

# Lösungsvorschlag *Termersetzungssysteme* – Blatt 9

## Aufgabe 1

a) Beweis mittels noetherscher Induktion über  $\triangleright$ . Sei nun  $s = f(s_1, \dots, s_n)$  ein Term mit  $s \succ_{rpo} s$ . Die Induktionshypothese ist dann, dass  $t \succ_{rpo} t$  für keinen Teilterm  $t$  von  $s$  gilt. Wir werden zeigen, dass jede der drei Regeln der Definition der RPO zu einem Widerspruch führt:

**Fall 1:** Es gilt  $s_i \succeq_{rpo} s$  für ein  $s_i$ . Da die RPO die Teiltermeigenschaft hat und  $s_i$  Teilterm von  $s$  ist, folgt  $s \succ_{rpo} s_i$ . Mit der Transitivität erhalten wir so aber  $s_i \succ_{rpo} s_i$ . Widerspruch zur Induktionshypothese.

**Fall 2:** Kann nicht eintreten, da wir keine zwei unterschiedlichen äußeren Funktionssymbole haben.

**Fall 3:** Um  $s \succ_{rpo} s$  mittels der dritten Regel zu zeigen müsste gelten

$$\{s_1, \dots, s_n\} (\succ_{rpo})_{mul} \{s_1, \dots, s_n\}.$$

Da beide Mengen aber identisch sind, kann dies nicht der Fall sein. Genauer: Sei  $L$  die linke Menge und  $R$  die rechte Seite. Für je zwei Menge  $X$  und  $Y$  mit  $R = (M \setminus X) \cup Y$  gilt  $X = Y$  wegen  $L = R$ . Hätten wir nun  $L (\succ_{rpo})_{mul} R$ , so müsste es für jedes  $y \in Y$  ein  $x \in X$  geben mit  $x \succ_{rpo} y$ . Aus  $X = Y$  folgt aber dann, dass es für jedes  $x \in X$  ein  $x' \in X$  gibt mit  $x \succ_{rpo} x'$ . Mit der Transitivität erhalten wir dann eine unendliche Kette  $x_0 \succ_{rpo} x_1 \succ_{rpo} x_2 \dots$ , die wegen der endlich vielen Elemente in  $X$  irgendwann zu einem  $x_i$  kommt, das schon dagewesen war. Damit würde  $x_i \succ_{rpo} x_i$  im Widerspruch zur Induktionshypothese gelten.

b) Die Terminierung von  $\mathcal{R}_1$  kann mit keiner Simplifikationsordnung gezeigt werden. Angenommen, wir hätten eine Simplifikationsordnung  $\succ$  mit  $l \succ r$  für alle Regeln in  $\mathcal{R}_1$ . Aufgrund der Stabilität von  $\succ$  gilt dann:

$$\text{plus}(s(x), s(x)) \succ s(\text{plus}(\text{minus}(x, s(x)), \text{double}(s(x))))$$

Da  $\succ_{emb} \subseteq \succ$ , gilt aber auch

$$s(\text{plus}(\text{minus}(x, s(x)), \text{double}(s(x)))) \succ \text{plus}(s(x), s(x))$$

im Widerspruch zur Irreflexivität von  $\succ$ .

Die Terminierung von  $\mathcal{R}_2$  lässt sich dagegen mit der RPOS zeigen, wobei die Argumente von **plus** lexikographisch und die Argumente von **times** als Multimengen verglichen werde. Ferner haben wir die Präzedenz  $\text{times} \sqsubset \text{plus}$ . Damit gilt:

$$\frac{\frac{\text{plus}(x, y) \succ x}{\text{plus}(x, y) \succ x} \succ_{emb} \quad \frac{\frac{\text{plus}(x, y) \succ y \succ_{emb} \quad \text{plus}(\text{plus}(x, y), z) \succ z \succ_{emb}}{\text{plus}(\text{plus}(x, y), z) \succ \text{plus}(y, z)} \quad 3}{\text{plus}(\text{plus}(x, y), z) \succ \text{plus}(x, \text{plus}(y, z))} \quad 3}{\text{times}(x, \text{s}(y)) \succ x \succ_{emb} \quad \frac{\frac{\text{s}(y) \succ y \succ_{emb}}{\text{times}(x, \text{s}(y)) \succ \text{times}(y, x)} \quad 3}{\text{times}(x, \text{s}(y)) \succ \text{plus}(x, \text{times}(y, x))} \quad 2}$$

