ARTICLE IN PRESS
YINCO:4298

Information and Computation ••• (••••) •••–•••

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

Deletion operations on deterministic families of automata [☆]

Joey Eremondi [a,2], Oscar H. Ibarra [b,1], Ian McQuillan [c,*,2]

[a] *Department of Information and Computing Sciences, Utrecht University, P.O. Box 80.089, 3508 TB Utrecht, The Netherlands*
[b] *Department of Computer Science, University of California, Santa Barbara, CA 93106, USA*
[c] *Department of Computer Science, University of Saskatchewan, Saskatoon, SK S7N 5A9, Canada*

ARTICLE INFO

Article history:
Received 26 August 2015
Available online xxxx

Keywords:
Automata and logic
Counter machines
Deletion operations
Reversal-bounds
Determinism
Finite automata

ABSTRACT

It is shown that one-way deterministic reversal-bounded multicounter languages are closed under right quotient with languages from many language families; even those defined by nondeterministic machines such as the context-free languages. Also, it is shown that when starting with one-way deterministic machines with one counter that makes only one reversal, taking the left quotient with languages from many different language families — again including those defined by nondeterministic machines such as the context-free languages — yields only one-way deterministic reversal-bounded multicounter languages. These results are surprising given the nondeterministic nature of the deletion. However, if there are two more reversals on the counter, or a second 1-reversal-bounded counter, taking the left quotient (or even just the suffix operation) yields languages that can neither be accepted by deterministic reversal-bounded multi-counter machines, nor by 2-way deterministic machines with one reversal-bounded counter.

1. Introduction

This paper involves the study of various types of deletion operations applied to languages accepted by deterministic classes of machines. Deletion operations, such as left and right quotients, and word operations such as prefix, suffix, infix, and outfix, are more commonly studied applied to languages accepted by classes of nondeterministic machines. Indeed, many language families accepted by nondeterministic acceptors form *full trios* (closure under homomorphism, inverse homomorphism, and intersection with regular languages), and every full trio is closed under left and right quotient with regular languages, prefix, suffix, infix, and outfix [2]. For families of languages accepted by deterministic machines however, the situation is more tricky due to the nondeterministic behaviour of the deletion. Indeed, deterministic pushdown automata are not even closed under left quotient with a set of individual letters. Here, most deterministic machine models studied will involve restrictions of one-way deterministic reversal-bounded multicounter machines (DCM). These are machines that operate like deterministic finite automata with an additional fixed number of counters, where there is a bound on the number of times each counter switches between increasing and decreasing [3,4]. The family $\mathsf{DCM}(k,l)$ consists of languages accepted by machines with k counters that are l-reversal-bounded. DCM languages have many decidable properties, such

☆ This is an extended version of the conference paper [1].

* Corresponding author.
E-mail addresses: j.s.eremondi@students.uu.nl (J. Eremondi), ibarra@cs.ucsb.edu (O.H. Ibarra), mcquillan@cs.usask.ca (I. McQuillan).

1 Supported, in part, by NSF Grant CCF-1117708 (Oscar H. Ibarra).
2 Supported, in part, by Natural Sciences and Engineering Research Council of Canada Grant 327486-2010 (Ian McQuillan).

http://dx.doi.org/10.1016/j.ic.2017.07.009
0890-5401/© 2017 Elsevier Inc. All rights reserved.

Please cite this article in press as: J. Eremondi et al., Deletion operations on deterministic families of automata, Inf. Comput. (2017), http://dx.doi.org/10.1016/j.ic.2017.07.009

as emptiness, infiniteness, equivalence, inclusion, universe, and disjointness [4]. Furthermore, DCM$(1, l)$ forms an important restriction of deterministic pushdown automata.

These machines have been studied in a variety of different applications, such as to membrane computing [5], verification of infinite-state systems [6–9], and Diophantine equations [9].

Recently, in [10], a related study was conducted for insertion operations; specifically operations defined by ideals obtained from the prefix, suffix, infix, and outfix relations, as well as left and right concatenation with languages from different language families. It was found that languages accepted by one-way deterministic reversal-bounded counter machines with one reversal-bounded counter are closed under right concatenation with Σ^*, but having two 1-reversal-bounded counters and right concatenating Σ^* yields languages outside of both DCM and 2DCM(1) (languages accepted by two-way deterministic machines with one counter that is reversal-bounded). It also follows from this analysis that the right input end-marker is necessary for even one-way deterministic reversal-bounded counter machines, when there are at least two counters. Furthermore, concatenating Σ^* to the left of some one-way deterministic 1-reversal-bounded one counter languages yields languages that are neither in DCM nor 2DCM(1). Other recent results on reversal-bounded multicounter languages include a technique to show languages are outside of DCM [11]. Closure properties of nondeterministic counter machines under other types of deletion operations were studied in [12].

