oder
(rpo3): $\sigma \not\leq \omega \quad \wedge \quad \exists j (1 \leq j \leq n \quad \wedge \quad s \leq_{\text{rpo}} t_j)$ ($\sigma > \omega$ oder $\sigma$ und $\omega$ unvergleichbar)

**Beispiel 24**

1. $q \in X, \quad \vee < \wedge < \neg$.
   Es ist
   $$\neg((\neg q \wedge q) \vee (\neg q \wedge \neg q)) \quad <_{\text{rpo}} \quad \neg(\neg q \wedge (q \vee \neg q)),$$
   Denn:
   wegen (rpo1) ist zu prüfen:
   $$\{\![(\neg q \wedge q) \vee (\neg q \wedge \neg q)]\!\} \quad \ll_{\text{rpo}} \quad \{\![\neg q \wedge (q \vee \neg q)]\!\} \text{ ?}$$
   $$\text{äquivalent zu } (\neg q \wedge q) \vee (\neg q \wedge \neg q) \quad <_{\text{rpo}} \quad \neg q \wedge (q \vee \neg q) \text{ ?}$$
   wegen (rpo2) und $\vee < \wedge$ ist nun zu prüfen:
   $$\neg q \wedge q \quad <_{\text{rpo}} \quad \neg q \wedge (q \vee \neg q) \text{ ?} \quad \textbf{und} \quad \neg q \wedge \neg q \quad <_{\text{rpo}} \quad \neg q \wedge (q \vee \neg q) \text{ ?}$$

| | |
|---|---|
| wegen (rpo1) ist zu prüfen: | wegen (rpo1) zu prüfen: |
| $\{\!\![\neg q \; , \; q]\!\!\} \quad \ll_{\text{rpo}} \quad \{\!\![\neg q \; , \; q \vee \neg q]\!\!\}$ ? | $\{\!\![\neg q \; , \; \neg q]\!\!\} \quad \ll_{\text{rpo}} \quad \{\!\![\neg q \; , \; q \vee \neg q]\!\!\}$ ? |
| gilt, weil | äquivalent zu |
| $q \quad <_{\text{rpo}} \quad q \vee \neg q$ | $\neg q \quad <_{\text{rpo}} \quad q \vee \neg q$ ? |
| wegen a) | wegen (rpo3) ist zu prüfen: |
| | $\neg q \leq_{\text{rpo}} q$ oder $\neg q \leq_{\text{rpo}} \neg q$ ? |
| | gilt wegen $\neg q \leq_{\text{rpo}} \neg q$ |

2. Vergleiche das Beispiel 11 eines Termersetzungssystems mit fünf Gleichungen eines Booleschen Verbandes auf Seite 23. Mit Hilfe der RPO ist die Termination diese Termersetzungssystems leicht nachzuweisen.
   Wir empfehlen dies als **Übungsaufgabe** durchzuführen.

3. Nicht immer lassen sich zwei Terme mit der RPO vergleichen. Betrachte zum Beispiel die Terme:

   $$f(g(x)) \quad \text{und} \quad h(k(x))$$

   mit der Stützordnung:

   $$f > k \quad \text{und} \quad g > h.$$

   Man sollte doch annehmen, daß dann der linke Term „gewichtiger“ ist als der rechte, aber in der RPO stimmt dies nicht. Man muß (rpo3) benutzen wegen der Unvergleichbarkeit von $f$ und $h$, aber es ist weder

   $$f(g(x)) \;\leq_{\text{rpo}}\; k(x) \quad \text{noch} \quad h(k(x)) \;\leq_{\text{rpo}}\; g(x) \;.$$

   Es gibt sogar Terme, die sich bei *keiner* Stützordnung (precedence) vergleichen lassen:
   Zum Beispiel:

   $$(\neg p \vee p) \;\wedge\; (\neg q \vee q) \quad \text{und} \quad (p \vee q) \;\wedge\; (p \vee q)$$

   Begründung:
   Zu vergleichen wären nach rpo1):

   $$\{\!\![\; (\neg p \vee p) \;,\; (\neg q \vee q) \;]\!\!\} \quad \text{und} \quad \{\!\![\; (p \vee q) \;,\; (p \vee q) \;]\!\!\} \;.$$

   Dies erfordert auf jeden Fall den Vergleich von

   $$(\neg p \vee p) \text{ oder } (\neg q \vee q) \quad \text{mit} \quad (p \vee q) \;,$$

   was stets auf den Vergleich von $p$ und $q$ hinausläuft, z. B. ist

   $$\{\!\![\; \neg p \;,\; p \;]\!\!\} \quad \text{zu vergleichen mit} \quad \{\!\![\; p \;,\; q \;]\!\!\} \;,$$

   also $\neg p$ mit $q$. $p, q$ als Variablen sind aber unvergleichbar, ebenso ist $q \notin \text{var}(\neg p)$, also sind die Terme unvergleichbar.

