UNIVERSITY OF OXFORD

DEPARTMENT OF
COMPUTER
SCIENCE

From: Prof. Andrzej Murawski
andrzej.murawski@cs.ox.ac.uk

6 November 2023

Report on the PhD thesis of Rafał Stefański

The single-use restriction for register automata and transducers over infinite alphabets

Context

Computational models over infinite alphabets have been studied intensively in the last two decades, not least because of compelling applications in program verification, markup languages and database systems, but also owing to interesting technical and foundational problems that infinite alphabets can pose. A natural question to ask in this context is whether various classic correspondences concerning automata over finite alphabets can be carried over to infinite alphabets. Unfortunately, most computational models proposed in the literature on infinite alphabets turned out to be incomparable, which is not a satisfying outcome.

The present thesis discovers that, by forcing computations to use resources (atoms) at most once (*single-use restriction*), it is possible to recover robustness and prove a wide range of equiexpressivity results, where the multi-use variants of the same models were previously disparate. The author studies several prominent models under this restriction, taking into account both one- and two-way models, and both automata and transducers.

Content

The thesis consists of an Introduction, four technical chapters, a Further Work section and the bibliography.

Introduction

The Introduction sets out the context of the work and places the thesis within that context as a "third wave" of research into infinite alphabets, where the first wave is seen to have contributed numerous incomparable models, the second one had a focus on systematic and principled generalizations through sets with atoms, whereas the third one aims to identify cases where classic relationships known from finite-alphabets can be recovered, e.g. by fine-tuning existing models. In the thesis, it turns out that imposing single-use constraints leads to a satisfying picture from this point of view.

Chapter 1

Chapter 1 presents the foundations of set theory over atoms and those of the corresponding automata theory. Among others, it recalls the definition of *orbit-finite* sets, which underpin most of the concepts considered in the thesis and which often play the role of finite sets. The chapter also surveys closure properties of sets with atoms (or lack of closure, if applicable) as well as introducing an auxiliary concept of straight sets, which facilitates technical arguments.

The chapter ends with a discussion of several mutually inequivalent models over infinite alphabets, notably two-way orbit-finite automata, one-way orbit-finite automata and orbit-finite monoids. They will turn out equivalent once the single-use restriction is imposed in the next chapter.

UNIVERSITY OF OXFORD

DEPARTMENT OF
COMPUTER
SCIENCE

From: Prof. Andrzej Murawski
andrzej.murawski@cs.ox.ac.uk

Chapter 2

The second chapter develops automata models with the single-use restriction. It begins with an adaptation of register automata, called *single-use register automata*, where the transition function is a branching program with destructive semantics (evaluating a condition destroys the content) and acceptance is specified by configuration (acceptance by state would be strictly weaker).

The author then introduces single-use (SU) functions, which will be used to recast the single-use register automata in much a more abstract and general form. The SU functions are defined over *polynomial orbit-finite sets* only, which are generated from the set of atoms $\mathbb{A}$ and a singleton atom-free set 1 via products and sums. The definition of SU functions is inductive and relies on combining a set of base functions (e.g. equality test for pairs of atoms) using composition, product and disjoint union. Crucially, the base functions do not include the copying map $\mathtt{copy} : \mathbb{A} \rightarrow \mathbb{A} \times \mathbb{A}$. Indeed, it is shown that its addition would make the set equal to that of all finitely supported functions (between polynomial orbit-finite sets). The author also proposes a concrete representation for SU functions in the form of decision trees in which on every branch each variable can appear at most once. Such trees are shown to define exactly the SU functions and their concrete nature helps to prove that the set of SU functions is orbit-finite (Theorem 5). This is in stark contrast to the set of arbitrary finitely supported functions.

Having established the concept of SU functions, the author can now proceed to define *single-use (one-way) automata*, where both the alphabet and the set of states can be arbitrary polynomial orbit-finite sets. The definition of the transition function requires some care here, because one wants to preserve unrestricted access to the input in an otherwise single-use framework. Consequently, the associated type must involve both single- (for states) and multiple-use (for input). In a similar style, the author then defines *single-use two-way automata*.