In this paper we investigate closure properties of types of deterministic machines. In Section 2, preliminary background and notation are introduced. In Section 3, erasing operations where DCM is closed are studied. It is shown that DCM is closed under right quotient with context-free languages, and that the left quotient of DCM$(1, 1)$ by a context-free language is in DCM. Both results are generalizable to quotients with a variety of different families of languages containing only semilinear languages. In Section 4, non-closure of DCM under erasing operations are studied. It is shown that the set of suffixes, infixes, or outfixes of a DCM$(1, 3)$ or DCM$(2, 1)$ language can be outside of both DCM and 2DCM(1). In Section 5, DPCMs (deterministic pushdown automata augmented by reversal-bounded counters), and NPCMs (the nondeterministic variant) are studied. It is shown that DPCM is not closed under prefix or suffix, and the right or left quotient of the language accepted by a 1-reversal-bounded deterministic pushdown automaton by a DCM$(1, 1)$ language can be outside DPCM. In Section 6, the effective closure of regular languages with other families is briefly discussed, and in Section 7, bounded languages are discussed.

2. Preliminaries

The set of non-negative integers is denoted by $\mathbb{N}_0$, and the set of positive integers by $\mathbb{N}$. For $c \in \mathbb{N}_0$, let $\pi(c)$ be 0 if $c = 0$, and 1 otherwise.

We assume knowledge of standard formal language theoretic concepts such as finite automata, determinism, nondeterminism, semilinearity, recursive, and recursively enumerable languages [3,13]. Next, we will give some notation used in the paper. The empty word is denoted by λ. If Σ is a finite alphabet, then Σ^* is the set of all words over Σ and $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$. For a word $w \in \Sigma^*$, if $w = a_1 \cdots a_n$ where $a_i \in \Sigma$, $1 \le i \le n$, the length of w is denoted by $|w| = n$, and the reversal of w is denoted by $w^R = a_n \cdots a_1$, which is extended to reversals of languages in the natural way. In addition, if $a \in \Sigma$, $|w|_a$ is the number of a's in w. A language over Σ is any subset of Σ^*. Given a language $L \subseteq \Sigma^*$, the complement of L, $\Sigma^* \setminus L$ is denoted by $\overline{L}$. Given two languages L_1, L_2, the left quotient of L_2 by L_1, $L_1^{-1}L_2 = \{y \mid xy \in L_2, x \in L_1\}$, and the right quotient of L_1 by L_2 is $L_1L_2^{-1} = \{x \mid xy \in L_1, y \in L_2\}$. A *full trio* is a language family closed under homomorphism, inverse homomorphism, and intersection with regular languages [13].

Let $n \in \mathbb{N}$. Then $Q \subseteq \mathbb{N}_0^n$ is a *linear* set if there is a vector $\vec{c} \in \mathbb{N}_0^n$ (the constant vector), and a set of vectors $V = \{\vec{v_1}, \dots, \vec{v_r}\}, r \ge 0$, each $\vec{v_i} \in \mathbb{N}_0^n$ such that $Q = \{c + t_1\vec{v_1} + \cdots + t_r\vec{v_r} \mid t_1, \dots, t_r \in \mathbb{N}_0\}$. A finite union of linear sets is called a *semilinear set*.

A language L is *word-bounded* or simply *bounded* if $L \subseteq w_1^* \cdots w_k^*$ for some $k \ge 1$ and (not-necessarily distinct) words $w_1, \dots, w_k$. Further, L is *letter-bounded* if each w_i is a letter. Also, L is *bounded-semilinear* if $L \subseteq w_1^* \cdots w_k^*$ and $Q = \{(i_1, \dots, i_k) \mid w_1^{i_1} \cdots w_k^{i_k} \in L\}$ is a semilinear set [14].

We now present notation for common word and language operations used throughout the paper.

Definition 1. For a language $L \subseteq \Sigma^*$, the prefix, suffix, infix, and outfix operations are defined by:

- $\text{pref}(L) = \{w \mid wx \in L, x \in \Sigma^*\}$,
- $\text{suff}(L) = \{w \mid xw \in L, x \in \Sigma^*\}$,
- $\text{inf}(L) = \{w \mid xwy \in L, x, y \in \Sigma^*\}$,
- $\text{outf}(L) = \{xy \mid xwy \in L, w \in \Sigma^*\}$.

Note that $\text{pref}(L) = L(\Sigma^*)^{-1}$ and $\text{suff}(L) = (\Sigma^*)^{-1}L$.

The outfix operation has been generalized to the notion of embedding [15]:

Definition 2. The m-embedding of a language $L \subseteq \Sigma^*$ is the following set: $\text{emb}(L, m) = \{w_0 \cdots w_m \mid w_0x_1 \cdots w_{m-1}x_mw_m \in L$, $w_i \in \Sigma^*, 0 \le i \le m, x_j \in \Sigma^*, 1 \le j \le m\}$.

Please cite this article in press as: J. Eremondi et al., Deletion operations on deterministic families of automata, Inf. Comput. (2017), http://dx.doi.org/10.1016/j.ic.2017.07.009