Information and Computation **159**, 2–21 (2000)
doi:10.1006/inco.2000.2874, available online at http://www.idealibrary.com on IDEAL®

The Combinator S

Johannes Waldmann

Institut für Informatik, Universität Leipzig, Augustusplatz 10–11, D-04109 Leipzig, Germany
E-mail: joe@informatik.uni-leipzig.de
URL: http://www.informatik.uni-leipzig.de/~joe/

The combinator S has the reduction rule $S\,x\,y\,z \to x\,z\,(y\,z)$. We investigate properties of ground terms built from S alone. The first part of the paper shows that this term rewriting system admits no ground loops. This extends the known result of the absence of cycles. In the second part, we give a procedure that decides whether an S-term has a normal form. This algorithm makes use of rational tree languages. Finally we show that the set of normalizing S-terms is in itself a rational tree language, by explicitly giving its grammar. In all, this paper shows the surprisingly rich structures that are implied by a seemingly small rewrite rule. © 2000 Academic Press

1. INTRODUCTION

It is well known that the combinators **S** and **K** with their reduction rules $\mathbf{S}\,x\,y\,z \to x\,z\,(y\,z)$ and $\mathbf{K}\,x\,y \to x$ form a complete basis for combinatory logic. Therefore, most of the interesting properties of an $(\mathbf{S}, \mathbf{K})$-term are necessarily undecidable.

Now $\mathbf{CL}(\mathbf{K})$ is strongly normalizing. One is led to assume that the difficulty of $\mathbf{CL}(\mathbf{S}, \mathbf{K})$ comes from **S**. That is the reason for studying the system $\mathbf{CL}(\mathbf{S})$. We will start by giving two examples of infinite reductions in $\mathbf{CL}(\mathbf{S})$.

Example 1 (Zachos [22]). Let $\mathsf{T} = \mathbf{S}\,\mathbf{S}$; $X_0 = \mathbf{S}\,\mathsf{T}\,\mathsf{T}$. The expression $X_0\,X_0$ has an infinite reduction.

Define $X_{n+1} = \mathsf{T}\,X_n$. All redexes that occur during the reduction of $X_0\,X_0$ have the form $X_n\,X_m$ for some n and m. There are two types of reduction steps: one goes *downward*,

$$X_{n+1}\,X_m = \mathbf{S}\,\mathbf{S}\,X_n\,X_m \to \mathbf{S}\,X_m\,\underline{(X_n\,X_m)},$$

and the other goes *upward*,

$$X_0\,X_m = \mathbf{S}\,\mathsf{T}\,\mathsf{T}\,X_m \to \mathsf{T}\,X_m\,(\mathsf{T}\,X_m) = X_{m+1}\,X_{m+1}.$$

It is interesting to note that the reduction graph of $X_0 X_0$ is a *pure line* (each expression contains exactly one redex).

Example 2 (Barendregt [3]). The expression $\mathsf{A}\,\mathsf{A}\,\mathsf{A}$, where $\mathsf{A} = \mathbf{S}\,\mathbf{S}\,\mathbf{S}$, has an infinite reduction.

Again the reduction can be described by a pattern,

$$Y_0 = \mathbf{S}\,\mathsf{A}\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}), \qquad Y_{n+1} = \mathsf{A}\;Y_n.$$

First we show that the above pattern is reached:

$$\begin{aligned} \mathsf{A}\,\mathsf{A}\,\mathsf{A} &\to \mathbf{S}\,\mathsf{A}\,(\mathbf{S}\,\mathsf{A})\,\mathsf{A} \to \mathsf{A}\,\mathsf{A}\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}) \to \mathbf{S}\,\mathsf{A}\,(\mathbf{S}\,\mathsf{A})(\mathbf{S}\,\mathsf{A}\,\mathsf{A}) \\ &\to \mathsf{A}\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A})\;Y_0 \to \mathbf{S}\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A})(\mathbf{S}\,\mathsf{A}\,\mathsf{A})(\mathbf{S}\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}))\;Y_0 \\ &\to \mathbf{S}\,\mathsf{A}\,\mathsf{A}\;Y_0 * \;\to\; Y_1\,Y_1 * . \end{aligned}$$

Here, $*$ denotes a subexpression whose value is irrelevant because no later reduction touches it.

Then we show how the pattern reduces. There are a *downward* move,

$$Y_{k+1}\,E = \mathsf{A}\;Y_k\,E \to \mathbf{S}\;Y_k\,(\mathbf{S}\;Y_k)\;E \to\; Y_k\,E *,$$

and an *upward* move,

$$Y_0\,F = \mathbf{S}\,\mathsf{A}\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A})\;F \to \mathsf{A}\,F\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}\,F) \to \mathbf{S}\,F\,(\mathbf{S}\,F)(\mathbf{S}\,\mathsf{A}\,\mathsf{A}\,F) \to F(\mathbf{S}\,\mathsf{A}\,\mathsf{A}\,F) * .$$

Here, the placeholder F is instantiated with Y_k to give

$$Y_0\;Y_k \twoheadrightarrow Y_k\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}\;Y_k) *$$

and now $E = \mathbf{S}\,\mathsf{A}\,\mathsf{A}\;Y_k$ gives

$$Y_k\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}\;Y_k) \twoheadrightarrow Y_0\,(\mathbf{S}\,\mathsf{A}\,\mathsf{A}\;Y_k) * \cdots \twoheadrightarrow \mathbf{S}\,\mathsf{A}\,\mathsf{A}\;Y_k * \;\cdots \to Y_{k+1}\,Y_{k+1} * \cdots .$$

This is an infinite head reduction.

Note that in both examples, the reduction follows a doubly periodic pattern that is more complicated than a cycle or a loop.

The rest of the paper is organized as follows. After providing some notation and preliminaries, in the first part of the paper we will prove that **CL**(**S**) does not admit ground loops. However, our method is not general enough to show the absence of non-ground loops. In the second part of the paper we will prove that it is decidable whether a ground **S**-term has a normal form. Finally, we give a regular grammar for the set of normalizing **S**-terms. The first part builds around one standard proof idea, recursive path orders. The second part applies a collection of useful ad hoc lemmas whose rather technical proofs are relegated to Appendix 1. The grammar presented in the third part is rather large. Its correctness has been verified by a computer program. Its underlying idea is described.

This paper is a revised and extended combination of [19, 20].

2. NOTATIONS AND PRELIMINARIES

We use standard notation for term rewriting systems (see [2]) and combinatory logic (see [3]).

We consider combinatory logic **CL**(**S**) as a term rewriting system. Its signature consists of one nullary symbol **S** and one binary symbol $\circ$ (application). **CL**(**S**) has only one rule,

$$(((\mathbf{S} \circ x) \circ y) \circ z) \to ((x \circ z) \circ (y \circ z)).$$

Symbols appear in different fonts. We use bold, uppercase, for combinators (**S**, **K**, ...); italic, lowercase, for variables in the system (x, y, ...), italic, uppercase; for meta-variables that denote terms or sets of terms (X, Y, ...); calligraphic, uppercase, for fixed sets of terms ($\mathscr{P}$, $\mathscr{Q}$, ...); and sans serif, uppercase, for certain fixed terms (T, A).

Unless stated otherwise, all terms are ground terms (they do not contain variables).

$X \trianglelefteq Y$ ($X \triangleleft Y$) means X is a (strict) subterm of Y.

$X \xrightarrow{R} Y$ means that X reduces to Y in one step by contracting the redex $R \trianglelefteq X$.

$X \to Y$ means $X \xrightarrow{R} Y$ for some R.

$X \hookrightarrow Y$ means that when X is reduced, Y occurs as a subterm: $\exists Z : X \to Z$ and $Z \trianglerighteq Y$.

The transitive closure of a relation R is denoted by R^+, while R^* denotes the transitive and reflexive closure of R.

We also write $\twoheadrightarrow$ for $\to^*$ and $\lhook\joinrel\twoheadrightarrow$ for $\hookrightarrow^*$.

We write $\downarrow(X)$ if X normalizes, and $X \twoheadrightarrow \infty$ or $\uparrow(X)$ if X has an infinite reduction.

Operations are understood to be extended from terms to sets of terms, in the following way:

If E and F are sets of terms, then $(E \circ F)$ denotes the set of terms $(X \circ Y)$ with $X \in E$ and $Y \in F$.

If E and F are sets, $E \to F$ denotes $\forall X \in E : \exists Y \in F : X \to Y$, while $E \not\to F$ stands for $\forall X \in E : \neg \exists Y \in F : X \to Y$.

Also, $\downarrow(E)$ denotes $\forall X \in E : \downarrow(X)$, and $\uparrow(E)$ denotes $\forall X \in E : \uparrow(X)$. We often write just a term when meaning a set containing exactly this term.

We follow the usual conventions about suppressing the application symbol and omitting parentheses. So the **S** reduction rule just given is written as $\mathbf{S}\,x\,y\,z \to x\,z\,(y\,z)$.

The system **CL**(**S**) is left-linear and non-overlapping, thus confluent. Moreover, it is non-erasing. Therefore [12], weak and strong normalization coincide: if one reduction leads to normal form, then all do. (In our notation, $\downarrow(X)$ and $\uparrow(X)$ are mutually exclusive.)

3. LOOPS

The introductory examples showed infinite reductions that obey certain patterns. Here we are going to show that these patterns cannot be too simple. (A precise formulation follows.)

The most basic repetitive pattern is a cycle:

DEFINITION 3. A *cycle* is a reduction $X \to^{+} X$.

