Ergebnisse zu $\mathcal{G}$ (3)

Definition 3.8.

Sei $(R_i, E_i, G_i)_{i\in\mathbb{N}}$ eine $\mathcal{G}$-Ableitung. Dann ist

$$R^\infty = \bigcup_{i\geq 0} \bigcap_{j\geq i} R_j \qquad E^\infty = \bigcup_{i\geq 0} \bigcap_{j\geq i} E_j$$

die Menge der persistenten Regeln bzw. Gleichungen.

Die $\mathcal{G}$-Ableitung heißt **fair**, falls

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

Also

Fairness bedeutet, dass im Grenzsystem (R^∞, E^∞) jedes bildbare kritische Paar auch gebildet worden ist.

Beachte

Die Forderung $E^\infty = \emptyset$ aus der KB-Vervollständigung ist hier i. Allg. unerfüllbar.

Satz 3.1.

Für jede faire $\mathcal{G}$-Ableitung gilt

a) $R^\infty(E^\infty)$ ist grundkonfluent und terminierend.
b) Für jede Grundgleichung $u = v$ gilt: $u =_E v$ gdw $u \downarrow v$ in $R_i(E_i)$ für ein i.

Ergebnisse zu $\mathcal{G}$

Zusammenfassung

Redundante Gleichungen

- kann man zum Simplifizieren benutzen,
- ohne sie für die Bildung kritischer Paare zu berücksichtigen.

Dies

- erhält die Simplifikationskraft und
- schränkt gleichzeitig den Suchraum ein.

Frage

Wie erkennt man redundante Gleichungen?

Ein Test auf Redundanz

Ansatz

Gilt $\sigma(s) \downarrow \sigma(t)$ in $R(E)$ für jede Grundsubstitution σ, so ist $s = t$ in (R, E) redundant.

Dies reicht nicht ganz! Es reicht nur für das Löschen von $s = t$, nicht aber für das Simplifizieren mit $s = t$: Der erste Beweisschritt in $\sigma(s) \downarrow \sigma(t)$ darf nicht zu groß sein.

Bezeichnung

$u \xrightarrow{s\rhd}_{R(E)} v \quad \text{gdw}$

- $u \longrightarrow_R v$ mit $l \to r \in R$ und $s \rhd l$ oder
- $u \longrightarrow_{E^\succ} v$ mit $l = r \in E$ und $s \rhd l$.

Definition 3.9.

$s \Downarrow_\rhd t$ in $R(E)$ gdw für jede Grundsubstitution σ gilt:

$\sigma(s) \equiv \sigma(t)$ oder

$\sigma(s) \succ \sigma(t)$ und $\sigma(s) \xrightarrow{s\rhd}_{R(E)} s_1 \downarrow \sigma(t)$ oder

$\sigma(t) \succ \sigma(s)$ und $\sigma(t) \xrightarrow{t\rhd}_{R(E)} t_1 \downarrow \sigma(s)$.

Lemma 3.6

Gilt $s \Downarrow_\rhd t$ in $R(E)$, so gilt auch $s = t \succ_{\mathcal{P}} R(E)$.

Redundanz bei AC-Gleichungen

Bemerkung

Der Redundanz-Test ist besonders dann hilfreich, wenn Ax AC-Axiome enthält.

Satz 3.2.

Sei $\succ$ eine LPO oder eine KBO mit totaler Präzedenz. Sei

R_0: $f(f(x,y),z) \to f(x, f(y,z))$

E_0: $f(x,y) = f(y,x)$
$f(x, f(y,z)) = f(y, f(x,z))$
$f(x, f(y,z)) = f(z, f(y,x))$
$f(x, f(y,z)) = f(y, f(z,x))$

Wenn $s = t$ nicht mit (R_0, E_0) simplifiziert oder subsumiert werden kann, so gilt: $s =_{R_0 \cup E_0} t$ impliziert $s \Downarrow_\rhd t$ in $R_0(E_0)$.

Beispiel

Es gilt $f(f(t_1,t_0), f(t_2,t_3)) =_{R_0 \cup E_0} f(t_0, f(t_1, f(t_2, t_3)))$, also ist diese Gleichung in (R, E) redundant, falls $R_0 \subseteq R$ und $E_0 \subseteq E$.

