Inferenzmethoden

Einheit 16

Termersetzungssysteme

1. Motivation und Grundbegriffe
2. Knuth-Bendix Vervollständigung
3. Narrowing
4. Unifikation durch Transformation

Effiziente Verarbeitung von Gleichheiten

- **Mechanismus zur Lösung des Wortproblems**
  - Wortproblem: Sind s und t gleich aufgrund der Axiome einer Theorie?
  - Methode: Vereinfachung der Ausdrücke bis textliche Identität erreicht
  - Hilfsmittel: Zielgerichtete Anwendung einer Menge von Gleichungen
- **Gleichheiten als Vereinfachung betrachtet**
  - Gleichheiten $a = b$ erhalten Richtung $a{\rightarrow}b$
  - Reduktion: Ersetzung von Teiltermen durch einfachere gleiche Terme
- **Eigenständiges Forschungsgebiet Rewriting**
  - Eigenschaften von Systemen syntaktischer Transformationsregeln
  - Verwendet eigene, z.T. abweichende Notationen
- **Vielfältige Anwendungen**
  - Integration in Theorembeweiser durch erweiterte Unifikationsalgorithmen
  - Mögliche Methode zur Implementierung von Theoriekonnektionen
  - Schlüssel für Unifikationstheorie, Logiksysteme, Berechnungsmodelle

Termersetzung am Beispiel der Gruppentheorie

- **Basisaxiome einer Gruppe mit Verknüpfung ·**

$$\begin{array}{rcl} e\cdot x \doteq x & & \textit{linksseitiges Einselement} \\ x\cdot e \doteq x & & \textit{rechtsseitiges Einselement} \\ \overline{y}\cdot y \doteq e & & \textit{Linksinverses} \\ (u\cdot v)\cdot w \doteq u\cdot(v\cdot w) & & \textit{Assoziativität} \end{array}$$

- **Als Reduktionsregeln**

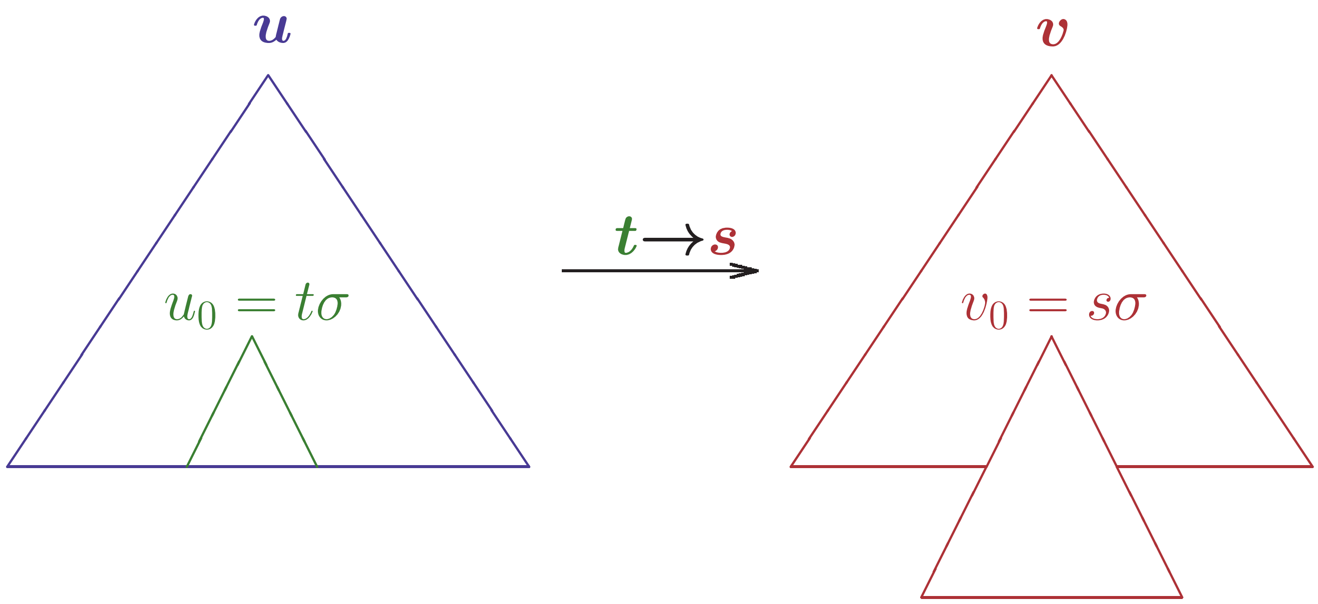
$$\begin{array}{lr} \mathbf{r_1} & e\cdot x \to x \\ \mathbf{r_2} & x\cdot e \to x \\ \mathbf{r_3} & \overline{y}\cdot y \to e \\ \mathbf{r_4} & (u\cdot v)\cdot w \to u\cdot(v\cdot w) \end{array}$$