A cycle can be composed with itself infinitely often, so the presence of cycles shows non-termination of a rewrite system.

The following is known:

PROPOSITION 4. $\mathbf{CL}(\mathbf{S})$ *admits no cycles.*

Proof (Bergstra and Klop [4]). All reductions in $\mathbf{CL}(\mathbf{S})$ increase the term size, except for $\mathbf{S}\,x\,y\,\mathbf{S} \to x\,\mathbf{S}\,(y\,\mathbf{S})$, but those change the shape of the term. ▮

Still a term rewriting system might have infinite reductions that are not cycles, but something more general:

DEFINITION 5. A *loop* is a reduction $X \to^{+} C[X\sigma]$, where X is a term possibly containing variables, $C[\cdot]$ is a context, and σ is a substitution.

Again a loop can be composed with itself, giving an infinite reduction. Yet infinite reductions in $\mathbf{CL}(\mathbf{S})$ do not seem to look like that:

CONJECTURE 6. $\mathbf{CL}(\mathbf{S})$ *admits no loops.*

For a general account of loops in rewriting, see [24]. For a treatment of cycles in combinatory logic, see [11].

In this section, we will prove a weaker version of Conjecture 6, namely that $\mathbf{CL}(\mathbf{S})$ does not admit *ground* loops.

DEFINITION 7. A *ground loop* is a reduction $X \to^{+} C[X]$.

A ground loop, composed with itself, would lead to an infinite reduction containing only finitely many different redexes. So the size of the redexes that are being reduced during this infinite reduction would be bounded by some constant. Yet we are going to show that for any fixed n, the restriction of $\mathbf{CL}(\mathbf{S})$ that only allows rewrites where the redex size does not exceed n is terminating.

In fact we are not bounding the *size* of the redexes but rather their *right depth* (length of their right spine). Moreover, we do not measure this depth exactly but use a safe approximation instead. This approximation is attached as a label to the application nodes and conservatively updated where reductions take place. Our method can be seen as a variant of *semantic labelling* as introduced by Zantema [23].

The labelled system $\mathbf{CL}_n(\mathbf{S})$ has the signature $\{\mathbf{S}, \circ_1, \circ_2, ..., \circ_n, \circ_\infty\}$, where $\mathbf{S}$ is as before, while the $\circ_i$ are labeled application nodes. There are $n+1$ of them. The rewrite rules of $\mathbf{CL}(\mathbf{S})$ are

$$\{((\mathbf{S} \circ_* x) \circ_* y) \circ_k z \to (x \circ_k z) \circ_{k+1} (y \circ_k z) \mid k \leqslant n\},$$

where $n+1=\infty$ and the $*$ match arbitrary numbers. Note that there is *no* rule that has $\circ_\infty$ as the top symbol of the left-hand side.

Next, we prove that $\mathbf{CL}_n(\mathbf{S})$ is terminating.

After that we derive how this carries over to the original $\mathbf{CL}(\mathbf{S})$.

THEOREM 8. *For all n, $\mathbf{CL}_n(\mathbf{S})$ is terminating.*

Proof. Order the signature by $\mathbf{S} > \circ_1 > \circ_2 > \cdots > \circ_n > \circ_\infty$.

This gives a recursive path order $>_{\mathrm{rpo}}$, which is a reduction order for $\mathbf{CL}_n(\mathbf{S})$, as can be easily verified:

We need to show that

$$L = ((\mathbf{S} \circ_* X) \circ_* Y) \circ_k Z >_{\mathrm{rpo}} (X \circ_k Z) \circ_{k+1} (Y \circ_k Z) = R$$

for all $k \leqslant n$. The top symbols of L and R are $\circ_k > \circ_{k+1}$.

We verify $L >_{\mathrm{rpo}} (X \circ_k Z)$ and $L >_{\mathrm{rpo}} (Y \circ_k Z)$. Here the top symbols coincide. We have to prove the multiset relations $\{(\mathbf{S} \circ_* X) \circ_* Y, Z\} \gg \{X, Z\}$ and $\{(\mathbf{S} \circ_* X) \circ_* X) \circ_* Y, Z\} \gg \{Y, Z\}$. They hold because of $(\mathbf{S} \circ_* X) \circ_* Y \rhd X$ as well as $(\mathbf{S} \circ_* X) \circ_* Y \rhd Y$. ▮

We are going to show that the labeling of the application nodes is preserved as a correct approximation (from below) of the right depth of the terms.

DEFINITION 9. The *right* (*resp. left*) *depth* of a term in $\mathbf{CL}(\mathbf{S})$ or $\mathbf{CL}_n(\mathbf{S})$ is

$$d_r(\mathbf{S}) = 0, \qquad d_r(X \circ_* Y) = 1 + d_r(Y)$$
$$d_l(\mathbf{S}) = 0, \qquad d_l(X \circ_* Y) = 1 + d_l(X).$$

DEFINITION 10. A labelled term X is called *consistent* iff for each subterm $X' \trianglelefteq X$ that has $\circ_k$ as its top symbol, $d_r(X') \geqslant k$

Obviously, subterms of consistent terms are consistent. Consistency is maintained by the reduction rules in $\mathbf{CL}_n(\mathbf{S})$.

PROPOSITION 11. *If X is consistent and $X \to^*_{\mathbf{CL}_n(\mathbf{S})} Y$, then Y is consistent.*

Proof. Consider one reduction step $X \xrightarrow{R} Y$ with $R = (\mathbf{S}\ B\ C) \circ_k D$.

Subterms of B, C, and D just get copied and remain consistent.

Subterms whose position is incomparable to the redex position are not affected at all.

Subterms lying on the path from the redex position to the root might increase their right depth due to the reduction, but they do not change their top label. So the approximation remains correct.

We consider terms $B \circ_k D$ and $C \circ_k D$ in the contractum. We know that the redex $R = * \circ_k D$ was consistent, so $k \leqslant d_r(R) = 1 + d_r(D)$. The attachment of the label k in $B \circ_k D$ is allowed because $d_r(B \circ D) = 1 + d_r(D) \geqslant k$. The same reasoning is valid for $C \circ_k D$. So $(B \circ_k D)$ and $(C \circ_k D)$ are consistent.

Finally $Z = (B \circ_k D) \circ_{k+1} (C \circ_k D)$ is consistent because $d_r(Z) = 1 + d_r(C \circ_k D) \geqslant 1 + k$. ▮

From an unlabeled term in $\mathbf{CL}(\mathbf{S})$, we can create a consistently labeled term in $\mathbf{CL}_n(\mathbf{S})$ (for large enough n) by just tagging each node:

Definition 12. The mapping

$$tag\colon \mathbf{CL}(\mathbf{S}) \to \mathbf{CL}_n(\mathbf{S})$$

replaces the top $\circ$ of each non-leaf subterm X with $\circ_{d_r(X)}$.

We might as well delete all labels:

Definition 13. The mapping

$$forget\colon \mathbf{CL}_n(\mathbf{S}) \to \mathbf{CL}(\mathbf{S})$$

replaces all $\circ_*$ with $\circ$.

Obviously, reductions in $\mathbf{CL}_n(\mathbf{S})$ can be unlabelled.

Proposition 14. *If $X \to_{\mathbf{CL}_n(\mathbf{S})} Y$ for some n, then $forget(X) \to_{\mathbf{CL}(\mathbf{S})} forget(Y)$.*

Can reductions in $\mathbf{CL}(\mathbf{S})$ be labelled? Yes, as long as the right depth of their redexes is bounded.

Proposition 15. *Assume there is a (finite or infinite) reduction*

$$X_1 \xrightarrow{R_1} X_2 \xrightarrow{R_2} X_3 \xrightarrow{R_3} \cdots$$

in $\mathbf{CL}(\mathbf{S})$. Assume that for all k we have $d_r(R_k) \leqslant n$. Then there is a reduction

$$X'_1 \xrightarrow{R'_1} X'_2 \xrightarrow{R'_2} X'_3 \xrightarrow{R'_3} \cdots$$

in $\mathbf{CL}_n(\mathbf{S})$ with $X'_1 = tag(X_1)$ and for all k, $forget(X'_k) = X_k$.

Proof. Each redex R'_k is a consistent subterm, by Proposition 11. That is why the top label of R'_k is less than or equal to $d_r(R'_k)$ but that is equal to $d_r(R_k) \leqslant n$. Therefore R'_k really was an $\mathbf{CL}_n(\mathbf{S})$-redex. ▮

Theorem 16. *Assume there is an infinite reduction*

$$X_1 \xrightarrow{R_1} X_2 \xrightarrow{R_2} X_3 \xrightarrow{R_3} \cdots$$

in $\mathbf{CL}(\mathbf{S})$. Then the sequence $d_r(R_k)$ is not bounded.

Proof. Otherwise, Proposition 15 would apply. Then all reductions could be placed in some $\mathbf{CL}_n(\mathbf{S})$. But this is a terminating system by Theorem 8, so the reduction chain could not be infinite. ▮

Corollary 17. $\mathbf{CL}(\mathbf{S})$ *admits no ground loops.*

Proof. A ground loop would give an infinite reduction chain with only finitely many different redexes, so their right depth would be bounded. This is impossible by Theorem 16. ▮

This method does not show the absence of non-ground loops because they could produce a sequence of redexes with unbounded right depth.

4. NORMALIZATION

In this part we are going to prove decidability of normalization of ground **S**-terms.

To decide whether an arbitrary **S**-term normalizes, we may assume that both children of the root already are in normal form. If they were not, we could first check these subterms recursively and compute their normal forms as long as they both existed.