Beachte

$R_0(E_0)$ ist auf den Grundtermen in Term(F) konfluent, falls $f \in F$ gilt.

Test auf AC-Gleichheit

Lemma 3.7

Seien $s, t \in \mathsf{Term}(F, V)$, $\{x_1, \ldots, x_n\} = \mathsf{Var}(s = t)$,

$\sigma = \{x_i \leftarrow a_i \mid i = 1, \ldots, n\}$ mit $a_i \notin F$.

Sei $\succ_0$ eine LPO oder KBO mit totaler Präzedenz $>_F$ auf $\{a_1, \ldots, a_n\} \cup F$.

Es gilt

$s =_{R_0 \cup E_0} t \text{ gdw } \sigma(s) \downarrow \sigma(t)$ in $R_0(E_0)$ bezüglich $\succ_0$.

Also

Der Test auf AC-Gleichheit ist einfach.

Beispiel Abelsche Gruppen

Sei

Ax:	$(x + y) + z = x + (y + z)$	$x + 0 = x$
	$x + y = y + x$	$x + -x = 0$
Ziel:	$a + m \cdot b + -a =_{Ax} m \cdot b$	$m > 0$

Hier ist $m \cdot x \equiv x + \cdots + x$ (m Summanden).

Das Redundanzkriterium aus dem letzten Satz

a) stoppt die Bildung kritischer Paare mit den AC-Gleichungen allein, nachdem (R_0, E_0) erzeugt ist.

b) erkennt viele sonstige Gleichungen als redundant.

c) beschleunigt die Zeit für die Herleitung des Ziels erheblich: Sei $\succ$ die LPO zur Präzedenz $- >_F + >_F 0 >_F a >_F b$: Beschleunigung der Beweissuche von fast 2 Stunden auf weniger als 2 Sekunden (für $m = 7$).

3.3 Allgemeinere Ziele

Beachte

Wir betrachten in Kapitel 3 nur Formeln über sig $= (F, P, \alpha)$ mit $P = \{=\}$. Atome haben also die Form $s = t$.

In Abschnitt 3.2 wurden Beweisaufgaben betrachtet, die zu Klauselmengen S führen mit
(1) Jede Klausel C aus S besteht nur aus einem Literal.
(2) Es gibt nur ein negatives Literal.
(3) Das negative Literal ist ein Grundliteral.

Jetzt

Wir fordern nur noch (1) von S.

Typische Beweisaufgabe

Gegeben:	$\mathcal{A}x = \{\varphi_1, \ldots, \varphi_n\}$ und φ mit
	$\varphi_i \equiv \forall^* s_i = t_i$ und $\varphi \equiv Q^* C$ mit C eine Klausel.
Frage:	Gilt $\mathcal{A}x \models \varphi$?

Beachte

Dies führt zu einer Klauselmenge S, die (1) erfüllt.

Beispiel/Bemerkung

Beispiel Sei

$\varphi \equiv \forall x.\exists y.(\neg A_1(x,y) \vee \neg A_2(x,y) \vee B_1(x,y) \vee B_2(x,y))$

Die Skolemisierung von $\neg\varphi$ liefert

$$\neg\forall x.\exists y.(\neg A_1(x,y) \vee \neg A_2(x,y) \vee B_1(x,y) \vee B_2(x,y))$$

$$\exists x.\forall y.\neg(\neg A_1(x,y) \vee \neg A_2(x,y) \vee B_1(x,y) \vee B_2(x,y))$$

$$\exists x.\forall y.(A_1(x,y) \wedge A_2(x,y) \wedge \neg B_1(x,y) \wedge \neg B_2(x,y))$$

$$\forall y.(A_1(a,y) \wedge A_2(a,y) \wedge \neg B_1(a,y) \wedge \neg B_2(a,y))$$

Dies liefert 4 Klauseln der Länge 1:

$A_1(a,y) \qquad A_2(a,y) \qquad \neg B_1(a,y) \qquad \neg B_2(a,y)$

Davon sind 2 Klauseln negativ.

Bemerkung

Enthält C kein positives Literal, so enthält S nur positive Unit-Klauseln. Dann ist S immer H-erfüllbar. Also gilt dann nicht $\mathcal{A}x \models Q^* C$.

H-Erfüllbarkeit von Unit-Klausel-Mengen

Bezeichnung

C heißt eine **Unit-Klausel**, falls C nur ein Literal enthält.

