On Restarting Transducers*

Norbert Hundeshagen

Fachbereich Elektrotechnik/Informatik
Universität Kassel,
34109 Kassel, Germany
hundeshagen@theory.informatik.uni-kassel.de

Abstract. We study the computional power of restarting transducers, a recently introduced model for computing binary (word) relations. We show that the classes of relations defined by these machines are incomparable to the most common ones. Further, a restricted version, the so-called monotone restarting transducer, yield a class of relations that is almost equivalent to the pushdown relations.

1 Introduction

Restarting automata [7] were invented to model the so-called "analysis by reduction". Simply, this linguistic technique is a method to verify the (syntactical) correctness of a given sentence by a stepwise simplification, under the condition that every step preserves the correctness or incorrectness of the sentence processed. From a linguistic point of view, the verification of correctness (or incorrectness) is not the only goal of performing analysis by reduction, as it is also a useful tool to gain deeper information on the structure of sentences of natural languages, such as word dependency information [9] and morphological ambiguities [10]. Therefore, we are interested in transductions and in ways to compute them by restarting automata.

Several new computational models for realizing transductions, based on restarting automata, have been introduced in recent years. For instance, there are special types of restarting automata that are enhanced to produce tree structures that mirror dependency trees of sentences of natural languages (see e.g., [11]), or parallel communicating systems of restarting automata that realize binary relations [6].

Here we continue a more classical approach by investigating the computational power of restarting automata, extended by the capability to produce strings. These machines are called restarting transducers and were first introduced in [5]. There it was shown that the relations defined by types of restricted restarting transducers form a hierarchy inside the well-known class of rational relations.

* This extended abstract reports on results contained in [4].

2 Definitions and General Observations

A restarting transducer (RRWW-Td for short) consists of a finite-state control, a flexible tape with end markers, a read/write window of a fixed size working on that tape and a write-only oneway output tape. Formally, a restarting transducer is defined as a 9-tuple $T = (Q, \Sigma, \Delta, \Gamma, ¢, \$, q_0, k, \delta)$ where Q is the finite set of states, Σ and Γ are the finite input and tape alphabet, Δ is the finite output alphabet, $¢, \$ \notin \Gamma$ are the markers for the left and right border of the tape, $q_0 \in Q$ is the initial state, $k \geq 1$ is the size of the read/write window, and δ is the transition function.

Such a device works in cycles. In each cycle it moves the head right across the tape and at some point it performs a single rewrite operation that shortens the tape contents. Every cycle ends with a restart operation that forces the transducer to reset the internal state to the initial one and output a word over Δ. After a finite number of cycles, it halts and accepts (or rejects) while also producing some symbols. Thus, a binary relation $\mathrm{Rel}(T) \subseteq \Sigma^* \times \Delta^*$ is defined that consists of all pairs of words (u, v) for which there is an accepting computation started on u and finished with v on the output tape.

Obviously, the relations computed by restarting transducers are length-bounded: There is a constant c, such that for each pair (u, v) $(u \neq \varepsilon)$ in the relation, $|v| \leq c \cdot |u|$. This is for the reason that in every cycle the tape content has to be shortened while only a finite number of output symbols can be produced. Furthermore, we can show that for each recursively enumerable language L, there is a deterministic restarting transducer T such that L is the output language of T.

The latter immediately yields incomparability results to the most common classes of relations, the rational and pushdown relations (see e.g., [2, 3]).

3 Monotone Restarting Transducers

We turn to a more restricted type of restarting transducer by introducing the notion of monotonicity, known from the underlying automaton (see e.g., [8]). Informally, a restarting transducer is called monotone if in each of its computations that starts from an initial configuration, the distance between the right border and the rewrite position is never increased.

We show that the relations computed by transducers of this type are included in the pushdown relations. To establish this result we make use of the concept of input/output-relations associated to restarting automata (see [6]). Furthermore, by using a grammar-based characterization of the pushdown relations (see e.g., [1]) we derive an equivalence to a special subclass, the length-bounded pushdown relations.

References

1. A. V. Aho and J. D. Ullman. Properties of Syntax Directed Translations. *J. Comput. Syst. Sci.*, 3(3):319–334, 1969.

2. J. Berstel. *Transductions and Context-Free Languages.* Leitfäden der angewandten Mathematik und Mechanik. Teubner, 1979.
3. C. Choffrut and K. Culik II. Properties of Finite and Pushdown Transducers. *SIAM J. Comput.*, 12(2):300–315, 1983.
4. N. Hundeshagen. *Relations and Transductions Realized by Restarting Automata.* PhD thesis, Fachbereich Elektrotechnik/Informatik, Universität Kassel, 2013. To appear.
5. N. Hundeshagen and F. Otto. Characterizing the Rational Functions by Restarting Transducers. In A. H. Dediu and C. Martín-Vide, editors, *LATA*, volume 7183 of *Lecture Notes in Computer Science*, pages 325–336. Springer, 2012.
6. N. Hundeshagen, F. Otto, and M. Vollweiler. Transductions Computed by PC-Systems of Monotone Deterministic Restarting Automata. In M. Domaratzki and K. Salomaa, editors, *CIAA*, volume 6482 of *Lecture Notes in Computer Science*, pages 163–172. Springer, 2010.
7. P. Jancar, F. Mráz, M. Plátek, and J. Vogel. Restarting Automata. In H. Reichel, editor, *FCT*, volume 965 of *Lecture Notes in Computer Science*, pages 283–292. Springer, 1995.
8. T. Jurdzinski, F. Mráz, F. Otto, and M. Plátek. Degrees of Non-Monotonicity for Restarting Automata. *Theor. Comput. Sci.*, 369(1-3):1–34, 2006.
9. M. Lopatková, M. Plátek, and V. Kubon. Modeling Syntax of Free Word-Order Languages: Dependency Analysis by Reduction. In V. Matousek, P. Mautner, and T. Pavelka, editors, *TSD*, Lecture Notes in Computer Science, pages 140–147. Springer, 2005.
10. M. Plátek, M. Lopatková, and K. Oliva. Restarting Automata: Motivations and Applications. In M. Holzer, editor, *13. Theorietag 'Automaten und Formale Sprachen', Proc.*, pages 90 – 96, Technische Universität München, 2003.
11. M. Plátek, F. Mráz, and M. Lopatková. Restarting Automata with Structured Output and Functional Generative Description. In A. H. Dediu, H. Fernau, and C. Martín-Vide, editors, *LATA*, volume 6031 of *Lecture Notes in Computer Science*, pages 500–511. Springer, 2010.