Information and Computation 206 (2008) 1018–1032

Contents lists available at ScienceDirect

Information and Computation

journal homepage: www.elsevier.com/locate/ic

Commutation-augmented pregroup grammars and push-down automata with cancellation

Nissim Francez *, Michael Kaminski

Department of Computer Science, Technion – Israel Institute of Technology, Haifa 32000, Israel

ARTICLE INFO

Article history:
Received 20 July 2007
Revised 25 February 2008
Available online 5 June 2008

Keywords:
Pregroup grammars
Commutation
Push-down automata
Pumping lemma

ABSTRACT

The paper proves a pumping lemma for a certain subclass of mildly context-sensitive languages, the one defined by commutation-augmented pregroup grammars; in addition, an automaton equivalent to such grammars is introduced, augmenting push-down automata by *cancellation* in the bottom of its stack.

© 2008 Elsevier Inc. All rights reserved.

1. Introduction

Pregroup grammars were introduced by Lambek [9], giving rise to a *radically lexicalized* theory of formal/programming/natural languages, by which properties of terminals determine the language recognized by a grammar. The rules are universal (algebraic in this case) and do not vary with the language defined, as in rewriting grammars. In [2] Buszkowski established the equivalence of pregroup grammars and context-free grammars in terms of *weak generative power*, i.e., they define the same class of (formal) languages.

In [4], a certain generalization of pregroup grammars was proposed, whereby the free pregroup is augmented with a finite set of inequations between types, expressing *commutation* and *cancellation* during reductions. This augmentation leads to a class of languages transcending the context-free languages and, in particular, including languages such as

(1) *Reduplication*: $\{ww : w \in \Sigma^+\}$.
(2) *Crossed dependency*: $\{a^i b^j a^i b^j : i,j = 1,2,\ldots\}$.
(3) *Multiple agreement*: $\{a^i b^i c^i : i = 1,2,\ldots\}$.

all known as *mildly context-sensitive languages* (MCSL) [7].

In this paper, we

- extend the study of lexicalized definitions of (some) mildly context-sensitive languages, and state and prove a pumping lemma for such languages; and
- introduce an automaton that recognizes these mildly context-sensitive languages.

A more detailed comparison with other formalisms expressing MCSL [12], e.g., *Tree-Adjoining Grammars* (TAG) [8] and [11], is deferred to future work. For a recent comparative study of MCSL see [10].

* Corresponding author.
E-mail addresses: francez@cs.technion.ac.il (N. Francez), kaminski@cs.technion.ac.il (M. Kaminski).

0890-5401/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.ic.2008.03.004

The paper is organized as follows. In the next section, we review the definition of pregroup grammars[1] and in Section 3, we provide an alternative proof (via push-down automata) of their equivalence to context-free grammars. Section 4 reviews the definition of restricted commutation-augmented pregroup grammars (RCAPGG) from [4] and their basic properties. It also contains the statement of the pumping lemma for RCAPGG languages whose proof is presented in Section 5. In Section 6, we introduce the restricted canceling push-down automata and prove their equivalence to RCAPGGs. Finally, the last section contains some concluding remarks.

2. Pregroup grammars

In this section, we define pregroup grammars and a certain extension thereof.

Definition 1. A *pregroup* is a tuple $\mathcal{G} = \langle G, \leq, \circ, \ell, r, 1\rangle$, such that $\langle G, \leq, \circ, 1\rangle$ is a *partially ordered monoid*,[2] i.e., satisfying (where A, B, and C range over G)

(mon) if $A \leqslant B$, then $CA \leqslant CB$ and $AC \leqslant BC$

and ℓ,r are unary operations (left/right inverses/adjoints) satisfying

(pre) $A^\ell A \leqslant 1 \leqslant AA^\ell$ and $AA^r \leqslant 1 \leqslant A^r A$

The following equalities can be shown to hold in any pregroup.

$$1^\ell = 1^r = 1, A^{\ell r} = A^{r\ell} = A, (AB)^\ell = B^\ell A^\ell, (AB)^r = B^r A^r \tag{1}$$

Also, (**mon**) together with (1) yield

$$A \leqslant B \text{ if and only if } B^\ell \leqslant A^\ell \text{ if and only if } B^r \leqslant A^r \tag{2}$$

Actually, for the definition of a *pregroup grammar* we need only the notion of a *quasi-pregroup*. Quasi-pregroups are defined as pregroups except that the relation $\leqslant$ being reflexive and transitive, does not have to be antisymmetric. That is, $\leqslant$ is a *quasi-ordering*.