Problemstellung jetzt

Gegeben: S, eine Menge von Unit-Klauseln.

Frage: Ist S H-erfüllbar?

Lemma 3.8

Sei $E = \{s_1 = t_1, \ldots, s_n = t_n\}$ und
$U = \{u_1 \neq v_1, \ldots, u_m \neq v_m\}$ und $S = E \cup U$. Es ist S genau dann H-unerfüllbar, wenn es ein i gibt, so dass $u_i = v_i$ E-lösbar ist.

Beweis: Übung. Siehe dazu die Vorüberlegung in Abschnitt 3.2.

Voraussetzung ab jetzt: $U \neq \emptyset$.

Berechnung von E-Lösungen: Narrowing

Problemstellung

Eingabe: R ein grund-konvergentes Regelsystem,
$s = t$ eine Gleichung.
Frage: Gibt es eine R-Lösung σ zu $s = t$?

Grundidee für Narrowing

1. Sei σ R-Lösung zu $s = t$, also $\sigma(s) =_R \sigma(t)$.
   Setze $\tau(x) :\equiv \sigma(x)\downarrow$. Dann gilt auch $\tau(s) =_R \tau(t)$.
   Also: Es reicht „irreduzible Lösungen“ τ zu suchen.
2. Sei $\succ$ Reduktionsordnung, total auf Term(F), $\longrightarrow_R \subseteq \succ$.
   Sei $\tau(s) =_R \tau(t)$, also $\tau(s) \downarrow \tau(t)$.
   Gilt $\tau(s) \equiv \tau(t)$, so sind s und t unifizierbar.
   Sonst sei o.B.d.A. $\tau(s) \succ \tau(t)$. Wegen $\tau(s) \downarrow_R \tau(t)$ ist $\tau(s)$ reduzierbar, etwa mit $l \rightarrow r$ aus R. Dann gibt es eine Überlappung von $l \rightarrow r$ auf $s \rightarrow t$ und das kritische Paar $\langle u \equiv \mu(s[r]_\pi), v \equiv \mu(t) \rangle$ mit $\mu = \mathrm{mgu}(s/\pi, l)$ und $\tau = \varrho \circ \mu$. Dann ist ϱ eine R-Lösung zu $u = v$ mit $\{\tau(s), \tau(t)\} \succ\!\!\succ \{\varrho(u), \varrho(v)\}$.

Also

Narrowing verkleinert Grundlösungen.

Narrowing (1)

Problemstellung

Gegeben: R grund-konvergentes Regelsystem.
$Z = \{u_1 = v_1, \ldots, u_m = v_m\}$ Menge von Zielen.
Frage: Ist ein $u_i = v_i$ aus Z R-lösbar?

Ansatz

1. Ist ein Paar (u_i, v_i) unifizierbar, so ist Z lösbar.
2. Sonst wird jedes $u_i = v_i$ durch Narrowing-Schritte gemäss der Grundidee überdeckt. Dies erzeugt neue Ziele.
3. Man darf jedes Ziel mit R simplifizieren ohne die Lösbarkeit von Z zu verlieren.

Definition 3.10.

Ein **kritisches Ziel** zu R und Z ist ein kritisches Paar zur Überlappung von $l \rightarrow r$ aus R auf $u \rightarrow v$ mit $u \doteq v$ aus Z, $\mu = \text{mgu}(u/\pi, l)$ und $\mu(u) \not\succ \mu(v)$. Sei $\boldsymbol{CP_{\mathcal{N}}(R, u = v)}$ die Menge der kritischen Ziele zu R auf $u = v$.

Narrowing (2)

Das Inferenzsystem $\mathcal{N}$

Sei R fest. Sei $\mathcal{N}$ das folgende Inferenzsystem.

$$\frac{Z \cup \{u \doteq v\}}{Z \cup \{s = v\}} \quad \text{falls } u \longrightarrow_R s$$

$$\frac{Z \cup \{u = v\}}{\text{ERFOLG}} \quad \begin{array}{l}\text{falls } u \text{ und } v \text{ unifizierbar und} \\ u, v \text{ irreduzibel sind}\end{array}$$

$$\frac{Z \cup \{u = v\}}{Z \cup CP_{\mathcal{N}}(R, u = v)} \quad \begin{array}{l}\text{falls } u \text{ und } v \text{ nicht unifizierbar und} \\ u, v \text{ irreduzibel sind}\end{array}$$

Satz 3.3.