- **Anwendung von Regeln zur Beweisführung**

$$\begin{array}{ll} & (\overline{a}\cdot a)\cdot b \doteq (e\cdot\overline{a})\cdot(a\cdot b) \\ \xrightarrow{r_1} & (\overline{a}\cdot a)\cdot b \doteq \overline{a}\cdot(a\cdot b) \\ \xrightarrow{r_4} & \overline{a}\cdot(a\cdot b) \doteq \overline{a}\cdot(a\cdot b) \end{array}$$

Rewriting: Grundbegriffe

- **Termersetzungssystem $(\mathcal{A}, \mathcal{R})$**
  – $\mathcal{A}$ Alphabet, $\mathcal{R}$ Menge von Reduktionsregeln über $\mathcal{A}^*$
- **Reduktionsregel $t \longrightarrow s$** (t, s Terme über $\mathcal{A}$)
  – t (Redex) keine Variable
  – Alle Variablen von s (Kontraktum) kommen in t vor
- **$u \xrightarrow{r} v$: Regelanwendung von $r = t \longrightarrow s$ auf u**
  – Teiltermmatching: Bestimme Substitution σ, so daß $t\sigma$ Teilterm von u
  – Ersetzung: Ersetze $t\sigma$ durch $s\sigma$



Rewriting: Grundbegriffe II

- **$u \xrightarrow{*} w$: Iterierte Anwendung von Regeln**
  – $u \xrightarrow{*} u$ (ohne Anwendung von Regeln)
  – $u \xrightarrow{*} w$, falls es ein $v \in \mathcal{A}^*$ gibt mit $u \xrightarrow{r} v$ und $v \xrightarrow{*} w$
- **Normalform: nichtreduzierbarer Term**
  – Term v, der nicht durch Regelanwendungen "reduziert" werden kann
  – **Wert von u**: Normalform v mit $u \xrightarrow{*} v$
- **Wichtige Anforderungen an Termersetzungssysteme**
  – Ist die Normalform eines Terms eindeutig? $\mapsto$ Konfluenz
  – Hat jeder Term eine Normalform? $\mapsto$ Normalisierbarkeit
  – Wenn ja, führt jede Anwendung von Regeln dorthin? $\mapsto$ Starke Normalisierbarkeit

Konfluenz

Ist die Normalform eines Terms eindeutig?

- **$(\overline{a} \cdot a) \cdot b$ hat zwei Normalformen**
  - $(\overline{a} \cdot a) \cdot b \xrightarrow{r_3} e \cdot b \xrightarrow{r_1} b$
  - $(\overline{a} \cdot a) \cdot b \xrightarrow{r_4} \overline{a} \cdot (a \cdot b)$
  - Die Normalformen sind nicht weiter reduzierbar, aber verschieden

- **Konfluenz von $(\mathcal{A}, \mathcal{R})$**
  - Ketten von Regelanwendungen sind immer zusammenführbar

    Aus $\boldsymbol{t{\uparrow}s}$ ($u \xrightarrow{*} t$ und $u \xrightarrow{*} s$ für ein u)

    folgt $\boldsymbol{t{\downarrow}s}$ ($t \xrightarrow{*} v$ und $s \xrightarrow{*} v$ für ein v)

$$\begin{tikzcd} t{\uparrow}s & v \arrow[dl, "\star"'] \arrow[dr, "\star"] & \\ s \arrow[dr, "\star"'] & & t \arrow[dl, "\star"] \\ t{\downarrow}s & u & \end{tikzcd}$$

  - Regeln entsprechen “echten” Gleichungen

    Gleiche Beweiskraft wie Anwendung der Gleichheiten

Das einfache Regelsystem für die Gruppentheorie ist nicht konfluent

Normalisierbarkeit

Hat jeder Term eine Normalform?