The Chapter culminates in Theorem 6, which states that both one-way and two-way single-use automata are equivalent to orbit-finite monoids (limited to polynomial orbit-finite sets), i.e. the three classes of models have the same expressivity. Two thirds of the proof (e.g. the embedding of two-way automata into orbit-finite monoids) are included right away, while the translation from orbit-finite monoids to the one-way model is delayed until the next chapter, when it becomes a simple corollary of a more general result for transducers.

At the very end of the chapter, the author discusses the difficulty of reconciling his methodology with nondeterminism. Depending on the approach taken, the problems stem from the lack of compositionality, lack of orbit-finiteness or simply stronger expressivity with respect to the deterministic setting.

Chapter 3

While Chapter 2 was dedicated to accepting languages, the next two chapters concern transducers, i.e. defining string-to-string functions. In particular, Chapter 3 focuses on deterministic letter-to-letter transducers, known as Mealy machines. In the finite-alphabet setting, such machines admit a Krohn-Rhodes decomposition, i.e. the induced function can be decomposed into certain prime functions. This chapter shows that a similar result can be recovered after Mealy machines are adapted to infinite alphabets using single-use functions (Theorem 8).

The translation from single-use prime functions to single-use Mealy machines is relatively routine. However, the proof of the converse is a real tour de force. There the author resorts to an algebraic approach, namely, semigroup transductions from finite alphabets. Their immediate orbit-finite extension is too strong to correspond to single-use Mealy machines. However, he succeeds in identifying a matching restriction, called *locality* (Theorem 9). Theorem 9 provides the missing link in the proof of Theorem 6 as well as an intermediate step on the way from single-use Mealy machines to single-use primes.

DEPARTMENT OF
COMPUTER
SCIENCE

UNIVERSITY OF OXFORD

From: Prof. Andrzej Murawski
andrzej.murawski@cs.ox.ac.uk

The remaining part of the argument, which shows how to translate local semigroup transductions into composites of single-use primes, is particularly technical and relies on factorization trees. Unfortunately, existing results on (idempotent) factorization trees do not suffice to reestablish the results for orbit-finite semigroups, so the author proposes a relaxation of idempotence, called smoothness, which does fit the bill (Theorem 11). As an aside, it is also shown how one can decide whether orbit-finite semigroup transductions are equivalent to local ones (Theorem 12).

The final section of Chapter 3 discusses unambiguous Mealy machines, which are nondeterministic variants of Mealy machines that generate the so-called rational functions in the finite-alphabet case. The nondeterminism prevents the author from adapting the results to atoms by introducing a single-use variant. Still, he succeeds in lifting the algebraic part of the theory (rational semigroup transductions) to the orbit-finite setting through local rational semigroup transductions and a Krohn-Rhodes decomposition result (Theorem 14).

Chapter 4

Chapter 4 studies two-way transductions with atoms. Here the author introduces four models that will turn out to be equiexpressive (Theorem 15). The first one, *single-use two-way transducers*, is obtained by applying the single-use methodology advocated in the thesis to the definition of two-way automata with output. The second (one-way) model, called *single-use streaming string transducers*, is inspired by Alur and Černý's copyless streaming string transducers, which are equipped with string-valued registers whose contents cannot be copied. The copyless nature of registers makes them amenable to a presentation through single-use functions already for finite alphabets and then the passage to polynomial orbit-finite sets yields the intended model over atoms. The third model extends that of regular list functions (due to Bojańczyk, Daviaud and Krishna) by adding $\mathbb{A}$ as a new type, interestingly, with the possibility of copying atoms. The fourth model is motivated by Krohn-Rhodes-like decompositions and based on a carefully chosen set of *single-use two-way prime functions*.

The proof of Theorem 15 consists of a chain of translations, with the most difficult case being the passage from two-way transducers to prime decompositions.