Es gilt $Z \overset{*}{\underset{\mathcal{N}}{\vdash}}$ ERFOLG gdw ein $u = v \in Z$ ist R-lösbar.

ohne Beweis.

Zur H-Erfüllbarkeit von S

Seien

E, Z Gleichungssysteme.

$S = E \cup \{s \neq t \mid s = t \text{ in } Z\}$.

Strategie für den H-Erfüllbarkeitstest

Parallel E zu R vervollständigen und den Narrowing-Algorithmus ablaufen lassen.

Also

Inferenzsystem $\mathcal{GN}$ arbeitet auf 4-Tupeln (E, R, G, Z) mit

E enthält die ungerichteten Gleichungen.

R enthält die gerichteten Gleichungen.

G enthält die redundanten Gleichungen.

Z enthält die Ziele.

$\mathcal{GN} = \mathcal{GN}_\succ$ arbeitet mit fester Grund-Reduktionsordnung.

Definition 3.11.

$CP_\mathcal{N}(R, E, s = t) = CP_\mathcal{N}(R, s = t) \cup CP_\mathcal{N}(E, s = t)$.

$CP_\mathcal{N}(E, s = t)$ ist die Menge der kritischen Paare $u = v$ zu $l \to r$ auf $s = t$ mit $l \doteq r$ in E, $\mu = \text{mgu}(s/\pi, l)$ und $u \not\succ \mu(s) \not\prec v$.

Das Inferenzsystem $\mathcal{GN}$

Definition 3.12.

Das Inferenzsystem $\mathcal{GN}_\succ$ ist gegeben durch

- die Inferenzregeln von $\mathcal{G}$ und
- die folgenden Narrowing-Regeln:

Simplifizieren eines Ziels

$$\frac{(E, R, G, Z \cup \{s \doteq t\})}{(E, R, G, Z \cup \{u = t\})} \quad \text{falls } s \longrightarrow_{R(E \cup G)} u$$

Subsumieren eines Ziels

$$\frac{(E, R, G, Z \cup \{s = t, u \doteq v\})}{(E, R, G, Z \cup \{s = t\})} \quad \begin{array}{l}\text{falls es } \sigma \text{ gibt mit}\\ u \equiv \sigma(s) \wedge v \equiv \sigma(t)\end{array}$$

Überdecken eines Ziels

$$\frac{(E, R, G, Z \cup \{s = t\})}{(E, R, G, Z \cup \{s = t, u = v\})} \quad \begin{array}{l}\text{falls } s, t \text{ nicht unifizierbar und}\\ u = v \in CP_\mathcal{N}(R, E, s = t)\end{array}$$

Erfolgsmeldung

$$\frac{(E, R, G, Z \cup \{s = t\})}{\text{ERFOLG}} \quad \text{falls } s, t \text{ unifizierbar}$$

Korrektheit von $\mathcal{GN}$

Bezeichnung

Z ist E-lösbar, falls ein $u = v$ aus Z E-lösbar ist.

Z ist (E, R)-lösbar, falls ein $u = v$ aus Z $(E \cup R)$-lösbar ist.

Sei

$(E_i, R_i, G_i, Z_i)_{i \geq 0}$ eine $\mathcal{GN}$-Ableitung.

Lemma 3.9

a) Z_i ist (E_i, R_i)-lösbar gdw Z_{i+1} ist (E_{i+1}, R_{i+1})-lösbar.

b) Gilt $(E, \varnothing, Z) \overset{*}{\underset{\mathcal{GN}}{\vdash}}$ ERFOLG, so ist Z E-lösbar.

Beweis:

a) Übung.

b) Dies folgt aus a).

Beachte

Sei $=_i \; = \; =_{E_i \cup R_i}$. Es gilt $=_i \; = \; =_{i+1}$ auf $\mathrm{Term}(F)$.

Also gilt für jede Gleichung $u = v$: Sie ist (E_i, R_i)-lösbar gdw sie ist (E_{i+1}, R_{i+1})-lösbar.

Also: „lösbar" heißt „(E_0, R_0)-lösbar".