Die Terminierung von  $\mathcal{R}_3$  lässt sich wieder mit keiner Simplifikationsordnung zeigen. Sei dazu wieder  $\succ$  eine solche Simplifikationsordnung. Es gilt (wieder mit der Stabilität von  $\succ$ ):

$$\begin{aligned} \text{gcd}(\text{s}(\text{true}), \text{s}(\text{s}(\text{true}))) \succ \text{if}(\text{le}(\text{s}(\text{true}), \text{true}), \text{s}(\text{true}), \text{s}(\text{s}(\text{true}))) \succ_{emb} \\ \text{if}(\text{true}, \text{s}(\text{true}), \text{s}(\text{s}(\text{true}))) \succ \text{gcd}(\text{minus}(\text{true}, \text{s}(\text{true})), \text{s}(\text{s}(\text{true}))) \succ_{emb} \text{gcd}(\text{s}(\text{true}), \text{s}(\text{s}(\text{true}))) \end{aligned}$$

im Widerspruch zur Fundiertheit von  $\succ$ .

Die Terminierung von  $\mathcal{R}_4$  lässt sich mittels RPO und Multimengenvergleich zeigen, wobei wir die Präzedenz  $\text{times} \sqsubset \text{minus}$  benötigen.

$$\frac{\frac{\frac{\text{times}(\text{minus}(y), y) \succ \text{times}(y, y) \succ_{emb}}{\text{times}(\text{minus}(y), y) \succ \text{minus}(\text{times}(y, y))} \quad 2}{\text{times}(x, \text{times}(\text{minus}(y), y)) \succ \text{times}(\text{minus}(\text{times}(y, y)), x)} \quad 3}$$

Die Terminierung von  $\mathcal{R}_5$  lässt mittels RPO zeigen: Mit der Präzedenz  $\text{h} \sqsubset \text{s} \sqsubset \text{g}$  erhalten wir:

$$\frac{\frac{\frac{\text{s}(x) \succ x \succ_{emb}}{\text{s}(x) \succ \text{g}(x)} \quad 2}{\text{f}(\text{s}(x), x) \succ \text{f}(x, \text{g}(x))} \quad 3}{\frac{\frac{\text{h}(x) \succ x \succ_{emb}}{\text{h}(x) \succ \text{s}(x)} \quad 2}{\text{f}(x, \text{h}(x)) \succ \text{f}(\text{s}(x), x)} \quad 3}$$

## Aufgabe 2

a) Wir benutzen folgende Polynominterpretation  $\mathcal{P}$ :

$$\begin{aligned} \mathcal{P}(\text{rev}(x)) &= 2x + 2 & \mathcal{P}(\text{r}(x)) &= 2x + y + 1 \\ \mathcal{P}(\text{nil}) &= 0 & \mathcal{P}(\text{cons}(x, y)) &= 1 + x + y \end{aligned}$$

Dann haben wir folgende Polynominterpretationen der linken und rechten Seiten von  $\mathcal{R}_1$ :

$$\begin{aligned}\mathcal{P}(\text{rev}(\text{rev}(x))) &= 4x + 6 > \mathcal{P}(x) = x \\ \mathcal{P}(\text{rev}(x)) &= 2x + 2 > \mathcal{P}(r(x, \text{nil})) = 2x + 1 \\ \mathcal{P}(r(\text{nil}, y)) &= y + 1 > y \\ \mathcal{P}(r(\text{cons}(x, z), y)) &= 2x + 2z + y + 3 > \mathcal{P}(r(z, \text{cons}(x, y))) = 2z + x + y + 2\end{aligned}$$

b) Diese TESe sind etwas kniffliger. Beginnen wir mit  $\mathcal{R}_2$ . Der Trick ist, dass bei der einfachen Verschachtelung  $f$  größer als  $s$  sein muss, bei der dreifachen Schachtelung aber dann der Term mit dem doppelten  $s$  größer als der mit dem doppelten  $f$  sein muss. Wir wählen:

$$\begin{aligned}\mathcal{P}(0) &= 0 \\ \mathcal{P}(f(x)) &= x + 3 \\ \mathcal{P}(s(x)) &= 2x + 2\end{aligned}$$