The decision procedure tests membership of terms in rational tree languages. For a complete treatment of tree languages and tree automata, see [5, 7]. Here we just recall basic definitions and facts on rational tree languages:

A set of trees is called

— *regular* iff it is generated by a regular tree grammar,

— *recognizable* iff it is accepted by a finite deterministic bottom-up tree automaton,

— *rational* iff it is generated by a rational expression.

The above conditions are equivalent. Conversions between different representations are computable. Rational tree languages are closed under union, intersection, and complement. With the languages represented by finite automata, these operations are computable, and emptiness, finiteness, membership, and inclusion are decidable.

Now we will define some sets of ground **S**-terms that will be used frequently.

Definition 18. $\mathsf{T} = \mathbf{S}\,\mathbf{S}$, $\mathsf{A} = \mathsf{T}\,\mathbf{S} = \mathbf{S}\,\mathbf{S}\,\mathbf{S}$.

Definition 19. $\mathscr{M}$ denotes the set of *all* ground terms. $\mathscr{N}$ denotes the set of all ground terms in *normal* form.

These sets are rational. They are generated by the grammars

$$\mathscr{N} \to \mathbf{S} \cup \mathbf{S}\,\mathscr{N} \cup \mathbf{S}\,\mathscr{N}\mathscr{N}$$

$$\mathscr{M} \to \mathbf{S} \cup \mathscr{M}\mathscr{M}.$$

Inside an expression, we often just write $*$ for $\mathscr{M}$.

Definition 20. $\mathsf{S}_1 = \mathbf{S}$, for $k \geqslant 1 : \mathsf{S}_{k+1} = \mathbf{S}\,\mathsf{S}_k$. For $k \geqslant 0 : \mathscr{P}_k = \mathsf{S}_1 \cup \cdots \cup \mathsf{S}_k$. For $k \geqslant 0 : \mathscr{Q}_k = \mathscr{M} \backslash \mathscr{P}_k$.

The term S_k has a right spine of length $k-1$ and all its left children are **S**. The set $\mathscr{P}_k$ is the collection of all S_i up to k (while $\mathscr{P}_0 = \varnothing$). The complement of $\mathscr{P}_k$ is denoted by $\mathscr{Q}_k$. These are exactly the terms whose right spine has length $\geqslant k$ or which have some subterm with left spine longer than 1. For example, $\mathscr{Q}_0$ are all terms, $\mathscr{Q}_2$ are all terms except **S** and T, and $\mathsf{A} = \mathbf{S}\,\mathbf{S}\,\mathbf{S} \in \mathscr{Q}_k$ for any k.

From the above description we immediately derive

Lemma 21. *For all $k \geqslant 0$ we have $\mathscr{P}_k \subseteq \mathscr{P}_{k+1}$ and $\mathscr{Q}_k \supseteq \mathscr{Q}_{k+1}$.*

LEMMA 22. *If $X \in \mathscr{Q}_k$, and $X \lhd Y$, then $Y \in \mathscr{Q}_{k+1}$.*

Now we define an operation on languages.

DEFINITION 23. For a set of terms Y, the set of *Y-directors*, denoted $\langle Y \rangle$, is generated by the grammar

$$D \rightarrow Y \cup \mathbf{S}\, D\, \mathscr{M} \cup \mathbf{S}\, \mathscr{M}\, D.$$

The name has been chosen in analogy to *director strings* [10], because an argument Z of an Y-director can be *directed* towards Y:

PROPOSITION 24. $\langle Y \rangle\, Z \hookrightarrow\!\!\!\!\twoheadrightarrow YZ$.

Proof. The claim is proved by structural induction. Let $Y' \in \langle Y \rangle$.

1. $Y' = Y$. Then the claim is vacuously true.
2. $Y' \in \mathbf{S}\langle Y \rangle *$. Then $Y'Z \in \mathbf{S}\langle Y \rangle * Z \rightarrow \underline{(\langle Y \rangle Z)}(*Z) \rhd \langle Y \rangle\, Z$.
3. $Y' \in \mathbf{S} * \langle Y \rangle$. Then $Y'Z \in \mathbf{S} * \langle Y \rangle\, Z \rightarrow\, *Z\underline{(\langle Y \rangle\, Z)} \rhd \langle Y \rangle Z$. ▮

DEFINITION 25. For sets X and Y, the set (X/Y) is defined by the grammar

$$D \rightarrow Y \cup X\, D.$$

We immediately have

PROPOSITION 26. $(\mathbf{S}\, X/Y) \subset \langle Y \rangle$.

The following will be needed later. It is easily proved by structural induction.

PROPOSITION 27. $(\mathbf{S}\, X/Y)\, Z \twoheadrightarrow (X\, Z/Y\, Z)$.

Note that (X/Y) and $\langle X \rangle$ are rational whenever X and Y are.

We will construct subsets of $\mathscr{N}\mathscr{N}$ that contain terms with similar reduction properties. The construction begins by partitioning the set $\mathscr{N}$ itself into classes $\mathscr{N}_0$, $\mathscr{N}_1$, and $\mathscr{N}_2$, and then combines pairs of these. It turns out that $\mathscr{N}_0$ is responsible for finite reductions and $\mathscr{N}_2$ for infinite ones.

During the process, the case $\mathscr{N}_1\, \mathscr{N}_1$ needs special attention. Here, a further partition of $\mathscr{N}_1$ into sets $\mathscr{L}_0$, $\mathscr{L}_1$, and $\mathscr{L}_2$ is needed to completely analyze this case.

To make the case distinction complete, we finally consider the case $\mathscr{N}_2\, \mathscr{P}_2$ by showing that after some reductions it is.transformed into one of the cases that have already been dealt with.

These constructions have been found empirically, by looking at lots of examples and generalizing them. The verification of the claims is relegated to Appendix 1.

4.1. The Case $\mathscr{N}\mathscr{N}$

We will classify the set $\mathscr{N}$ of normal forms according to the existence of certain subterms that might or might not occur during the reduction of $X\, a$, where $X \in \mathscr{N}$ and a is a variable.

The most restricted class is

Definition 28. $\mathcal{N}_0 = (\mathsf{T}/\mathsf{S} \cup \mathsf{S}\,\mathcal{N})$.

Example 29. $\mathsf{A} = \mathsf{S}\,\mathsf{S}\,\mathsf{S} = \mathsf{T}\,\mathsf{S} \in (\mathsf{T}/\mathsf{S}) \subseteq \mathcal{N}_0$.

A term from $\mathcal{N}_0$ never moves its argument into functional position (it never *activates* it):

Example 30. $\mathsf{A}\,a = \mathsf{S}\,\mathsf{S}\,\mathsf{S}\,a \to \mathsf{S}\,a\,(\mathsf{S}\,a)$.

Proposition 31. *$\downarrow(\mathcal{N}_0\,a)$, and $\mathcal{N}_0 a \not\hookrightarrow\!\!\!\twoheadrightarrow a\,*$.*

Proof. See Appendix 1. ▮

A slightly more extended class is

Definition 32. $\mathcal{N}_{0,1} = (\mathsf{S}\,\mathcal{P}_2/\mathsf{S} \cup \mathsf{S}\,\mathcal{N} \cup \mathsf{S}(\mathsf{S}\,\mathsf{T})\,\mathcal{P}_2)$.

Example 33. $\mathsf{S}\,\mathsf{T}\,\mathsf{T} \in \mathsf{S}\,\mathcal{P}_2(\mathsf{S}\,\mathcal{N}) \subseteq (\mathsf{S}\,\mathcal{P}_2/\mathsf{S}\,\mathcal{N}) \subseteq \mathcal{N}_{0,1}$.

A term from $\mathcal{N}_{0,1}$ might activate its argument but supply it with at most one argument.

Example 34. $\mathsf{S}\,\mathsf{T}\,\mathsf{T}a \to \mathsf{T}\,a(\mathsf{T}\,a) \to \mathsf{S}(\mathsf{T}\,a)\underline{(a(\mathsf{T}\,a))}$.

Proposition 35. *$\downarrow(\mathcal{N}_{0,1}\,a)$, and $\mathcal{N}_{0,1} a \not\hookrightarrow\!\!\!\twoheadrightarrow a\,*\,*$.*

Proof. See Appendix 1. ▮

By definition we have $\mathcal{N}_0 \subseteq \mathcal{N}_{0,1}$. We introduce some more names:

Definition 36. $\mathcal{N}_1 = \mathcal{N}_{0,1} \setminus \mathcal{N}_0$; $\mathcal{N}_2 = \mathcal{N} \setminus \mathcal{N}_{0,1}$.

So $\mathcal{N}$ is the disjoint union of $\mathcal{N}_0$, $\mathcal{N}_1$, and $\mathcal{N}_2$. Recall that $\mathcal{N}_0$ *ignores* its argument, while $\mathcal{N}_1$ *activates* it *once*.

Essentially, $\mathcal{N}_0$ is responsible for normalization, and $\mathcal{N}_2$ for non-normalization:

Proposition 37. *The following table shows the normalization ($\downarrow$) resp. nonnormalization ($\uparrow$) of ground terms $X\,Y$, with X and Y already in normal form.*

	$Y \in \mathcal{N}_0$	$Y \in \mathcal{N}_1$	$Y \in \mathcal{N}_2$
$X \in \mathcal{N}_0$		$\downarrow$ *by Proposition* 57	
$X \in \mathcal{N}_1$	$\downarrow$ *by Proposition* 58	*see Section* 4.2	$\uparrow$ *by Proposition* 68
$X \in \mathcal{N}_2$	*see Section* 4.3	$\uparrow$ *by Proposition* 67	

Proof. See the propositions and proofs given in Appendix 1. ▮

4.2. The Case $\mathcal{N}_1\mathcal{N}_1$

We look deeper into the set $\mathcal{N}_1$. We single out the subset of terms that move their argument a into functional position but which only supply them with the argument $\mathsf{S}\,a$.