Vollständigkeit von $\mathcal{GN}$ (1)

Definition 3.13. (Komplexität von Zielen)

Sei $\succ_Z = (\ggcurly, \rhd\!\rhd)$ die lexikographische Kombination von $\ggcurly$ und $\rhd\!\rhd$. Dabei ist $\rhd$ die Umschließungsordnung. Sei

$$K(s = t, \sigma) = \begin{cases} \{\{\sigma(s), \sigma(t)\}, \{s, t\}\} & s = t \text{ lösbar} \\ \{\{\max\}, \{s, t\}\} & \text{sonst} \end{cases}$$

$K(s = t) = \min_{\succ_Z}\{K(s = t, \sigma) \mid \sigma \in \mathrm{GSub}(s = t)\}$

$K(Z) = \min_{\succ_Z}\{K(s = t) \mid s = t \text{ in } Z\}$

Beachte

Alle Ordnungen sind Noethersch. Also existieren die Minima. $K(Z)$ ist die Komplexität der kleinsten Lösung einer Gleichung aus Z.

Definition 3.14. (Fairness)

Eine $\mathcal{GN}$-Ableitung heißt **fair**, falls gilt:

$CP(R^\infty, E^\infty) \subseteq \bigcup_{i \in \mathbb{N}} E_i$ (alle kritischen Paare sind gebildet).

$CP_{\mathcal{N}}(R^\infty, E^\infty, Z^\infty) \subseteq \bigcup_{i \in \mathbb{N}} Z_i$ (alle Ziele sind überdeckt).

Auf kein (E_i, R_i, G_i, Z_i) ist die ERFOLG-Regel anwendbar oder es gibt ein (E_j, R_j, G_j, Z_j), auf das die ERFOLG-Regel angewendet wird.

Vollständigkeit von $\mathcal{GN}$ (2)

Lemma 3.10

Sei $(E_i, R_i, G_i, Z_i)_{i\geq 0}$ eine $\mathcal{GN}$-Ableitung.

a) Es gilt $K(Z_{i-1}) \succeq_Z K(Z_i)$ für alle i der Folge.
b) Ist die Ableitung fair und ist Z_0 (E_0, R_0)-lösbar, so gilt für alle i der Folge entweder $(E_i, R_i, G_i, Z_i) \vdash$ ERFOLG oder es gibt ein $j > i$ mit $K(Z_i) \succ K(Z_j)$. Also bricht die Folge dann mit ERFOLG ab.

Beweis:

a) Man rechne dies nach.
b) Da Z_0 (E_0, R_0)-lösbar ist, ist auch Z_i (E_i, R_i)-lösbar. Sei $\sigma(s) = \sigma(t)$ mit $s = t$ aus Z_i die kleinste Lösung, die Z_i enthält. Nach Satz 3.1 gibt es ein k mit $\sigma(s) \downarrow \sigma(t)$ in (E_j, R_j) für alle $j \geq k$.
   Sind s, t unifizierbar, so gilt $(E_i, R_i, G_i, Z_i) \vdash$ ERFOLG. Sonst wird wegen der Fairness zu einem Zeitpunkt $j > i, k$ die Gleichung $s = t$ überdeckt und dabei ein kritisches Ziel $u = v$ erzeugt, das eine echt kleinere Lösung $\varrho(u) = \varrho(v)$ enthält als $\sigma(s) = \sigma(t)$. Dann gilt $K(Z_i) \succ_Z K(Z_j)$. Da $\succ_Z$ Noethersch ist, muss nach endlicher Zeit der Erfolgsfall eintreten. Dann bricht die Kette ab.

Ergebnis zu $\mathcal{GN}$

Satz 3.4.

$\mathcal{GN}_\succ$ ist korrekt und vollständig: Es gilt $(E, \emptyset, \emptyset, Z) \overset{*}{\vdash}$ ERFOLG gdw Z ist E-lösbar.
Jede faire $\mathcal{GN}$-Ableitung $(E_i, R_i, G_i, Z_i)_{i\geq 0}$ mit $(E_0, R_0, G_0, Z_0) = (E, \emptyset, \emptyset, Z)$ bricht genau dann mit ERFOLG ab, wenn Z E-lösbar ist.

Beweis:

Dies folgt aus Lemma 3.9 und Lemma 3.10.

Satz 3.5.

Sei $S = E \cup U$ die Klauselmenge zum Problem „$\mathcal{A}x \models Q^*C$?“ und sei $Z = \{u = v \mid u \neq v \text{ in } U\}$.