Damit gilt:

$$\begin{aligned}\mathcal{P}(f(0)) &= 0 + 3 = 3 > 2 = 2 \cdot 0 + 2 = \mathcal{P}(s(0)) \\ \mathcal{P}(f(s(0))) &= (2 \cdot 0 + 2) + 3 = 5 > \mathcal{P}(s(0)) = 2 \\ \mathcal{P}(f(s(s(x)))) &= (2 \cdot (2x + 2) + 2) + 3 = 4x + 9 > \mathcal{P}(f(f(s(x)))) = \\ &((2x + 2) + 3) + 3 = 2x + 8\end{aligned}$$

Damit ist die Terminierung von  $\mathcal{R}_2$  gezeigt. Eine andere Möglichkeit wäre

$$\begin{aligned}\mathcal{P}(0) &= 0 \\ \mathcal{P}(f(x)) &= 2 + x \\ \mathcal{P}(s(x)) &= 1 + 3x\end{aligned}$$

Nun zu  $\mathcal{R}_3$ . Hier schreibt man am besten die allgemeinen Formen hin und stellt gewisse Constraints auf, indem man  $x$  und/oder  $y$  auf Null setzt. Mit etwas Probieren erhält man

$$\begin{aligned}\mathcal{P}(0) &= 1 \\ \mathcal{P}(f(x, y)) &= 2x + y \\ \mathcal{P}(s(x)) &= 2x + 2\end{aligned}$$

Damit gilt:

$$\begin{aligned}\mathcal{P}(f(0,0)) &= 2 \cdot 1 + 1 = 3 > 1 = \mathcal{P}(0) \\ \mathcal{P}(f(s(x), y)) &= 2(2x + 2) + y = 4x + y + 4 > \\ \mathcal{P}(f(x, s(x))) &= 2x + 2x + 2 = 4x + 2 \\ \mathcal{P}(f(x, s(y))) &= 2x + 2y + 2 > \mathcal{P}(f(y, x)) = 2y + x\end{aligned}$$

und die Terminierung von  $\mathcal{R}_3$  ist gezeigt. Eine andere Möglichkeit wäre

$$\begin{aligned}\mathcal{P}(0) &= 0 \\ \mathcal{P}(f(x, y)) &= 1 + 2x + y \\ \mathcal{P}(s(x)) &= 1 + 2x\end{aligned}$$

c) Bei  $\mathcal{R}_2$  haben wir nur einstellige Symbole, der Status ist also unerheblich und wir müssen es mit der LPO versuchen. Die erste Regel erfordert bereits die Präzedenz  $f \sqsupset s$ . In der dritten Regel scheitern wir dann, weil wir  $s(s(x)) \succ_{lpo} f(s(x))$  nicht zeigen können.

Bei  $\mathcal{R}_3$  erfordert die zweite Regel einen lexikographischen Vergleich mit Status  $\langle 1, 2 \rangle$ , die dritte Regel aber würde einen Multimengenvergleich erfordern. Beides gleichzeitig geht nicht, also versagt RPOS hier ebenfalls.

### Aufgabe 3

a) Gemäß dem Algorithmus UNIFY wenden wir  $\implies$  an bis es nicht mehr geht.

$$\begin{aligned}\implies & \{f(x, g(x), h(a)) =^? f(h(z), g(y), y) \quad \} \\ \implies & \{x =^? h(z), g(x) =^? g(y), h(a) =^? y \quad \} \\ \implies & \{x =^? h(z), g(h(z)) =^? g(y), h(a) =^? y \quad \} \\ \implies & \{x =^? h(z), h(z) =^? y, h(a) =^? y \quad \} \\ \implies & \{x =^? h(z), h(z) =^? y, y =^? h(a) \quad \} \\ \implies & \{x =^? h(z), h(z) =^? h(a), y =^? h(a) \quad \} \\ \implies & \{x =^? h(z), z =^? a, y =^? h(a) \quad \}\end{aligned}$$

Fertig, ausgabe TRUE, da Unifikationsproben in gelöster Form vorliegt.