- **Regeln für Gruppentheorie sind unvollständig**
  - Gleichung der Assoziativität gilt in zwei Richtungen
  - Naive Lösung: Zusatzregel für Rückwärtsanwendung der Assoziativität

$$
\begin{array}{ll}
\mathbf{r_1} & e \cdot x \to x \\
\mathbf{r_2} & x \cdot e \to x \\
\mathbf{r_3} & \overline{y} \cdot y \to e \\
\mathbf{r_4} & (u \cdot v) \cdot w \to u \cdot (v \cdot w) \\
\mathbf{r_5} & u \cdot (v \cdot w) \to (u \cdot v) \cdot w
\end{array}
$$

  - Reduktionskette $(\overline{a} \cdot a) \cdot b \xrightarrow{r_4} \overline{a} \cdot (a \cdot b) \xrightarrow{r_5} (\overline{a} \cdot a) \cdot b \xrightarrow{r_4}$ .. terminiert nicht
  - Normalform existiert dennoch: $(\overline{a} \cdot a) \cdot b \xrightarrow{r_3} e \cdot b \xrightarrow{r_1} \boldsymbol{b}$
- **Schwache Normalisierbarkeit**
  - Jeder Term besitzt eine Normalform
  - Der λ-Kalkül ist nicht schwach normalisierbar (z.B. $(\lambda x.x\,x)(\lambda x.x\,x)$)
- **Starke Normalisierbarkeit ($(\mathcal{A}, \mathcal{R})$ noethersch)**
  - Jede Kette von Regelanwendungen terminiert

Nachweis starker Normalisierbarkeit

Das System $r_1..r_4$ ist stark normalisierbar

- **Definiere wohlfundierte $\succ$ Ordnung auf $\mathcal{A}^*$**
  - z.B. lexikographische Ordnung: Länge und Ordnung im Alphabet
  - Sinnvolle Ordnung auf $\mathcal{A}$ ist z.B. $(\succ) \succ \cdot \succ z \succ ... \succ a$
- **Zeige, daß jede Regel $t{\longrightarrow}s$ die Ordnung respektiert**
  - $r_1 : \; e \cdot x \succ x$
  - $r_2 : \; x \cdot e \succ x$
  - $r_3 : \; \overline{y} \cdot y \succ e$
  - $r_4 : \; (u \cdot v) \cdot w \succ u \cdot (v \cdot w)$
- **Konsequenz: Jede Reduktionsfolge terminiert**
  - Jede Reduktionsfolge liefert eine bzgl. $\succ$ absteigende Kette von Termen
  - Wohlfundiertheit von $\succ$: es gibt keine unendlichen absteigenden Ketten

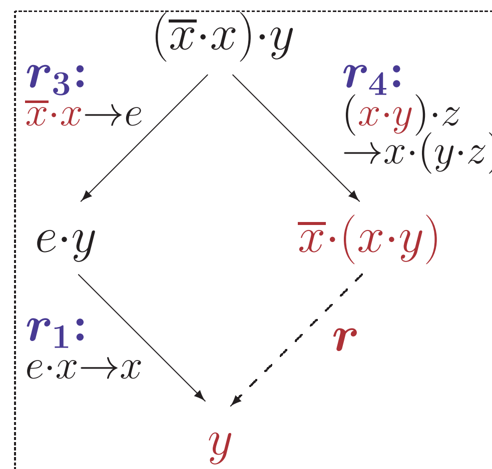
Steuerung von Termersetzungsystemen

- **Vermeide nicht normalisierbare Regelsysteme**
  – Terme ohne Normalform sind für automatisches Schließen ungeeignet
  – λ-Kalkül ist Logik höherer Stufe und hochgradig unentscheidbar
- **Verwende heuristische Steuerung bei schwacher Normalisierbarkeit**
  – Strategische Steuerung vermeidet nichtterminierende Reduktionsketten
- **Konfluenz ist sehr wichtig**
  – Andernfalls wird Backtracking über Reduktionen erforderlich um Gleichheit von Termen nachzuweisen
- **Optimal: vollständige Termersetzungssysteme**
  – $(\mathcal{A}, \mathcal{R})$ noethersch und konfluent

  Der Einsatz vollständiger Termersetzungssysteme führt immer zum Ziel!

