Information Processing Letters 116 (2016) 729–734

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

The (nested) word problem

Christopher S. Henry

ARTICLE INFO

Article history:
Received 19 October 2015
Received in revised form 3 June 2016
Accepted 28 June 2016
Available online 12 July 2016
Communicated by Krishnendu Chatterjee

Keywords:
Word problem
Formal languages
Visibly pushdown languages

ABSTRACT

In this paper we provide a new perspective on the word problem of a group by using languages of nested words. These were introduced by Alur and Madhusudan as a way to model data with both a linear ordering and a hierarchically nested matching of items, like HTML or XML documents. We demonstrate how a class of nested word languages called visibly pushdown can be used to study the word problem of virtually free groups in a natural way.

1. Introduction

There are deep connections between formal language theory and group theory, and this relationship has been used to successfully explore the structure of groups by mathematicians. An important example of this is the study of the word problem for groups: given a group G with finite generating set X, the word problem is the set words over the alphabet $A = X \cup X^{-1}$ which represent the identity element in G. For a class $\mathscr{L}$ of formal languages, an interesting question is whether a given finitely generated group has word problem in $\mathscr{L}$.

A classical result of Anisimov [2] says that finite groups are characterized by having a word problem that is a regular language. In [13] Muller and Schupp proved a remarkable analogue of this theorem, characterizing groups whose word problems are context-free languages. Their work has been extended by considering still other classes of word languages, see for example [7,8,5,9]. Similar kinds of results have also been obtained by analyzing the complement of the word problem, see [10] and [11], and in general this fruitful line of research remains active even today, see for example [3] and [4].

Here we wish to take a slightly different approach by investigating possible connections between group theory and other kinds of formal languages besides word languages, for example operator precedence languages (also known as Floyd languages), or tree languages. A good place to start is with nested word languages, introduced by Alur and Madhusudan in [1]. Nested words model data with both a linear ordering and hierarchically nested matching of items, and provide a generalization of both words and ordered trees. In addition, regular languages of nested words have a close relationship to both regular and context free word languages.

Specifically, in this article we use nested words to study the word problem for finitely generated groups. The motivation for doing so comes from the example of the free group; any word representing the identity can be naturally viewed as a nested word (see Example 3.2). We extend the definition of the word problem to languages of nested words, and show that regular languages of nested words characterize the same class of groups as context-free languages. Finally, we use closure properties of nested word languages to give precise characterizations of two classes of virtually free groups.

2. Preliminaries

2.1. Context-free languages

Let A be a finite set, which we will call an alphabet. For each $n \in \mathbb{N}$, let $A^n = \{w \mid w : \{1, 2, \ldots, n\} \to A$

E-mail address: chenry6@uwo.ca.

http://dx.doi.org/10.1016/j.ipl.2016.06.013
0020-0190/

is a function}. An element $w \in A^n$ is called a word of length $|w| = n$, and is written as $w = a_1 \cdots a_n$ where $w(i) = a_i \in A$. Denote by ϵ the unique element $\epsilon : \varnothing \to A$ of A^0 called the empty word. Finally let $A^* = \bigcup_{n=0}^{\infty} A^n$ be the set of all finite words over the alphabet A. A *language* over A is any subset $L \subset A^*$.

We refer the reader to the classic text [12] for a detailed background on context-free and regular languages. Recall that a *pushdown automaton* (PDA) is a tuple $M = (A, S, s_0, Y, \Gamma, \gamma_0, \delta)$ where: A is an alphabet; S is a finite set of states, with $s_0 \in S$ the initial state and $Y \subset S$ the set of accepting states; Γ is the stack alphabet, with bottom of stack symbol $\gamma_0 \in \Gamma$; and $\delta \subseteq (S \times A \cup \{\epsilon\} \times \Gamma) \times (S \times \Gamma^*)$ is the transition relation, where we sometimes write $\delta(s, a, \gamma) = (s', \chi)$ when $(s, a, \gamma, s', \chi) \in \delta$. If $\delta \subseteq (S \times A \times \Gamma) \times (S \times \Gamma^*)$ and for every $(s, a, \gamma) \in S \times A \times \Gamma$ there is at most one transition $\delta(s, a, \gamma) = (s', \chi)$, then the PDA is called *deterministic*.

Conceptually, we think of M as reading in a word $w \in A^*$ one letter at a time, and making transitions based on the current letter, state, and top of stack symbol. Based on this data the transition relation δ updates the state, erases the top of stack symbol, and writes a new word $\chi \in \Gamma^*$ to the top of the stack. A word w is accepted by M if, after reading w, the machine is in an accepting state $y \in Y$; denote by $L(M)$ the set of all words accepted by M.

Definition 2.1. A language $L \subset A^*$ is context-free (CF) if there exists some PDA M such that $L = L(M)$. We denote the class of all context-free languages by $\mathscr{L}_{CF}$.

We can view regular languages as a special type of CF language. In particular, a finite state automaton (FSA) is a PDA satisfying $\Gamma = \varnothing$, and a language is called *regular* if there is some FSA N such that $L = L(N)$. The class of all regular languages, denoted $\mathscr{L}_{reg}$, satisfies a wide range of closure properties, some of which do not necessarily hold for CF languages in general.