The following construction of a *free* quasi-pregroup is due to Lambek, see [9]. Let $\mathcal{B}$ be a (finite) set. *Terms* are of the form $A^{(n)}$, $A \in \mathcal{B}$ and $n = 0, \pm 1, \pm 2, \cdots$. The set of all terms generated by $\mathcal{B}$ is denoted by $\tau(\mathcal{B})$.

The elements of the free quasi-pregroup based on $\langle \mathcal{B}, \leqslant\rangle$ are *types*[3] which are finite strings of terms, '$\circ$' is just the concatenation of types, and 1 is the empty string. The set of all types generated by $\mathcal{B}$ is denoted by $\kappa(\mathcal{B})$. The *length* of a type (finite string of terms) x is denoted by $|x|$. The adjoints are given by

- $\left(A_1^{(n_1)} \cdots A_k^{(n_k)}\right)^\ell = \left(A_k^{(n_k-1)} \cdots A_1^{(n_1-1)}\right)$ and
- $\left(A_1^{(n_1)} \cdots A_k^{(n_k)}\right)^r = \left(A_k^{(n_k+1)} \cdots A_1^{(n_1+1)}\right)$.

Extend '$\leq$' to $\kappa(\mathcal{B})$ by setting it to the smallest quasi-partial-order satisfying (where $\gamma, \delta \in \kappa(\mathcal{B})$)

(con) $\gamma A^{(n)} A^{(n+1)} \delta \leqslant \gamma\delta$ (*contraction*)

(exp) $\gamma\delta \leqslant \gamma A^{(n+1)} A^{(n)} \delta$ (*expansion*)

and

(ind) $\gamma A^{(n)} \delta \leqslant \gamma B^{(n)} \delta$ if $\begin{cases} A \leqslant B \text{ and } n \text{ is even, or} \\ B \leqslant A \text{ and } n \text{ is odd} \end{cases}$ (*induced steps*)

We also use the following two *derived* inequalities.

(gcon) $\gamma A^{(n)} B^{(n+1)} \delta \leqslant \gamma\delta$, if $\begin{cases} A \leqslant B \text{ and } n \text{ is even, or} \\ B \leqslant A \text{ and } n \text{ is odd} \end{cases}$ (*generalized contraction*)

and

(gexp) $\gamma\delta \leqslant \gamma A^{(n+1)} B^{(n)} \delta$, if $\begin{cases} A \leqslant B \text{ and } n \text{ is even, or} \\ B \leqslant A \text{ and } n \text{ is odd} \end{cases}$ (*generalized expansion*)

Obviously, **(con)** and **(exp)** are particular cases of **(gcon)** and **(gexp)**, respectively. Conversely, **(gcon)** can be obtained as[4] **(ind)** followed by **(con)**, and **(gexp)** can be obtained as **(exp)** followed by **(ind)**.

Consequently, if $\alpha' \leqslant \alpha''$, then there exists a *derivation*

$$\alpha' = \gamma_0 \leqslant \gamma_1 \leqslant \cdots \leqslant \gamma_m = \alpha'', \ m \geqslant 0$$

---

1 In fact, our definition is a bit more general than that in [2].
2 '$\circ$' is usually omitted.
3 Known also as *categories*.
4 Throughout, we systematically abuse notation, by using a rule name for an application of that rule.

Download English Version:

https://daneshyari.com/en/article/426258

Download Persian Version:

https://daneshyari.com/article/426258

Daneshyari.com