Es gilt der hier aus Zeitgründen nur noch angeführte, aber nicht mehr bewiesene

**Satz 5.21**

*Für jede beliebige Ordnung $>$ über der Menge der Operatoren ist die von ihr induzierte RPO $>_{\text{rpo}}$ eine Simplifikationsordnung.*

# Index


$\mathcal{A}$–Belegung, 2
ableitbar, 9
Ableitung, 14
Abschluß
  invarianter, 8, 9
  kompatibler, 6
  kontextueller, 8, 9
  stabiler, 8
  transitiver, 6
abstrakte Datentypen, 41
Adresse, 3
äquivalent
  Reduktionssysteme, 34
Äquivalenz
  syntaktische, 9
Algebra, 1
Algorithmus
  KNUTH–BENDIX–, 34
allgemeinster Unifikator, 29
Arität, 1

Belegung, 2

CHURCH–ROSSER–Eigenschaft, 17
CP/C–Prozedur, 34
CP($R$), 29
CP($r, R$), 34

Einbettung
  homöomorphe, 46
Einsetzung, 2
erfüllt, 5

Gleichung, 5
  ableitbare, 9
Gleichungssystem, 5
Grundterm, 1
Gruppenaxiome, 37, 41
gültig, 5
Gültigkeitsproblem, 15

Halbordnung, 43
homöomorphe Einbettung, 46
Homomorphismus, 2
HUET–Theorem, 30

induktionslose Induktion, 42
instantiation, 3
Instanz, 3
invariant, 8
Invarianz, 51
irreduzibel, 19

kanonisch, 23
KB–Prozedur, 34
KNUTH–BENDIX–Algorithmus, 34
KNUTH–BENDIX–Theorem, 33
kompatibel, 6
kompatibler Abschluß, 6
konfluent, 18
  lokal –, 25
Konfluenz, 19
Kongruenzrelation, 6
Konkretisierung, 3
Konstante, 1
Kontext, 4, 5
  leerer, 5
kontextueller Abschluß, 8
Kontraktum, 14
konvergent, 23
Korrektheitssatz, 35
kritische Situation, 27, 28
kritisches Paar, 29, 37
KRUSKAL's Baum–Theorem, 47

leerer Kontext, 5
linke Seite, 13
lokal konfluent, 25
Lücke, 4
Lückensorte, 5

$\mathcal{M}(E)$, 55
mgu, 29
Modell, 5
monoton, 8
Monotonie, 51
most general unifier, 29
Multimenge, 55
Multimengenordnung, 55

neuer Operator, 36
NEWMAN–Lemma, 25
NF, 20
$NF(t, R)$, 34
Nichttermination, 45
noethersch, 20, 43

reduktions–, 50
Normalform, 19
Normalisierung, 36

operationale Semantik, 41
Operator, 1
neuer, 36
Ordnung, 43
Halb–, 43
Multimengen–, 55
partielle, 43
reflexive, 43
rekursive Pfade–, 55, 56
Simplifikations–, 51
Stütz–, 55, 56
totale, 43
Orientierungswahl, 35

partielle Ordnung, 43
Platzhaltervariable, 4
Position, 4
precedence, 55, 56
PSO, 55

RDO, 55
rechte Seite, 13
recursive path ordering, 55
Redex, 14
Redukt, 14
Reduktionsfolge, 14
reduktionsnoethersch, 50
Reduktionsrelation, 14
Reduktionssystem, 13
Regel, 13
Instanz der, 14
linke Seite der, 13
rechte Seite der, 13
Regelsystem, 13
rekursive Pfade–Ordnung, 55, 56
$R$–noethersch, 50
RPO, 55, 56

Schleife, 45
schwach normalisierend, 20
selbsteinbettend, 47
Semantik
operationale, 41
$\Sigma$–Algebra, 1
$(\Sigma, E)$–Algebra, 5
$\Sigma$–Reduktionssystem, 13
$\Sigma$–Regelsystem, 13
$\Sigma$–Termalgebra, 2
Signatur, 1
Simplifikationsordnung, 51
Sorte, 1, 3
Stützordnung, 55, 56
stabil, 8
Standardterm, 2
Standardtermalgebra, 2
Stelle, 3
Streichen
von Regeln, 36
Substitution, 2
Superposition, 29, 37

Teilterm, 3
Teiltermeigenschaft, 51
Teiltermrelation, 48
echte, 48
Term, 3
mit Variablen, 1
Standard–, 2
variablenfreier, 1
Termalgebra, 2
Termersetzungssystem, 13
Termination, 37
Terminationskriterien, 49
Terminationstest, 37
terminierend, 20
totale Ordnung, 43

überlappend, 28
Unifikator
allgemeinster, 29

Varietät, 5
Vervollständigung, 34
der Gruppenaxiome, 37
vollinvariant, 8
vollständig, 23
Vorkommen, 4
Zahl der, 55

Wortproblem, 15, 34, 41

Zahl der Vorkommen, 55
Zyklus, 45