Dann gilt

$\mathcal{A}x \models Q^*C \text{ gdw } (E, \emptyset, \emptyset, Z) \overset{*}{\vdash}_{\mathcal{GN}}$ ERFOLG

Beweis:

Dies folgt aus Satz 3.4 und der Vorüberlegung.

Beispiel 1

$\mathcal{A}x : \forall x, y, z.[x \cdot 1 = x \wedge x \cdot x^{-1} = 1 \wedge (x \cdot y) \cdot z = x \cdot (y \cdot z)]$.
$\varphi \equiv \exists x \forall y.\ x \cdot y = y$.

Dies liefert $S = E \cup \{\neg x \cdot g(x) = g(x)\}$ mit
$E : x \cdot 1 = x \qquad x \cdot x^{-1} = 1 \qquad (x \cdot y) \cdot z = x \cdot (y \cdot z)$.

Es ist $Z = \{x \cdot g(x) = g(x)\}$.

Die Vervollständigung von E mit einer LPO liefert u.a. die Regeln

(1) $x \cdot 1 \to x$ (3) $(x \cdot y) \cdot z \to x \cdot (y \cdot z)$
(2) $x \cdot x^{-1} \to 1$ (4) $1 \cdot x \to x$

Überlappung der Regel (4) in das Ziel liefert

$$\begin{tikzcd} & 1 \cdot g(1) \arrow[dl] \arrow[dr] & \\ g(1) & & g(1) \end{tikzcd}$$

also das neue Ziel $g(1) = g(1)$. Unifikation liefert jetzt ERFOLG.

Also ist Z E-lösbar und es gilt $\mathcal{A}x \models \varphi$.

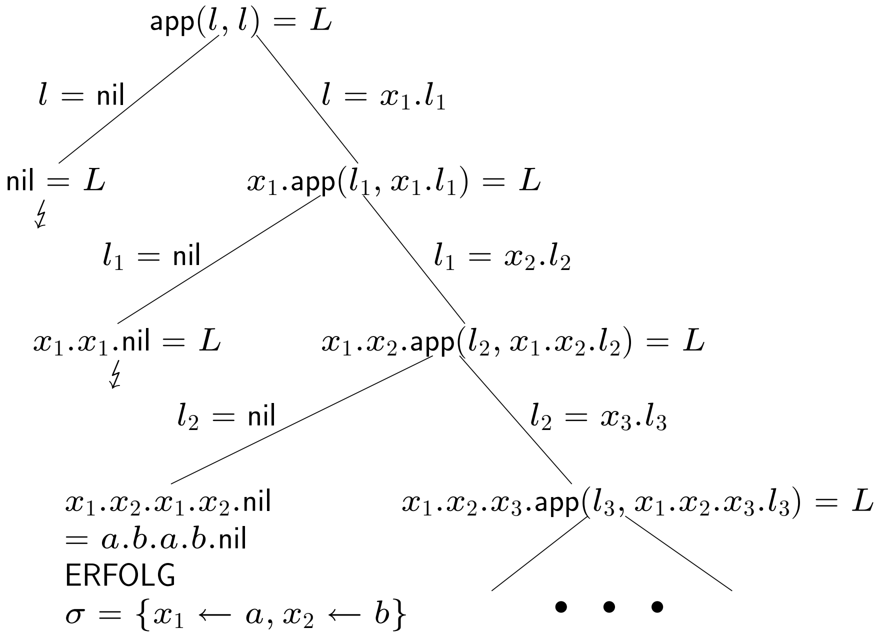
Beachte: Die berechnete E-Lösung zu Z ist $\sigma = \{x \leftarrow 1\}$.

Beispiel Narrowing

R : $\mathsf{app}(\mathsf{nil}, l_2) \to l_2$
$\mathsf{app}(x.l_1, l_2) \to x.\mathsf{app}(l_1, l_2)$

Gesucht: R-Lösungen zu $\mathsf{app}(l, l) = a.b.a.b.\mathsf{nil} \equiv L$

Narrowing: Suchbaum



$\rightsquigarrow \sigma(l) = a.b.\mathsf{nil}$ ist eine R-Lösung zu $\mathsf{app}(l, l) = a.b.a.b.\mathsf{nil}$