Assessment

The thesis features a series of highly non-trivial and original results that succeed in recovering, in the infinite-alphabet setting, several well-known correspondences from the finite-alphabet framework, subject to imposing the single-use restriction. On the automata side, they show the equiexpressivity of one-way and two-way models, together with their equivalence to recognition by monoids. On the transducer side, they contribute a Krohn-Rhodes-like decomposition for Mealy machines along with an equivalent algebraic definition, as well as extending the correspondence between two-way transducers, copyless streaming string transducers, regular list functions and Krohn-Rhodes-like decompositions.

These are all substantial technical achievements and I have no hesitation in describing the technical quality of the thesis as outstanding. This is further confirmed by the high-quality narrative and the wide variety of techniques employed in the thesis, which showcase excellent technical skills of the author.

In addition to theoretical value, the equiexpressivity results presented in the thesis allow one to carry over results from one setting to another, where it might not be equally easy to establish them. For the formalisms studied in the thesis, this is a case in point when it comes to compositionality and decidability of the associated equivalence problems.

DEPARTMENT OF
COMPUTER
SCIENCE
UNIVERSITY OF
OXFORD

From: Prof. Andrzej Murawski
andrzej.murawski@cs.ox.ac.uk

Although the thesis develops the theory of single use sufficiently to study various kinds of automata, one wonders whether it could not be generalized further, e.g. beyond polynomial orbit-finite sets. Techniques borrowed form linear logic and the associated category theory might turn out helpful in this endeavour. For example, the present definition of single-use functions resembles free category constructions and the result on currying suggests that it might be possible to turn single-use functions into a category with function spaces.

Most of the presented results have already been published by the author (in a joint paper with his supervisor) at ICALP 2020, which is regarded as a top venue in this area. But the thesis is much more than a "journal version" of the paper with full proofs. Thanks to a refined definition of single use, excellent examples, frequent introductory paragraphs and well-informed references to the history of the subject, the document feels more like a monograph on single-use functions and their applications to automata models. What is especially valuable in the context of a doctoral thesis is the fact that the author does not shy away from presenting his own perspective on the results and motivating various technical decisions. All of the above-mentioned aspects are testament to his scientific maturity.

Despite its length (218 pages), the thesis reads well and captures the attention of the reader thanks to a pedagogical exposition involving well-chosen examples and helpful comments. It is written in very good English and the number of typographical slip-ups is relatively small given the size of the document. I have listed some of them in the Appendix.

In conclusion, there can be no doubt that the research reported in the thesis and the quality of its presentation merit the award of a PhD. What is more, the technical depth and breadth as well as exemplary clarity of exposition make the thesis a clear candidate for a distinction.

Andrzej Murawski

Andrzej Murawski

Department of Computer Science, Wolfson Building, Parks Road, Oxford OX1 3QD
General Enquiries Tel: +44 (0)1865 273838 Web: www.cs.ox.ac.uk

From: Prof. Andrzej Murawski
andrzej.murawski@cs.ox.ac.uk

Appendix

9	oribit (orbit)
10	Defintion (Definition)
16	wihtout (without)
17	function (functions)
19	`type Q` (`data Q` would be more Haskell-like)
22	element (elements), Pits (Pitts)
23	if it for (if for)
25	its orbit (its orbits)
27	discuses (discusses)
29	representant (representative)
43	Claim 12: $supp(x)$ ($supp(a)$)
43	universally (universality)
45	finte (finite)
46	straight forward (straightforward)
49	Lemma 20: orbit-finite (finite)
60	$X_1 \xrightarrow{g} Y_1$ ($X_2 \xrightarrow{g} Y_2$)
62	can from (can be built from)
69	proof (prove), quieries (queries), nodes contain (leaves contain)
71	som (some)
72	tress (trees)
73	$X_1 \xrightarrow{g} X_2$ ($X_2 \xrightarrow{g} Y_2$)
87	nondeterminsim (nondeterminism)
93	mealy (Mealy), is can (can)
94	δ equations incompatible with δ's type
97	function (functions), is the equal (is equal), $\{=, \neq\}$ ($\{=, \neq\}^*$)
110	Theorem 11: idempotent (smooth)
152	transduction (transductions)
179	coplyless (copyless)