Erzeugung vollständiger Regelsysteme

- **Ausgangspunkt: System von Gleichungen**
  - Axiomatische Beschreibung der Bedeutung von Termen einer Theorie
- **Transformiere Gleichungen in gerichtete Regeln**
  - Regel $t \rightarrow s$ ersetzt Gleichung $t = s$
  - Regel ist gerichtet: Anwendung in Gegenrichtung nicht möglich
- **Regeln sollen Gleichungen vollständig ersetzen**
  - Konfluenz kann durch Ausrichtung verloren gehen
  - Zusatzregel für Gegenrichtung würde starke Normalisierbarkeit zerstören
  - Verfahren zur Vervollständigung von Reduktionssystemen nötig
- **Superposition von Regeln r_i, r_j**
  - Unifiziere Teilterme der linken Seiten beider Regeln
  - Bilde kritischen Term t aus instantiierten Teiltermen
  - Bilde Termpaar s, s' mit $t \xrightarrow{r} s$ und $t \xrightarrow{r'} s'$:
  - Normalisiere s, s' in $(\mathcal{A}, \mathcal{R})$ zu kritischem Termpaar u, v
  - Falls $u \neq v$ bilde neue Regel $u \rightarrow v$ (bzw. $v \rightarrow u$, falls $v \succ u$)



Knuth Bendix Vervollständigung

Vervollständigung durch Superposition

- **Ausgangspunkt: noethersches Regelsystem**
  - i.a. einfache Umwandlung eines Gleichungssystems
- **Ziel: vollständiges Regelsystem**
- **Methode: schrittweise Superposition aller Regeln**
  - unter Verwendung einer wohlfundierten Termordnung $\succ$
- **Abbruchkriterium: lokale Konfluenz**
  - Je zwei Regelanwendungen sind zusammenführbar

$$\begin{tikzcd} & v \arrow[dl, "r"'] \arrow[dr, "r'"] & \\ s \arrow[dr, "\star"'] & & t \arrow[dl, "\star"] \\ & u & \end{tikzcd}$$

- **Korrektheit:**
  - Starke Normalisierbarkeit: neue Regeln erhalten die Termordnung $\succ$
  - Konfluenz folgt aus lokaler Konfluenz und starker Normalisierbarkeit

Das Diamond Lemma

$\xrightarrow{r}$ stark normalisierbar, lokal konfluent $\Rightarrow$ $\xrightarrow{r}$ konfluent

- **Für jeden Term v gibt es ein n, so daß jede in v beginnende Reduktionsfolge in maximal n Schritten terminiert**
  - Stark normalisierbar: jede Reduktionsfolge von v terminiert
  - Endliche Verzweigung von $\xrightarrow{r}$: es gibt eine maximale Schrittzahl

- **Für alle v folgt $t{\downarrow}s$ aus $v \xrightarrow{*} t$ und $v \xrightarrow{*} s$**

  Induktion über maximale Länge n aller Reduktionsketten

  $n{=}0$: Es folgt $v{=}t{=}s$, also $t{\downarrow}s$ trivialerweise

  $n{+}1$: Falls $v \xrightarrow{0} t$ oder $v \xrightarrow{0} s$, dann $v{=}t$ oder $v{=}s$