Definition 38. $\mathcal{L}_0 = (\mathsf{T}/\mathsf{S}\,(\mathsf{S}\,\mathcal{P}_2)\,\mathsf{S})$.

Proposition 39. *$\downarrow(\mathscr{L}_0\, a)$ and $\{Z\colon \mathscr{L}_0\, a \lhook\joinrel\twoheadrightarrow a\, Z\} = \{\mathsf{S}\, a\}$.*

Proof. See Appendix 1. ▮

Example 40. $\mathsf{S}\,(\mathsf{S}\,\mathsf{T})\,\mathsf{S}\,a \to \mathsf{S}\,\mathsf{T}\,a(\mathsf{S}\,a) \to \mathsf{T}\,(\mathsf{S}\,a)\,(a\,(\mathsf{S}\,a)) \to \mathsf{S}\,\underline{(a\,(\mathsf{S}\,a))}\,(\mathsf{S}\,a\,\underline{(a\,(\mathsf{S}\,a))})$.

Another useful subset of $\mathscr{N}_1$ is

Definition 41. $\mathscr{L}_1 = (\mathsf{T}/\mathsf{S}\,\mathsf{T}\,(\mathsf{S}\,\mathscr{P}_2))$.

It has the following property:

Proposition 42. *$\downarrow(\mathscr{L}_1 a)$, and $\{Z : \mathscr{L}_1 a \lhook\joinrel\twoheadrightarrow a\, Z\} = \{\mathsf{S}\,\mathscr{P}_2\, a\}$.*

Proof. See Appendix 1. ▮

Example 43. $\mathsf{S}\,\mathsf{T}\,(\mathsf{S}\,\mathsf{T})\,a \to \mathsf{T}\,a\,(\mathsf{S}\,\mathsf{T}\,a) \to \mathsf{S}\,(\mathsf{S}\,\mathsf{T}\,a)\underline{(a\,(\mathsf{S}\,\mathsf{T}\,a))}$.

It would seem more intuitive to take Proposition 42 as definition and to derive Definition 41 from that, and similarly for Proposition 39 and Definition 38. But the given presentation is more compact.

Definition 44. $\mathscr{L}_{1,2} = \mathscr{N}_1 \backslash \mathscr{L}_0$; $\mathscr{L}_2 = \mathscr{L}_{1,2} \backslash \mathscr{L}_1$.

So $\mathscr{N}_1$ is the disjoint union of $\mathscr{L}_0$, $\mathscr{L}_1$, and $\mathscr{L}_2$.

Proposition 45. *The following table shows the normalization ($\downarrow$) resp. non-normalization ($\uparrow$) of ground terms $X\,Y$, with X and Y from the set $\mathscr{N}_1$.*

	$Y \in \mathscr{L}_0$	$Y \in \mathscr{L}_1 \cup \mathscr{L}_2 = \mathscr{L}_{1,2}$
$X \in \mathscr{L}_0$	$\downarrow$ *by Proposition* 69	
$X \in \mathscr{L}_1$	$\downarrow$ *by Proposition* 70	$\uparrow$ *by Proposition* 71
$X \in L_2$	$\uparrow$ *by Proposition* 72	

Proof. See propositions and proofs given in the Appendix. ▮

4.3. The Case $\mathscr{N}_2\, \mathscr{N}_0$

We have $\mathscr{N}_0 \subseteq \mathscr{N} \subseteq \mathscr{P}_2 \cup \mathscr{Q}_2$. By Proposition 67 (see Appendix 1), $\mathscr{N}_2\, \mathscr{Q}_2 \twoheadrightarrow \infty$. Together with Propositions 37 and 45, this gives

Proposition 46. *It is decidable whether a term from $\mathscr{N}\,\mathscr{Q}_2$ has a normal form.*

So we are left with the case $\mathscr{N}_2\, \mathscr{P}_2$. Here we start any innermost reduction and continue until Proposition 46 can be used. We exploit the fact that redex sizes (again we actually use their right depths) eventually increase. This will be made precise now.

Definition 47. A reduction $X \xrightarrow{R} X'$ is called *innermost*, iff the redex R has no proper subterm that is itself a redex.

We highlight subterm positions that are important for innermost reductions. We call a term *active* iff its left depth is at least 3 and its left child has a normal form.

Definition 48. The set *act* (X) of active subterms of a term X is

$$\mathrm{act}(X) = \{\, Y : B\,C = Y \trianglelefteq X,\ d_l(Y) \geqslant 3,\ \downarrow(B)\}.$$

Here, and in the remainder of this section, we deliberately confuse subterms and their positions.

Proposition 49. *If* $X \xrightarrow{R} X'$ *with* R *innermost, then* $R \in \mathrm{act}(X)$, *but no proper subterm of* R *is* $\in \mathrm{act}(X)$.

For active positions Y with fixed right child C, we collect information on the left children B:

Definition 50. The *weight* of C in X is

$$\mathrm{weight}_C(X) = \sum \{\, |\mathrm{nf}(B)| : B\,C = Y \in \mathrm{act}(X)\}.$$

We set $\sum \varnothing = 0$; furthermore nf (B) denotes the normal form of B, and $|C|$ is the size of C, defined by $|\mathbf{S}| = 1$, $|D\,E| = |D| + |E|$.

Because we are dealing with $\mathcal{N}\,\mathcal{P}_2$, we are only interested in the weights of $\mathbf{S}$ and T:

Definition 51. The *total weight* of X is the pair

$$\mathrm{weight}(X) = (\mathrm{weight}_{\mathbf{S}}(X), \mathrm{weight}_{\mathsf{T}}(X)).$$

During innermost reductions, the total weight decreases lexicographically.

Proposition 52. *If* $X \xrightarrow{R} X'$ *with* $R = \mathbf{S}\,D\,E\,F$ *innermost, then* $\mathrm{weight}(X) \geqslant_{\mathrm{lex}} \mathrm{weight}(X')$ *with strict inequality for* $F \in \mathcal{P}_2$.

Proof. See Appendix 1. ∎

Corollary 53. *For any n, an infinite innermost reduction reduces only finitely many redexes* $R \in \mathcal{N}\,\mathcal{P}_2$.

This is, essentially, a statement about the minimum right depth of redexes during innermost reductions. Compare this with the situation in Theorem 16. This was a statement about the *maximum* right depth of the redexes, in arbitrary reductions. The assumption of innermost reduction cannot be dropped in Corollary 53.

In an infinite innermost reduction, eventually all redexes are of the form $\mathcal{N}\,\mathcal{Q}_2$. At least one of them has no normal form.

Proposition 54. *It is decidable whether a term from* $\mathcal{N}\,\mathcal{P}_2$ *has a normal form.*

Proof. Call the term X. Perform any innermost reduction sequence, starting from X. For each redex that is reduced, and that has the form $\mathcal{N}\,\mathcal{Q}_2$, check whether it has a normal form, using Proposition 46. By the previous argument, this sequence either stops (because X normalizes) or contains a redex in $\mathcal{N}\,\mathcal{Q}_2$ without a normal form (then X itself has no normal form). ∎

4.4. Deciding Normalization

THEOREM 55. *There is a procedure that decides whether a ground term in* **CL**(**S**) *has a normal form.*

Proof. **S** has a normal form. Therefore assume the expression is $X\,Y$.

Recursively apply the procedure to X and Y and compute their normal forms if they exist. This recursion is terminating because it is structural. If one (or both) of the normal forms does not exist, halt and answer not *normalizing*. Otherwise call them X' and Y', respectively. If $Y' \in \mathcal{Q}_2$, apply Proposition 46. If $Y' \in \mathcal{P}_2$, use Proposition 54. ▌

5. RATIONALITY

The procedure that decides normalization of **S**-terms checks membership in rational sets. This does not directly imply that the set of normalizing **S**-terms is itself rational because the procedure also performs some intermediate reductions. Perhaps surprisingly, the following nevertheless holds:

THEOREM 56. *The set of normalizing terms in* **CL**(**S**) *is rational.*

Appendix 2 contains a grammar that produces the set of normalizing **S**-terms. It has been found by manual completion w.r.t. backward reduction in **CL**(**S**), starting from the normalizing subsets of $\mathcal{N}\mathcal{N}$, obtained from Propositions 37 and 45.

By a tedious case analysis it can be checked that this grammar indeed produces normalizing terms. On the other hand it is more difficult to verify that it indeed produces *all* normalizing terms, or equivalently, that it generates $\mathcal{N}$, and is closed w.r.t. backwards application of the **S** rule.

To check this, one verifies that each non-terminal that produces a term $X\,Z(Y\,Z)$ also produces the term **S** $X\,Y\,Z$. Translated to automata this reads: for all states x, y, z, a, b, c with transitions $x\,z \to a$, $y\,z \to b$, $a\,b \to c$, there must be states p, q and transitions **S** $x \to p$, $p\,y \to q$, $q\,y \to c$. Because of the **S** rule being non-linear, this only works for a deterministic automaton (we have to make sure that both Zs in $X\,Z(Y\,Z)$ are accepted in the same state).

Indeed the computer program `RX` [21] is able to make the given grammar deterministic (the result has 43 states and 1600 transitions) and to check its backward closure.