$$\begin{aligned}\implies & \{f(y, z, g(z), y) =^? f(h(x), g(u), x, g(x)) \quad \} \\ \implies & \{y =^? h(x), z =^? g(u), g(z) =^? x, y =^? g(x) \quad \} \\ \implies & \{g(x) =^? h(x), z =^? g(u), g(z) =^? x, y =^? g(x)\quad \}\end{aligned}$$

Der herkömmliche UNIFY-Algorithmus würde hier noch nicht stoppen. Im Skript wird jedoch beschrieben, dass sich der Algorithmus verbessern lässt, indem bei

bemerken *Occur*- und *Clash-Failures* sofort abgebrochen wird. In diesem Fall können wir nun mit Ausgabe **FALSE** abbrechen, da  $g(x) =^? h(x)$  bereits einen *Clash-Failure* liefert.

$$\begin{aligned}
&\Longrightarrow \{c(f(g(x), h(z), y, z), f(y, x, g(h(z)), h(y))) =^? c(u, u) \} \\
&\Longrightarrow \{f(g(x), h(z), y, z) =^? u, f(y, x, g(h(z)), h(y)) =^? u \} \\
&\Longrightarrow \{u =^? f(g(x), h(z), y, z), f(y, x, g(h(z)), h(y)) =^? u \} \\
&\Longrightarrow \{u =^? f(g(x), h(z), y, z), f(y, x, g(h(z)), h(y)) =^? f(g(x), h(z), y, z) \} \\
&\Longrightarrow \{u =^? f(g(x), h(z), y, z), y =^? g(x), x =^? h(z), g(h(z)) =^? y, h(y) =^? z \} \\
&\Longrightarrow \{u =^? \dots, y =^? g(x), x =^? h(z), g(h(z)) =^? g(x), h(g(x)) =^? z \} \\
&\Longrightarrow \{u =^? \dots, y =^? g(h(z)), x =^? h(z), g(h(z)) =^? g(h(z)), h(g(h(z))) =^? z \} \\
&\Longrightarrow \{u =^? \dots, y =^? g(h(z)), x =^? h(z), h(g(h(z))) =^? z \} \\
&\Longrightarrow \{u =^? \dots, y =^? g(h(z)), x =^? h(z), z =^? h(g(h(z))) \}
\end{aligned}$$

Es ist keine weitere Regel mehr anwendbar, das Unifikationsproblem ist aber nicht in gelöster Form, wir terminieren daher mit Ausgabe **FALSE**. Grund ist der *Occur-Failure* in der letzten Gleichung.

$$\begin{aligned}
&\Longrightarrow \{f(g(x_1, x_1), x_2, x_3) =^? f(x_4, g(x_3, x_3), g(x_4, x_4)) \} \\
&\Longrightarrow \{g(x_1, x_1) =^? x_4, x_2 =^? g(x_3, x_3), x_3 =^? g(x_4, x_4) \} \\
&\Longrightarrow \{x_4 =^? g(x_1, x_1), x_2 =^? g(x_3, x_3), x_3 =^? g(x_4, x_4) \} \\
&\Longrightarrow \{x_4 =^? g(x_1, x_1), x_2 =^? g(x_3, x_3), x_3 =^? g(g(x_1, x_1), g(x_1, x_1)) \} \\
&\Longrightarrow \{ \begin{array}{l} x_4 =^? g(x_1, x_1), x_2 =^? g(g(g(x_1, x_1), g(x_1, x_1)), g(g(x_1, x_1), g(x_1, x_1))), \\ x_3 =^? g(g(x_1, x_1), g(x_1, x_1)) \end{array} \}
\end{aligned}$$

Das Unifikationsproblem liegt nun in gelöster Form vor und wir terminieren mit **TRUE**.

**b)** Sei  $S$  ein lösbares Unifikationsproblem. Dann terminiert der Unifikationsalgorithmus **UNIFY** und liefert in seiner gelösten Form einen allgemeinsten Unifikator  $\sigma_S$ . Nach Lemma 5.1.6 gilt für *alle* Unifikatoren  $\delta$  von  $S$  jeweils  $\delta = \sigma_S \delta$ . Dies gilt also insbesondere für  $\delta = \sigma_S$  und wir erhalten  $\sigma_S = \sigma_S \sigma_S$ . Der Unifikationsalgorithmus liefert also einen *idempotenten* allgemeinsten Unifikator.