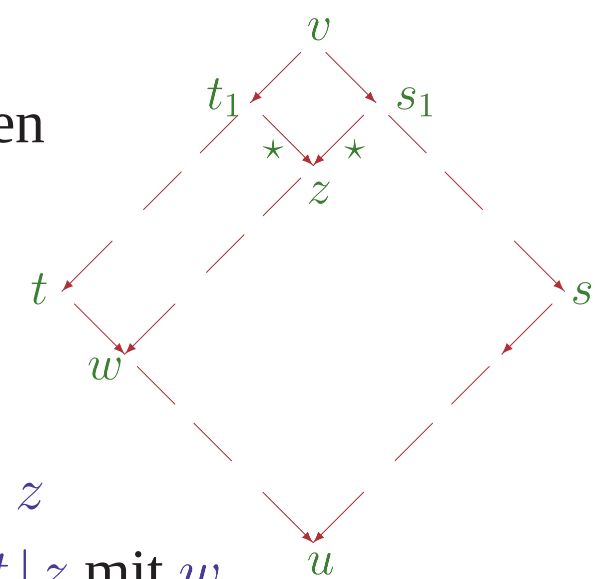
  Ansonsten $v \xrightarrow{r} t_1 \xrightarrow{*} t$ und $v \xrightarrow{r} s_1 \xrightarrow{*} s$

  Lokale Konfluenz: $t_1{\downarrow}s_1$ mit einem Term z

  Induktionsannahme für t_1: $t_1 \xrightarrow{*} t$, $t_1 \xrightarrow{*} z$ also $t{\downarrow}z$ mit w

  Induktionsannahme für s_1: $s_1 \xrightarrow{*} z \xrightarrow{*} w$, $s_1 \xrightarrow{*} s$ also $w{\downarrow}s$ mit u

  Insgesamt $t \xrightarrow{*} w \xrightarrow{*} u$ und $s \xrightarrow{*} u$, also $t{\downarrow}s$



Knuth–Bendix Verfahren

Eingabe: Endliche Menge $\mathcal{G}$ von Axiomgleichungen, Termordnung $\succ$.
Ausgabe: Bei Terminierung vollständiges Termersetzungssystem
oder Fehlermeldung

Initialisiere Regelmenge $\mathcal{R} := \emptyset$
Solange $\mathcal{G}$ nicht leer
Wähle Gleichung aus $\mathcal{G}$ und reduziere sie mit Regeln aus $\mathcal{R}$
Falls reduzierte Gleichung nicht von der Form $x \doteq x$
Dann Falls reduzierte Gleichung läßt sich nicht mit $\succ$ zu neuer Regel richten
Dann Abbruch *'Termordnung nicht ausreichend'*
Sonst Bilde neue Regel und füge sie zu $\mathcal{R}$ hinzu;
Reduziere alle Regeln aus $\mathcal{R}$ untereinander;
Falls eine der reduzierten Regeln die Termordnung $\succ$ verletzt
Dann entferne sie aus $\mathcal{R}$ und
nehme sie in $\mathcal{G}$ auf, sofern sie nicht von der Gestalt $x \doteq x$ ist;
Bilde alle kritischen Paare zwischen der neuen Regel
und den übrigen Regeln und füge sie als Gleichungen zu $\mathcal{G}$ hinzu
Ergebnis $\mathcal{R}$

Knuth Bendix Vervollständigung am Beispiel

- **Vollständiges Regelsystem für die Gruppentheorie**

$\mathbf{r_1}$	$e{\cdot}x \rightarrow x$	
$\mathbf{r_2}$	$x{\cdot}e \rightarrow x$	
$\mathbf{r_3}$	$\overline{x}{\cdot}x \rightarrow e$	
$\mathbf{r_4}$	$(x{\cdot}y){\cdot}z \rightarrow x{\cdot}(y{\cdot}z)$	
$\mathbf{r_5}$	$\overline{e} \rightarrow e$	Superposition von r_2 und r_3
$\mathbf{r_6}$	$\overline{x}{\cdot}(x{\cdot}y) \rightarrow y$	Superposition von r_3 und r_4
$\mathbf{r_7}$	$\overline{\overline{x}} \rightarrow x$	Superposition von r_3 und r_6
$\mathbf{r_8}$	$x{\cdot}\overline{x} \rightarrow e$	Superposition von r_3 und r_7
$\mathbf{r_9}$	$x{\cdot}(\overline{x}{\cdot}y) \rightarrow y$	Superposition von r_6 und r_7
$\mathbf{r_{10}}$	$\overline{(x{\cdot}y)} \rightarrow \overline{y}{\cdot}\overline{x}$	Superposition von r_4, r_6, r_8, r_9 (aufwendig)