6. CONCLUSION

Apparently, a seminar held by Barendregt, Bergstra, Klop, and Volken in 1975 started the detailed study of **S**-terms. They placed a bet on the existence of terms without normal form, and Barendregt collected 25 guilders from the other three by producing A A A (and later **S** A A (**S** A A)). Later Duboué found **S** T **S S S S** (and its reduct A T **S S**) to be the smallest **S**-terms that do not normalize.

The decidability of normalization had already been conjectured by Zachos in 1978. He completely analyzed the normalization of **S**-terms of size up to 9, by either finding the normal form or exhibiting patterns (similar to those given in the Examples 1 and 2) that lead to an infinite reduction.

Still a proof that each non-normalizing term admits such a pattern would be desirable. (It does not follow directly from the results presented here. The main problem is that the case distinctions in the proofs of Propositions 61 and 63 are not "constructive.")

One would also like to be able to automatically derive the grammar for non-normalizing **S**-terms, just starting from the grammar for $\mathcal{N}$, and the **S** rule. There are a number of completion and approximation results (see [8, 9]), but none of them applies to **CL**(**S**) because the **S** rule is neither linear nor shallow.

The combinator **L** with reduction rule $\mathbf{L}\,x\,y \to x\,(y\,y)$ is called the *lark* in Smullyan's book [16]. It has been investigated in depth by Statman [18] and Sprenger and Wymann-Böni [17]. Convertibility is decidable for **CL**(**L**). I conjecture that this should also be decidable for **S** (the *starling*) but it might be substantially more difficult than for the lark: If deciding normalization is a step toward deciding convertibility, then this first step is trivial for the lark, but, as we have seen, quite hard for the starling.

I proved [19] that **CL**(**S**) is *top-terminating*, meaning that, even in infinite reductions, the root of a term is rewritten only finitely often. Top-termination guarantees the existence of limits (of fair infinite reductions). They seem to admit finite descriptions that could be used to decide convertibility of non-normalizing terms.

This paper's introductory question was "does the difficulty of **CL**(**S**, **K**) come from **S**?" The answer is "no," in a sense. We have seen that **CL**(**S**) certainly is complicated, but still manageable (by tree automata). The important step from here to undecidability is caused by the addition of the **K** combinator. So the (non-disjoint) union of **CL**(**S**) and **CL**(**K**) is a vastly more powerful system than each of its constituents.

This paper also shows that the seemingly small **S** rule is far from being completely understood. It is a challenging test case for the application of known results on tree automata and term rewriting and a motivation to develop new techniques that would better explain the results obtained so far, by putting them in a more general framework.

APPENDIX 1: PROOFS

At a few places during the presentation (in the proofs of Propositions 67, 71, and 72) we rely on inclusion/exclusion relations between certain rational tree languages. They could be verified by pen and paper as well, but we employed the computer program `RX` [21] that performs the standard operations on rational tree languages (represented by finite automata).

PROPOSITION 31. *$\downarrow(\mathcal{N}_0\, a)$ and $\mathcal{N}_0 \not\hookrightarrow\!\!\twoheadrightarrow a\, *$.*

Proof. By Proposition 27, $\mathcal{N}_0\, a = (\mathsf{T}/\mathbf{S} \cup \mathbf{S}\,\mathcal{N})\, a \twoheadrightarrow (\mathbf{S}\, a/\mathbf{S}\, a \cup \mathbf{S}\,\mathcal{N}\; a)$ and no further reductions can happen. We see that a always is the right child of its parent. ▮

Proposition 57. $\downarrow(\mathcal{N}_0\,\mathcal{N})$.

Proof. The normal form of $\mathcal{N}_0\,\mathcal{N}$ is that of $\mathcal{N}_0\,a$ with a replaced by $\mathcal{N}$. No reductions can happen inside $\mathcal{N}$, because it already is a normal form. No new redexes can be created. ▮

Proposition 35. *$\downarrow(\mathcal{N}_{0,1}\,a)$ and $\mathcal{N}_{0,1}a \not\hookrightarrow\!\!\!\!\twoheadrightarrow a * *$. Moreover, if $\mathcal{N}_{0,1}a \hookrightarrow\!\!\!\!\twoheadrightarrow a\,X$, then $a \lhd X$.*

Proof. The base cases are

1. $\mathsf{S}\,a$ is in normal form,
2. $\mathsf{S}\,\mathcal{N}\,a$ is in normal form,
3. $\mathsf{S}\,(\mathsf{S}\,\mathsf{T})\,\mathsf{S}\,a \rightarrow \mathsf{S}\,\mathsf{T}\,a\,(\mathsf{S}\,a) \rightarrow \mathsf{T}\,(\mathsf{S}\,a)(a\,(\mathsf{S}\,a)) \rightarrow \mathsf{S}\,\underline{(a\,(\mathsf{S}\,a))}(\mathsf{S}\,a\,\underline{(a\,(\mathsf{S}\,a))})$,
4. $\mathsf{S}\,(\mathsf{S}\,\mathsf{T})\,\mathsf{T}\,a \rightarrow \mathsf{S}\,\mathsf{T}\,a\,(\mathsf{T}\,a) \rightarrow \mathsf{T}\,(\mathsf{T}\,a)(a\,(\mathsf{T}\,a)) \rightarrow \mathsf{S}\,(a\,(\mathsf{T}\,a))(\mathsf{T}\,a\,(a\,(\mathsf{T}\,a))) \rightarrow \mathsf{S}\,\underline{(a\,(\mathsf{T}\,a))}(\mathsf{S}\,\underline{(a\,(\mathsf{T}\,a))}\underline{(a\,(a\,(\mathsf{T}\,a)))})$,

where the claim can be verified at the underlined subterms.

The inductive steps are

1. $\mathsf{S}\,\mathsf{S}\,\mathcal{N}_{0,1}\,a \rightarrow \mathsf{S}\,a\,\underline{(\mathcal{N}_{0,1}\,a)}$,
2. $\mathsf{S}\,\mathsf{T}\,\mathcal{N}_{0,1}\,a \rightarrow \mathsf{T}\,a\,(\mathcal{N}_{0,1}\,a) \rightarrow \mathsf{S}\,\underline{(\mathcal{N}_{0,1}\,a)(a(\mathcal{N}_{0,1}a))}$ ▮

Proposition 58. $\downarrow(\mathcal{N}_{0,1}\,\mathcal{N}_0)$.

Proof. First reduce $\mathcal{N}_{0,1}\,\mathcal{N}_0$ as if $\mathcal{N}_0$ were a variable. By Proposition 35, the only redexes that might arise are of the form $\mathcal{N}_0\,*$. They might be nested, but all of them can be reduced to normal form by Proposition 57. ▮

Definition 59. $\mathcal{F} = \langle \mathcal{Q}_2\,\mathcal{Q}_1 \rangle$.

Lemma 60. $\mathcal{Q}_1\,\mathcal{F} \subseteq \mathcal{F}$.

Proof. Let $X\,Y \in \mathcal{Q}_1\,\mathcal{F}$. We use the fact that $\mathcal{Q}_1 = \mathsf{T} \cup \mathcal{Q}_2$.

1. $X = \mathsf{T}$. Then $X\,Y \in \mathsf{S} * \mathcal{F} \subseteq \langle \mathcal{F} \rangle \subseteq \mathcal{F}$.
2. $X \in \mathcal{Q}_2$. As $Y \in \mathcal{F} \subseteq \mathcal{Q}_1$, we have $X\,Y \in \mathcal{Q}_2\,\mathcal{Q}_1 \subseteq \mathcal{F}$. ▮

Proposition 61. $\langle \mathcal{Q}_2\,\mathcal{Q}_1 \rangle \langle \mathcal{Q}_2\,\mathcal{Q}_1 \rangle \twoheadrightarrow \infty$.

Example 62. $\mathsf{S}\,\mathsf{T}\,\mathsf{T}$ is the smallest term in $\langle \mathcal{Q}_2\,\mathcal{Q}_1 \rangle$. We have $\mathsf{S}\,\mathsf{T}\,\mathsf{T}\,(\mathsf{S}\,\mathsf{T}\,\mathsf{T}) \twoheadrightarrow \infty$, as seen already in Example 1.

Proof. We have $\langle \mathcal{Q}_2\,\mathcal{Q}_1 \rangle \langle \mathcal{Q}_2\,\mathcal{Q}_1 \rangle \hookrightarrow\!\!\!\!\twoheadrightarrow \mathcal{Q}_2\,\mathcal{Q}_1\,\mathcal{F}$. Let $X\,Y\,Z \in \mathcal{Q}_2\,\mathcal{Q}_1\,\mathcal{F}$.
If $X \twoheadrightarrow \infty$, then the $XYZ \twoheadrightarrow \infty$. So assume X has a normal form. It could have shape $\mathsf{S}\,*$ or $\mathsf{S} * *$.

1. $X \twoheadrightarrow \mathsf{S}\,\mathcal{Q}_1$.

$\mathsf{S}\,\mathcal{Q}_1\,Y\,\mathcal{F} \rightarrow \mathcal{Q}_1\,\mathcal{F}\,(Y\,\mathcal{F}) \subseteq \mathcal{Q}_1\,\mathcal{F}\,(\mathcal{Q}_1\,\mathcal{F}) \subseteq \mathcal{F}\,\mathcal{F} \twoheadrightarrow \infty$ by applying Lemma 60 twice.

2. $X \twoheadrightarrow \mathbf{S} * *$.