Proposition 2.2 *([12] or [6]). Let L, L_1, and $L_2 \in \mathscr{L}_{reg}$ be regular languages over the alphabet A. The following languages are also regular:*

(1) $L_{\text{comp}} = A^* \setminus L$,
(2) $L_1 \cap L_2$,
(3) $L_1 \cup L_2$,
(4) $L_1 \cdot L_2 = \{w_1 w_2 \mid w_1 \in L_1 \text{ and } w_2 \in L_2\}$,
(5) $L^* = \{w_1 \dots w_n \mid n \in \mathbb{N} \text{ and } w_i \in L\}$, *the Kleene-star closure of L*,
(6) $L^R = \{w^R = a_n \cdots a_1 \mid w = a_1 \cdots a_n \in L\}$,
(7) $L_{\text{pre}} = \{w \mid wu = v \in L\}$, *the prefix closure of L*,
(8) $\phi(L)$ *for any language homomorphism ϕ.*

For $L, L_1, L_2 \in \mathscr{L}_{CF}$, it is not true in general that 1 and 2 belong to $\mathscr{L}_{CF}$.

2.2. Nested word languages

Now we turn to nested words and their languages, which were introduced by Alur and Madhusudan in [1].

Definition 2.3. A nested word is a pair $(w, \frown)$ with $w = a_1 \dots a_n \in A^*$ and $\frown$ is a subset of $\{-\infty, 1, \dots, n\} \times \{1, \dots, n, \infty\}$ satisfying:

(1) (Matching edges go forward) $i \frown j \Rightarrow i < j$,
(2) (Uniqueness) $\forall\ 1 \le i \le n$, $|\{j \mid i \frown j\}| \le 1$ and $|\{j \mid j \frown i\}| \le 1$,
(3) (Nesting Property) if $i_1 \frown j_1$ and $i_2 \frown j_2$ with $i_1 < i_2$, then either $j_2 < j_1$ or $j_1 < i_2$.

Let $NW(A)$ denote the set of all nested words over A. If $i \frown j$ then we say that a_i is a call and a_j is return; if a_k is neither a call nor a return it is called an internal symbol. A nested word $(w, \frown)$ is called *well-matched* if $\frown \subseteq \{1, \dots, n\} \times \{1, \dots, n\}$, and we denote the set of all well-matched nested words over A by $WM(A)$.

We can encode nested words over A by extending the alphabet to the tagged alphabet $\widetilde{A} = {}^{\ulcorner}A \cup A \cup A^{\urcorner}$. The alphabets ${}^{\ulcorner}A$ and $A^{\urcorner}$ are disjoint copies of A where each element $a \in A$ is replaced with ${}^{\ulcorner}a$ and $a^{\urcorner}$, respectively. The idea is that we can tag the letters of any word $w \in A^*$ to be either a call or return, or left as an internal symbol. In doing so, there is only one possible interpretation of the tagging that gives a nested word.

Lemma 2.4 *(2.1 in [1]). There is a natural bijection $\tau : NW(A) \to \widetilde{A}^*$.*

We use $\widetilde{a} \in \widetilde{A}$ to denote an element in the tagged alphabet, so $\widetilde{a} \in \{{}^{\ulcorner}a, a, a^{\urcorner}\}$. A tagged word $\widetilde{w} \in \widetilde{A}^*$ can then be written as $\widetilde{w} = \widetilde{a}_1 \cdots \widetilde{a}_n$, and a language of nested words over A is any subset $L \subset \widetilde{A}^*$. This leads to a convenient way of defining *visibly pushdown languages* (VPLs), which are also called *regular languages of nested words*. Given an alphabet A, a visibly pushdown automaton (VPA) is a deterministic PDA over the extended alphabet $\widetilde{A}$, where the transition relation $\delta = \delta_c \cup \delta_i \cup \delta_r$ consists of call transitions δ_c, return transitions δ_r, and internal transitions δ_i, which satisfy:

- $\delta_c \subseteq \{(s, {}^{\ulcorner}a, \gamma, s', \gamma\gamma') \mid s, s' \in S, {}^{\ulcorner}a \in {}^{\ulcorner}A, \gamma, \gamma' \in \Gamma\}$,
- $\delta_i \subseteq \{(s, a, \gamma, s', \gamma) \mid s, s' \in S, a \in A, \gamma \in \Gamma\}$,
- $\delta_r \subseteq \{(s, a^{\urcorner}, \gamma, s', \epsilon) \mid s, s' \in S, a^{\urcorner} \in A^{\urcorner}, \gamma \in \Gamma\}$.

This says that the input symbol completely determines the stack operation. If $\widetilde{a}$ is a call, then a single letter is added to the stack; if $\widetilde{a}$ is a return the transition depends on the current top of stack symbol, which is subsequently removed. Internal symbols do not induce a stack operation.

Definition 2.5. Given an alphabet A, a language of nested words $L \subset \widetilde{A}^*$ is called visibly pushdown (or a VPL) if there exists a VPA $\widetilde{M}$ such that $L = L(\widetilde{M})$.

The following results emphasize how VPLs are closely related to CF languages.

Theorem 2.6 *(5.1 in [1]). If $L \subset \widetilde{A}^*$ is a VPL, then it is also a CF language over the alphabet $\widetilde{A}$.*