Theorieunifikation mit Termersetzung

- **Komplementarität von Theoriekonnektionen**

$$\left[\begin{array}{cc} & R(z\cdot(\overline{c}\cdot c), z\cdot(\overline{z}\cdot b))^F \\ R(c,b)^T & \end{array}\right] \quad \sigma{=}[c/z]$$

Group

  – Konnektion benutzt Theorieunifikation für Gruppentheorie

- **Unifikation von $R(z\cdot(\overline{c}\cdot c), z\cdot(\overline{z}\cdot b))^F$ und $R(c,b)^T$**

$$\begin{array}{ll} & R(z\cdot(\overline{c}\cdot c), z\cdot(\overline{z}\cdot b)) = R(c,b) \\ \xrightarrow{r_3} & R(z\cdot e, z\cdot(\overline{z}\cdot b)) = R(c,b) \\ \xrightarrow{r_9} & R(z\cdot e, b) = R(c,b) \\ \xrightarrow{r_2} & R(z,b) = R(c,b) \\ \xrightarrow{unif} & R(c,b) = R(c,b) \qquad \sigma{=}[c/z] \end{array}$$

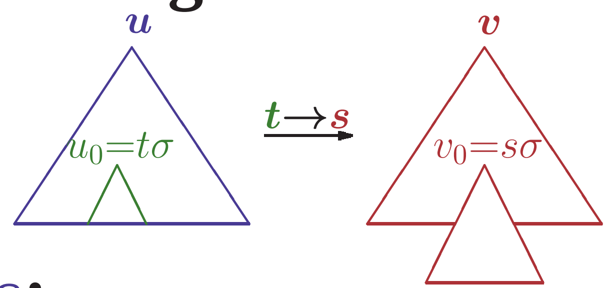
- **Integriere Unifikation und Termersetzung**
  – Bestimme Substitution während der Ersetzung

Verallgemeinerte Anwendung von Rewrite-Regeln

- **Rewriting einer Gleichung $u{=}w$ mit der Regel $t{\rightarrow}s$:**

  $$u{=}w \;\rightarrow\; v{=}w$$

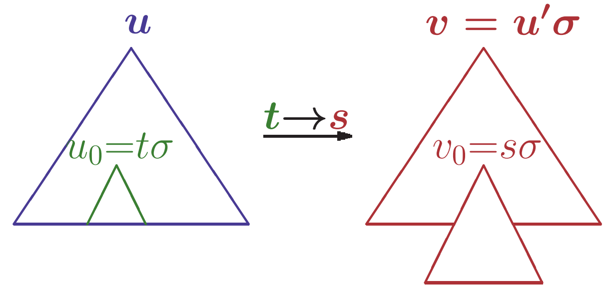
  v entsteht aus u durch Ersetzung
  eines Teilterms $t\sigma$ mit $s\sigma$



- **Einfaches Verengen mit der Regel $t{\rightarrow}s$:**

  $$u{=}w, E \;\rightarrow\; v{=}w, E$$

  v entsteht aus u durch Ersetzung
  eines Teilterms $t\sigma$ mit s und anschließender
  Anwendung von σ auf den ganzen Term



- **Lässiges Verengen mit der Regel $ft_1\ldots t_n{\rightarrow}s$:**

  $$u{=}w,\; E \;\rightarrow\; v{=}w,\; u_1{=}t_1, \ldots, u_n{=}t_n,\; E$$

  v entsteht aus u durch Ersetzung des Auftretens von $fu_1\ldots u_n$ durch s.