$\mathbf{S} * * \mathcal{Q}_1 \mathcal{F} \to * \mathcal{Q}_1(* \mathcal{Q}_1) \mathcal{F}$. By Lemmata 22 and 21 we have $* \mathcal{Q}_1 \subseteq \mathcal{Q}_2 \subseteq \mathcal{Q}_1$, so $* \mathcal{Q}_1(* \mathcal{Q}_1) \mathcal{F} \subseteq \mathcal{Q}_2 \mathcal{Q}_1 \mathcal{F} \subseteq \mathcal{F}\ \mathcal{F} \twoheadrightarrow \infty$.

In either case we find an infinite reduction because we have done at least one reduction step and the result again has the form $\mathcal{F}\ \mathcal{F}$. ▮

Proposition 63. $\langle \mathcal{Q}_3\, \mathcal{Q}_2 \rangle \mathcal{Q}_1 \twoheadrightarrow \infty$.

Example 64. $\mathsf{A}\ \mathsf{A}\ \mathsf{A} \in \mathcal{Q}_3\ \mathcal{Q}_1$, so $\mathsf{A}\ \mathsf{A}\ \mathsf{A} \twoheadrightarrow \infty$. See Example 2.

Proof. Let $\langle \mathcal{Q}_3 \mathcal{Q}_2 \rangle\ \mathcal{Q}_1 \lhook\joinrel\twoheadrightarrow \mathcal{Q}_3\ \mathcal{Q}_2\ \mathcal{Q}_1$, and let $X\ Y\, Z \in \mathcal{Q}_3\ \mathcal{Q}_2\ \mathcal{Q}_1$.
If $X \twoheadrightarrow \infty$, then the $X\ Y\, Z \twoheadrightarrow \infty$. So assume X has a normal form. It could be $\mathbf{S}\ \mathcal{Q}_2$ or $\mathbf{S} * *$.

1. $X \twoheadrightarrow \mathbf{S}\ \mathcal{Q}_2$.

$\mathbf{S}\ \mathcal{Q}_2\ \mathcal{Q}_2\ \mathcal{Q}_1 \to \mathcal{Q}_2\ \mathcal{Q}_1\ (\mathcal{Q}_2\ \mathcal{Q}_1) \subseteq \mathcal{F}\ \mathcal{F} \twoheadrightarrow \infty$ by Proposition 61

2. $X \twoheadrightarrow \mathbf{S} * *$.

$\mathbf{S} * * \mathcal{Q}_2\ \mathcal{Q}_1 \to * \mathcal{Q}_2\ (* \mathcal{Q}_2)\ \mathcal{Q}_1$ By Lemmata 22 and 21 we have $* \mathcal{Q}_2 \subseteq \mathcal{Q}_3 \subseteq \mathcal{Q}_2$, so $* \mathcal{Q}_2\ (* \mathcal{Q}_2)\ \mathcal{Q}_1 \subseteq \mathcal{Q}_3\ \mathcal{Q}_2\ \mathcal{Q}_1 \twoheadrightarrow \infty$ by induction. ▮

Proposition 65. $\langle \mathbf{S}\ \mathcal{Q}_3\ \mathcal{Q}_0 \rangle\ \mathcal{Q}_2 \twoheadrightarrow \infty$.

Example 66. $\mathbf{S}\,\mathsf{A}\,\mathbf{S}\,(\mathbf{S}\,\mathsf{A}\,\mathbf{S}) \in \langle \mathbf{S}\ \mathcal{Q}_3\ \mathcal{Q}_0 \rangle\ \mathcal{Q}_2$. Its reduction graph is a pure line, and all reductions are head reductions.

Proof. $\langle \mathbf{S}\ \mathcal{Q}_3\ \mathcal{Q}_0 \rangle\ \mathcal{Q}_2 \lhook\joinrel\twoheadrightarrow \mathbf{S}\ \mathcal{Q}_3\ \mathcal{Q}_0\ \mathcal{Q}_2 \to \mathcal{Q}_3\ \mathcal{Q}_2\ (\mathcal{Q}_0\ \mathcal{Q}_2) \subseteq \mathcal{Q}_3\ \mathcal{Q}_2\ \mathcal{Q}_1$. Then apply Proposition 63. ▮

Proposition 67. $\mathcal{N}_2\ \mathcal{Q}_2 \twoheadrightarrow \infty$.

Proof. It can be shown that $\mathcal{N}_2 \subseteq \langle \mathcal{Q}_3\ \mathcal{Q}_2 \rangle \cup \langle \mathbf{S}\ \mathcal{Q}_3\ \mathcal{Q}_0 \rangle$. (See the introductory remark on verification of such relations.) This is exactly the right form to apply Propositions 63 and 65. ▮

Proposition 68. $\mathcal{N}_1\ \mathcal{N}_2 \twoheadrightarrow \infty$.

Proof. Let $X\ Y \in \mathcal{N}_1\ \mathcal{N}_2$. There is a reduction from $X\ Y$ that activates Y, i.e., that produces a subterm $Y Z$. By the second part of Proposition 35, we have $Y \lhd Z$. Since $Y \in \mathcal{N}_2 \subseteq \mathcal{Q}_1$, clearly $Z \in \mathcal{Q}_2$. So we have $Y Z \in \mathcal{N}_2\ \mathcal{Q}_2$ and can apply Proposition 67. ▮

Proposition 39. *$\downarrow (\mathcal{L}_0\, a)$, and $\{Z : \mathcal{L}_0\, a \lhook\joinrel\twoheadrightarrow a\, Z\} = \{\mathbf{S}\, a\}$.*

Proof. The base cases are

1. $\mathbf{S}\,\mathsf{T}\,\mathbf{S}\, a \to \mathsf{T}\, a\, (\mathbf{S}\, a) \to \mathbf{S}\, (\mathbf{S}\, a) \underline{(a\, (\mathbf{S}\, a))}$,
2. $\mathbf{S}\, (\mathbf{S}\,\mathsf{T})\, \mathbf{S}\, a \to \mathbf{S}\,\mathsf{T}\, a\, (\mathbf{S}\, a) \to \mathsf{T}\, (\mathbf{S}\, a)(a\, (\mathbf{S}\, a)) \to \underline{(a\, (\mathbf{S}\, a))}(\mathbf{S}\, a\, \underline{(a\, (\mathbf{S}\, a))})$.

The inductive step reads $\mathsf{T}\, \mathcal{L}_0\, a \to \mathbf{S}\, a \underline{(\mathcal{L}_0 a)}$. ▮

Proposition 69. $\downarrow (\mathcal{L}_0\, \mathcal{N}_1)$.

Proof. Reduce $\mathscr{L}_0\,\mathscr{N}_1$ as if $\mathscr{N}_1$ were a variable. By Proposition 39, the only redexes that may arise have the form $\mathscr{N}_1(\mathbf{S}\,\mathscr{N}_1)$. But $\mathbf{S}\,\mathscr{N}_1 \subseteq \mathbf{S}\,\mathscr{N} \subseteq \mathscr{N}_0$, so $\mathscr{N}_1\,(\mathbf{S}\,\mathscr{N}_1) \subseteq \mathscr{N}_1\,\mathscr{N}_0$, and therefore they normalize, by Proposition 58. ▮

Proposition 42. *$\downarrow(\mathscr{L}_1\,a)$ and $\{Z\colon \mathscr{L}_1\,a \lhook\joinrel\twoheadrightarrow a\,Z\} = \{\mathbf{S}\,\mathscr{P}_2\,a\}$.*

Proof. The base case is $\mathbf{S}\,\mathsf{T}\,(\mathbf{S}\,\mathscr{P}_2)\,a \to \mathsf{T}\,a\,(\mathbf{S}\,\mathscr{P}_2\,a) \to \mathbf{S}\,(\mathbf{S}\,\mathscr{P}_2\,a)\underline{(a\,(\mathbf{S}\,\mathscr{P}_2\,a))}$. The inductive step is $\mathsf{T}\,\mathscr{L}_1\,a \to \mathbf{S}\,a\,\underline{(\mathscr{L}_1\,a)}$. ▮

Proposition 70. *$\downarrow(\mathscr{L}_1\,\mathscr{L}_0)$.*

Proof. Reduce $\mathscr{L}_1\,\mathscr{L}_0$, treating $\mathscr{L}_0$ as a variable. By Proposition 42, the only new redexes have the form $\mathscr{L}_0\,(\mathbf{S}\,\mathscr{P}_2\,\mathscr{L}_0)$. We have $\mathscr{L}_0 \subseteq \mathscr{N}_1$; therefore $\mathbf{S}\,\mathscr{P}_2\,\mathscr{L}_0 \subseteq \mathbf{S}\,\mathscr{P}_2\,\mathscr{N}_1 \subseteq \mathscr{N}_1$ by definition. But then $\mathscr{L}_0\,(\mathbf{S}\,\mathscr{P}_2\,\mathscr{L}_0) \subseteq \mathscr{L}_0\,\mathscr{N}_1$, which normalizes by Proposition 69. ▮

Proposition 71. *$\mathscr{L}_{1,2}\,\mathscr{L}_{1,2} \twoheadrightarrow \infty$.*

Proof. We have $\mathscr{L}_{1,2} \subseteq \langle \mathscr{Q}_2\,\mathscr{Q}_1\rangle$. (See the introductory remark on verification of such relations.) So Proposition 61 is applicable. ▮

Proposition 72. *$\mathscr{L}_2\,\mathscr{N}_1 \twoheadrightarrow \infty$.*

Proof. We use the fact that $\mathscr{L}_2 \subseteq (\mathsf{T}/\mathbf{S}\,(\mathbf{S}\,\mathsf{T})\,\mathsf{T} \cup \mathbf{S}\,\mathsf{T}\,\mathscr{Q}_3)$. (See the introductory remark on verification.) Also, for any k, we have $\mathscr{N}_1 \subseteq \mathscr{Q}_k$, because $\mathscr{N}_1 \cap \mathscr{P}_k = \varnothing$ since $\mathscr{P}_k \subseteq \mathscr{N}_0$. Then we have two base cases:

1. $\mathbf{S}\,\mathsf{T}\,\mathscr{Q}_3\,\mathscr{N}_1 \to \mathsf{T}\,\mathscr{N}_1\,(\mathscr{Q}_3\,\mathscr{N}_1) \to \mathbf{S}\,(\mathscr{Q}_3\,\mathscr{N}_1)\underline{(\mathscr{N}_1\,(\mathscr{Q}_3\,\mathscr{N}_1))}$. By Proposition 35, applied to the underlined expression, there is a reduction $\mathscr{N}_1\,(\mathscr{Q}_3\,\mathscr{N}_1) \lhook\joinrel\twoheadrightarrow \mathscr{Q}_3\,\mathscr{N}_1\,X$ with $\mathscr{Q}_3\mathscr{N}_1 \lhd X$, so that $X \subseteq \mathscr{Q}_2$. Moreover we have $\mathscr{N}_1 \subseteq \mathscr{Q}_2$, therefore the $\mathscr{Q}_3\,\mathscr{N}_1\,X \subseteq \mathscr{Q}_3\,\mathscr{Q}_2\,\mathscr{Q}_1 \twoheadrightarrow \infty$ by Proposition 63.

2. $\mathbf{S}(\mathbf{S}\,\mathsf{T})\,\mathsf{T}\,\mathscr{N}_1 \to \mathbf{S}\,\mathsf{T}\,\mathscr{N}_1(\mathsf{T}\,\mathscr{N}_1) \subseteq \mathbf{S}\,\mathsf{T}\,\mathscr{Q}_3\,\mathscr{N}_1$ because $\mathscr{N}_1 \subseteq \mathscr{Q}_3$. This gets us back to the previous case.

The inductive step is just $\mathsf{T}\,\mathscr{L}_2\,\mathscr{N}_1 \to \mathbf{S}\,\mathscr{N}_1\,\underline{(\mathscr{L}_2\,\mathscr{N}_1)}$. ▮

Proposition 52. *If $X \xrightarrow{R} X'$ with $R = \mathbf{S}\,B\,C\,D$ innermost, then weight $(X) \geqslant_{\text{lex}}$ weight (X') with strict inequality for $F \in \mathscr{P}_2$.*

Proof. Denote the contractum of $R = \mathbf{S}\,B\,C\,D$ by $R' = B\,D\,(C\,D)$. Consider an active subterm $E\,F = Y \trianglelefteq X$. Call $Y' = E'F'$ the subterm of X' that is at the same position as Y in X. We list the relative positions of Y and R:

1. None is a subterm of the other. Then weight $(Y) =$ weight(Y').

2. $Y = R$. The left child of R is $\mathbf{S}\,B\,C$. In R', we possibly have active subterms $B\,D$ and $C\,D$, but $|\mathbf{S}\,B\,C| > |B| + |C|$.

Additionally, we might create an active subterm $B\,D(C\,D)$. This could (in case $C = D = \mathbf{S}$) affect the second component of weight (X'). But this is irrelevant for the lexicographic ordering, since the first component strictly decreases.

3. Y is a proper subterm of R. Impossible by Proposition 49.

4. R is a subterm of the left child E of Y. The reduction does not change the size of the normal form: $|E| = |E'|$

5. R is a subterm of the right child F of Y. Then F is not in $\mathscr{P}_2$, so the weight is not affected.

Adding this for all possible Y, the first claim is proved. Since case 2 happens exactly once, strict inequality is proved. ▮

APPENDIX 2: GRAMMAR

The set of (many step) predecessors of a set Z is denoted by

Definition 73. $\operatorname{pred}(Z) = \{X : X \twoheadrightarrow Z\}$.

We are also interested in the set of all terms X that reduce to a term in Z when supplied with an argument from Y.

Definition 74. $\operatorname{pred}_Y(Z) = \{X : XY \twoheadrightarrow Z\}$.

In this section, we will give a grammar for $\operatorname{pred}(\mathscr{N})$, the set of normalizing **S**-terms. From Propositions 37 and 45 we derive

$$\begin{aligned}\operatorname{pred}(\mathscr{N}) \to \mathbf{S} &\cup \operatorname{pred}(\mathscr{N}_0)\operatorname{pred}(\mathscr{N}) \cup \operatorname{pred}(\mathscr{L}_0)\operatorname{pred}(\mathscr{N}_{0,1})\\ &\cup \operatorname{pred}(\mathscr{L}_1)\operatorname{pred}(\mathscr{L}_0) \cup \operatorname{pred}(\mathscr{N}_{0,1})\operatorname{pred}(\mathscr{N}_0)\\ &\cup \operatorname{pred}_{\mathbf{S}}(\mathscr{N})\,\mathbf{S} \cup \operatorname{pred}_{\mathsf{T}}(\mathscr{N})\,\mathsf{T}.\end{aligned}$$

Here and later, all expressions $\operatorname{pred}(Z)$ and $\operatorname{pred}_Y(Z)$ are considered as different non-terminals of the grammar. Its productions are

$$\begin{aligned}\operatorname{pred}(\mathscr{N}_0) &\to \mathbf{S} \cup \mathbf{S}\operatorname{pred}(\mathscr{N}) \cup \mathsf{T}\operatorname{pred}(\mathscr{N}_0)\\ &\quad\cup \operatorname{pred}_{\mathbf{S}}(\mathscr{N}_0)\,\mathbf{S} \cup \operatorname{pred}_{\mathsf{T}}(\mathscr{N}_0)\,\mathsf{T}\\ \operatorname{pred}(\mathscr{N}_{0,1}) &\to \mathbf{S} \cup \mathbf{S}\operatorname{pred}(\mathscr{N}) \cup \mathbf{S}(\mathbf{S}\,\mathsf{T})\,\mathscr{P}_2 \cup \mathbf{S}\mathscr{P}_2\operatorname{pred}(\mathscr{N}_{0,1})\\ &\quad\cup \operatorname{pred}_{\mathbf{S}}(\mathscr{N}_{0,1})\,\mathbf{S} \cup \operatorname{pred}_{\mathsf{T}}(\mathscr{N}_{0,1})\,\mathsf{T}\\ \operatorname{pred}_{\mathbf{S}}(\mathscr{L}_0) &\to (\mathsf{T}/\mathbf{S}(\mathbf{S}\,\mathscr{P}_2))\\ \operatorname{pred}(\mathscr{L}_0) &\to (\mathsf{T}/\operatorname{pred}_{\mathbf{S}}(\mathscr{L}_0)\,\mathbf{S})\\ \operatorname{pred}_{\mathsf{T}}(\mathscr{L}_1) &\to \mathbf{S}\,\mathsf{T} \cup \mathsf{A}\\ \operatorname{pred}_{\mathbf{S}}(\mathscr{L}_1) &\to (\mathsf{T}/(\mathsf{T}/\mathbf{S}\mathsf{T})\,\mathbf{S})\\ \operatorname{pred}(\mathscr{L}_1) &\to (\mathsf{T}/\mathbf{S}\,\mathsf{T}\,(\mathbf{S}\,\mathsf{T}) \cup \operatorname{pred}_{\mathsf{T}}(\mathscr{L}_1)\,\mathsf{T} \cup \operatorname{pred}_{\mathbf{S}}(\mathscr{L}_1)\,\mathbf{S}).\end{aligned}$$

It is interesting to note that the set

$$\mathscr{N}_{00} \to \mathbf{S} \cup \mathsf{T}\,\mathscr{N}_{00} \cup \mathscr{N}_{00}\,\mathbf{S}$$

is closed w.r.t. forward and backward reduction.

$$\mathrm{pred}_{\mathsf{T}}(\mathscr{N}_0) \to \mathscr{P}_2;\ \mathrm{pred}_{\mathbf{S}}(\mathscr{N}_0) \to \mathscr{N}_{00}$$

$$\mathrm{pred}_{\mathsf{T}}(\mathscr{N}_{0,1}) \to \mathscr{N}_{00} \cup \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_0)$$

$$\mathrm{pred}_{\mathbf{S}}(\mathscr{N}_{0,1}) \to \mathrm{pred}_{\mathsf{T}}(\mathscr{N}_{0,1}) \cup (\mathsf{T}/(\mathsf{T}/\mathbf{S}\ \mathscr{N}_{00})\ \mathbf{S})$$

$$\mathrm{pred}_{\mathsf{T}}(\mathscr{N}) \to \mathbf{S} \cup \mathbf{S}\ \mathrm{pred}(\mathscr{N}) \cup \mathrm{pred}_{\mathsf{T}}(\mathscr{N}_{0,1})\ \mathscr{P}_2 \cup \mathbf{S}\ \mathscr{P}_2\ \mathrm{pred}_{\mathsf{T}}(\mathscr{N})$$

$$\cup\ \mathrm{pred}_{\mathbf{S}}(\mathrm{pred}_{\mathsf{T}}(\mathscr{N}))\ \mathbf{S} \cup \mathrm{pred}_{\mathsf{T}}(\mathrm{pred}_{\mathsf{T}}(\mathscr{N}))\ \mathsf{T}$$

$$\mathrm{pred}_{\mathbf{S}}(\mathrm{pred}_{\mathsf{T}}(\mathscr{N})) \to \mathrm{pred}_{\mathbf{S}}(\mathscr{N}_{0,1}) \cup (\mathsf{T}/\mathbf{S}\ \mathrm{pred}_{\mathsf{T}}(\mathscr{N}_{0,1}))$$

$$\mathrm{pred}_{\mathsf{T}}(\mathrm{pred}_{\mathsf{T}}(\mathscr{N})) \to (\mathsf{T}/\mathbf{S}\ \mathrm{pred}_{\mathsf{T}}(\mathscr{N}_{0,1}))$$

$$\mathrm{pred}_{\mathsf{T}}(\mathrm{pred}_{\mathbf{S}}(\mathscr{N})) \to (\mathsf{T}/\mathsf{S}(\mathsf{S}\ \mathsf{T})\ \mathsf{T} \cup \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_1).$$