  Bestimmung der Substitution wird in Gleichungssystem verschoben

Verengen + Martelli–Montanari–Regeln liefert vollständiges (Unifikations-) Verfahren für Lösung einer Menge von Gleichungen unter einer Theorie

Termersetzungssystem für einfache Arithmetik

Arithmetische Regeln für ganze Zahlen

r₁	$v(n(x)) \to x$	Vorgänger/Nachfolger
r₂	$0 + x \to x$	Null als Linksidentität
r₃	$x + 0 \to x$	Null als Rechtsidentität
r₄	$n(x) + y \to n(x + y)$	Addition links
r₅	$x + n(y) \to n(x + y)$	Addition rechts
r₆	$x + v(y) \to v(x + y)$	Subtraktion

Martelli–Montanari–Regeln

Termdekomposition	$\{f(s_1, ..s_n) \doteq f(t_1, ..t_n)\} \cup E \to \{s_1 \doteq t_1, ..s_n \doteq t_n\} \cup E$	
Ausdünnen	$\{x \doteq x\} \cup E \to E$	
Umstellung	$\{t \doteq x\} \cup E \to \{x \doteq t\} \cup E$	$(t \notin \mathcal{V})$
Variablenelimination	$\{x \doteq t\} \cup E \to \{x \doteq t\} \cup E\{x \backslash t\}$	$(x \notin t,\ x \in E)$

Theorie-Unifikation durch Transformation

Gleichungen	**angewandte Regeln**
$(x+x)+v(0)=n(0)$	
$v(x_1+y_1)=n(0),\ x_1=x+x,\ v(y_1)=v(0)$	lässige Verengung mit r_6
$v(x_1+y_1)=n(0),\ x_1=x+x,\ y_1=0$	Termdekomposition
$v(x_1+0)=n(0),\ x_1=x+x$	Variablenelimination $y_1=0$
$v((x+x)+0)=n(0)$	Variablenelimination $x_1=x+x$
$x_2=n(0),\ n(x_2)=(x+x)+0$	lässige Verengung mit r_1
$x_2=n(0),\ n(x_2)=x_3,\ x_3=x+x,\ 0=0$	lässige Verengung mit r_3
$x_2=n(0),\ n(x_2)=x_3,\ x_3=x+x$	Ausdünnung $0=0$
$x_2=n(0),\ n(x_2)=x+x$	Variablenelimination $x_3=x+x$
$x_2=n(0),\ n(x_2)=n(x_4+y_4),\ x=n(x_4),\ x=y_4$	lässige Verengung mit r_4
$n(n(0))=n(x_4+y_4),\ x=n(x_4),\ x=y_4$	Variablenelimination $x_2=n(0)$
$n(0)=x_4+y_4,\ x=n(x_4),\ x=y_4$	Termdekomposition
$n(0)=x_4+y_4,\ y_4=n(x_4)$	Variablenelimination $x=y_4$
$n(0)=x_4+n(x_4)$	Variablenelimination $y_4=n(x_4)$
$n(0)=n(x_5+y_5),\ x_4=x_5,\ n(x_4)=n(y_5)$	lässige Verengung mit r_5
$n(0)=n(x_5+y_5),\ n(x_5)=n(y_5)$	Variablenelimination $x_4=x_5$
$0=x_5+y_5,\ n(x_5)=n(y_5)$	Termdekomposition
$0=x_5+y_5,\ x_5=y_5$	Termdekomposition
$0=y_5+y_5$	Variablenelimination $x_5=y_5$
$0=x_6,\ y_5=0,\ y_5=x_6$	lässige Verengung mit r_2
$0=x_6,\ 0=x_6$	Variablenelimination $y_5=0$
$0=x_6,\ x_6=0$	Umstellung
$0=0$ $\qquad \boldsymbol{\sigma(x)=n(0)}$	Variablenelimination $x_6=0$

Anwendung von Termersetzung in der Deduktion

- **Unifikationsalgorithmus für Theoriekonnektionen**
  - Nur bei vollständigen Regelsystemen möglich

    z.B. Einfache Arithmetik, Gruppentheorie, Gleichheitstheorien
  - Nicht jede Theorie kann als vollstandiges Regelsystem formuliert werden
  - Heuristische Steuerung für unvollständige Systeme praktisch erfolgreich
- **Deduktion als Termersetzungssystem formulierbar**
  - In beschränkten Bereichen sinnvoll einsetzbar
  - Niedriges Niveau, kein Ersatz für verdichtete logische Verfahren

Effiziente Einbettung in Konnektionsmethode noch offen