For notational convenience, we introduce short names for these non-terminals of the grammar:

Definition 75.

$$\mathit{one} = \mathrm{pred}_{\mathbf{S}}(\mathscr{N}), \qquad \mathit{two} = \mathrm{pred}_{\mathbf{S}}(\mathit{one})$$

$$\mathit{three} = \mathrm{pred}_{\mathbf{S}}(\mathit{two}), \qquad \mathit{four} = \mathrm{pred}_{\mathbf{S}}(\mathit{three}).$$

The following rules complete the grammar:

$$\mathit{one} \to \mathsf{S}\ \mathscr{N}_{00}\ \mathit{one},\ \mathit{two} \to \mathsf{T}\ \mathit{two},\ \mathit{three} \to \mathsf{T}\ \mathit{three},\ \mathit{four} \to \mathsf{T}\ \mathit{four}$$

$$\mathit{one} \to \mathit{two}\ \mathbf{S},\ \mathit{two} \to \mathit{three}\ \mathbf{S},\ \mathit{three} \to \mathit{four}\ \mathbf{S}$$

$$\mathit{four} \to \mathscr{N}_{00},\ \mathit{three} \to \mathscr{N}_{00},\ \mathit{two} \to \mathscr{N}_{00},\ \mathit{one} \to \mathscr{N}_{00}$$

$$\mathit{four} \to \mathsf{S}\ \mathsf{A} \cup \mathsf{S}\ \mathsf{A}\ \mathsf{T}$$

$$\mathit{three} \to \mathsf{S}\ \mathscr{N}_{00} \cup \mathsf{S}(\mathsf{T}/\mathsf{S}\ \mathsf{T}) \cup \mathsf{S}\ \mathsf{A}\ \mathscr{N}_{00} \cup \mathsf{A}\ \mathsf{S}\ \mathsf{A} \cup \mathsf{T}\ \mathsf{T}\ \mathsf{A}$$

$$\mathit{two} \to \mathsf{S}\ \mathscr{N}_{00}\ \mathscr{N}_{00} \cup \mathsf{S}\ \mathsf{T}\ \mathsf{T}\ \mathsf{T} \cup \mathsf{S}\ \mathrm{pred}_{\mathbf{S}}(\mathscr{N}_{0,1})$$

$$\cup\ \mathsf{S}(\mathsf{T}/\mathsf{S}\ \mathrm{pred}_{\mathsf{T}}(\mathscr{N}_{0,1})) \cup \mathsf{A}\ \mathsf{S}\ \mathscr{N}_{00} \cup \mathsf{T}\ \mathsf{T}\ \mathscr{N}_{00}$$

$$\mathit{one} \to \mathsf{S}\ \mathrm{pred}(\mathscr{N}) \cup \mathbf{S}\ \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_0)\ \mathrm{pred}_{\mathbf{S}}(\mathscr{N}_{0,1}) \cup \mathbf{S}\ \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_1)\ \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_0)$$

$$\cup\ (\mathsf{T}/\mathbf{S}\ \mathrm{pred}_{\mathbf{S}}(\mathscr{N}_{0,1}))\ \mathrm{pred}_{\mathbf{S}}(\mathscr{N}_0)$$

$$\cup\ \mathrm{pred}_{\mathsf{T}}(\mathrm{pred}_{\mathbf{S}}(\mathscr{N}))\ \mathsf{T} \cup \mathscr{N}_{00}\ \mathscr{N}_{00} \cup (\mathsf{T}/\mathbf{S}\ \mathsf{T}\ \mathsf{T})\ \mathscr{N}_{00}$$

$$\cup\ \mathsf{A}\ \mathbf{S}\ \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_0) \cup \mathbf{T}\ \mathbf{T}\ \mathrm{pred}_{\mathbf{S}}(\mathscr{L}_0) \cup \mathsf{A}(\mathbf{S}\ \mathsf{T}).$$

ACKNOWLEDGMENT

I thank Henk Barendregt, Thomas Genet, Alfons Geser, Dieter Hofbauer, Florent Jacquemard, Jan Willem Klop, Vincent van Oostrom, and Rick Statman, who provided many useful comments as well as general remarks and historical anecdotes on combinators. I also thank the anonymous referees for valuable suggestions for improving the presentation of this paper.

> *Several years ago when I began to look at the problem... I found that 25 of the letters were comparatively easy to deal with. The other letter was "S". For three days and nights I had a terrible time trying to understand how a proper "S" could really be defined. The solution I finally came up with turned out to involve some interesting mathematics.*
>
> —DONALD E. KNUTH.

Received September 1998

REFERENCES

1. Abramsky, S., Gabbay, D., and Maibaum, T. (1992), "Handbook of Logic in Computer Science," Vol. 2, Clarendon, Oxford.
2. Baader, F., and Nipkow, T. (1998), "Term Rewriting and All That," Cambridge Univ. Press, Cambridge, UK.
3. Barendregt, H. P. (1984), "The Lambda Calculus, Its Syntax and Semantics," Elsevier, Amsterdam/New York.
4. Bergstra, J., and Klop, J. W. (1979), Church–Rosser strategies in the lambda calculus, *Theoret. Comput. Sc.* **9**, 27–38.
5. Comon, H., Dauchet, M., Gilleron, R., Lugiez, D., Tison, S., and Tommasi, M. (1997), "Tree Automata Techniques and Applications," *available at* http://www.grappa.univ-lille3.fr/tata.
6. Ganzinger, H. Ed., (1996), "Rewriting Techniques and Applications," Lecture Notes in Computer Science, No. 1103, Springer-Verlag, Berlin/New York.
7. Gécseg, F., and Steinby, M. (1984), "Tree Automata," Akadémiai Kiadó, Budapest.
8. Genet, T. (1998), Decidable approximations of sets of descendants and sets of normal forms, *in* "Rewriting Techniques and Applications" (T. Nipkow, Ed.), pp. 151–165, Springer-Verlag, Berlin/New York.
9. Jacquemard, F. (1996), Decidable approximations of term rewriting systems, *in* "Rewriting Techniques and Applications" (H. Ganzinger, Ed.), pp. 362–376, Springer-Verlag, Berlin/New York.
10. Kennaway, R., and Sleep, R. (1988), Director strings as combinators, *ACM Trans. Program. Languages Systems* **10**, (4) 602–626.
11. Klop, J. W. (1980), Reduction cycles in combinatory logic, *in* "To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism" (J. P. Seldin and J. R. Hindley, Eds.), pp. 193–214, Academic Press, San Diego.
12. Klop, J. W. (1992), Term rewriting systems, *in* "Handbook of Logic in Computer Science" (S. Abramsky, D. Gabbay, and T. Maibaum, Eds.), Vol. 2, pp. 2–117, Clarendon, Oxford.
13. Knuth, D. E. (1980), The letter S, *Math. Intelligencer* **2**, 114–122.
14. Nipkow, T. Ed., (1998), "Rewriting Techniques and Applications," Lecture Notes in Computer Science, Vol. 1379, Springer-Verlag, Berlin/New York.
15. Seldin, J. P., and Hindley, J. R. Eds., (1980), "To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism," Academic Press, San Diego.
16. Smullyan, R. (1985), "To Mock a Mockingbird: And Other Logic Puzzels Including an Amazing Adventure in Combinatory Logic," Knopf, New York.
17. Sprenger, M., and Wymann-Böni, M. (1993), How to decide the lark, *Theoret. Comput. Sc.* **110**, 419–432.
18. Statman, R. (1989), The word problem for Smullyan's lark combinator is decidable, *J. Symbolic Comput.* **7**, 103–112.
19. Waldmann, J. (1997), Nimm Zwei, Research Group Theoretical Computer Science, Internal Report IR-432, Vrije Universiteit Amsterdam.

20. Waldmann, J. (1998), Normalization of S terms is decidable, *in* “Rewriting Techniques and Applications” (T. Nipkow, Ed.), pp. 138–150, Springer-Verlag, Berlin/New York.

21. Waldmann, J. (1998), The RX home page, *available at* http://www.informatik.uni-leipzig.de/~joe/rx/.

22. Zachos, E. (1978), “Kombinatorische Logik und S-Terme,” Berichte des Instituts für Informatik, No. 26, Eidgenössische Technische Hochschule Zürich, Zürich.

23. Zantema, H. (1993), “Termination of Term Rewriting by Semantic Labelling,” Technical Report UU-CS-93-24, Utrecht University.

24. Zantema, H., and Geser, A. (January 1996), “Non-Looping Rewriting,” Technical Report UU-CS-1996-03, Utrecht University.