Information and Computation 237 (2014) 268–293

Contents lists available at ScienceDirect

Information and Computation

www.elsevier.com/locate/yinco

An extension of context-free grammars with one-sided context specifications [☆]

Mikhail Barash [a,b], Alexander Okhotin [a,*]

[a] Department of Mathematics and Statistics, University of Turku, Turku FI-20014, Finland
[b] Turku Centre for Computer Science, Turku FI-20520, Finland

ARTICLE INFO

Article history:
Received 13 September 2013
Received in revised form 18 February 2014
Available online 24 March 2014

Keywords:
Context-free grammars
Conjunctive grammars
Contexts
Context-sensitive grammars
Parsing

ABSTRACT

The paper introduces an extension of context-free grammars equipped with an operator for referring to the left context of the substring being defined. For example, a rule $A \to a \,\&\, \lhd B$ defines a symbol a, as long as it is preceded by a string defined by B. The conjunction operator in this example is taken from conjunctive grammars (Okhotin, 2001), which are an extension of ordinary context-free grammars that maintains most of their practical properties, including many parsing algorithms. This paper gives two equivalent definitions of grammars with left contexts—by logical deduction and by language equations—and establishes their basic properties, including a transformation to a normal form and a cubic-time parsing algorithm, with a square-time version for unambiguous grammars.

1. Introduction

Context-free grammars are best understood as a logic for defining the syntax of languages. In this logic, definitions are inductive, so that the properties of a string are determined by the properties of its substrings. This is how a rule $S \to aSb$ asserts that if a string $a^{n-1}b^{n-1}$ has the property S, then the string $a^n b^n$ has the property S as well. Besides the concatenation, the formalism of this logic includes a disjunction operation, represented by having multiple rules for a single symbol. This logic can be further augmented with conjunction and negation operations, which was done by the second author [19,21] in *conjunctive grammars* and *Boolean grammars*, respectively. These grammars preserve the main idea of the context-free grammars (that of defining syntax inductively, as described above), maintain most of their practically important features, such as efficient parsing algorithms [21,23,24,27], and have been a subject of diverse research [1,8,12,13,17,28,29]. As the applicability of a rule of a Boolean grammar to a substring is independent of the context, in which the substring occurs, Boolean grammars constitute a natural general case of context-free grammars. Ordinary context-free grammars can be viewed as their disjunctive fragment. For a detailed account of the research on conjunctive and Boolean grammars, an interested reader is referred to a recent survey paper [26].

When Chomsky [6] introduced the term "context-free grammar" for an intuitively obvious model of syntax, he had a further idea of a more powerful model, in which one could define rules applicable only in some particular contexts [6, p. 142]. However, Chomsky's attempt to formalize his idea using the tools available at the time (namely, string-rewriting systems) led to nothing but space-bounded nondeterministic Turing machines; the "nonterminal symbols" in that model

[☆] A preliminary version of this paper was presented at the 6th International Conference on Language and Automata Theory and Applications (LATA 2012, A Coruña, Spain, 5–9 March 2012), and its extended abstract appeared in the conference proceedings.

[*] Corresponding author.
E-mail addresses: mikbar@utu.fi (M. Barash), alexander.okhotin@utu.fi (A. Okhotin).

http://dx.doi.org/10.1016/j.ic.2014.03.003
0890-5401/© 2014 Elsevier Inc. All rights reserved.

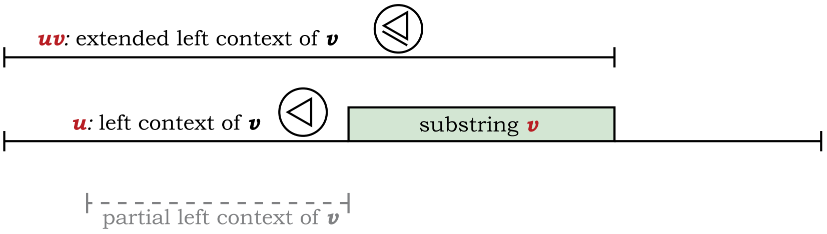

Fig. 1. A substring v with a left context u, denoted by $u\langle v\rangle$, and context operators applied to it.

no longer represent any syntactic classes, but are simply bits in the memory of those Turing machines. Even though the resulting devices are still known under the name of "context-sensitive grammars", they have nothing to do with the syntax of languages, and, in particular, they fail to implement Chomsky's original idea of a phrase-structure rule applicable in a context.

Even though Chomsky's terminology for formal grammars, such as the term "context-free", was generally accepted by the research community, the actual idea of a rule applicable in a context was never investigated again. None of the successful extensions of context-free grammars, such as tree-adjoining grammars [14] or multi-component grammars [36,33], allow expressing any conditions on the contexts—in spite of the nickname "mildly context-sensitive" [15]. It should also be noted that both tree-adjoining grammars and multi-component grammars define the properties of strings inductively on their length, and have nothing to do with Chomsky's "context-sensitive" string rewriting. Thus, the theory of formal grammars beyond ordinary context-free has developed in a different direction than initially pointed by Chomsky. Nevertheless, the general idea of context-sensitive rules in formal grammars remains interesting and deserves investigation.

This paper undertakes to reconsider Chomsky's [6] idea of contexts in grammars, this time using the more appropriate tools of deduction systems and language equations. The concept of a formal grammar as a logic and its semantics as logical inference was first presented in a monograph by Kowalski [18, Ch. 3], and then elaborated by Pereira and Warren [31]. Later, Rounds [32] used first-order logic over positions in a string augmented with a fixpoint operator, FO(LFP), to represent formal grammars as formulae of this logic. What is particularly important about this representation, is that all the aforementioned successful extensions of the context-free grammars, such as tree-adjoining grammars and multi-component grammars, are not only expressible in this logic, but the way they are expressed represents these grammars exactly as they are intuitively understood.

This paper derives the general outlook on grammars from the cited work [31,32], and draws from the experience of developing the conjunctive grammars [26] to define the desired grammar model. The new model proposed in this paper are *grammars with one-sided contexts*, which are based on conjunctive grammars and introduce two special operators for referring to the context of a substring being defined. The *left context operator* refers to the "past" of the current substring: an expression $\lhd\alpha$ defines any substring that is directly preceded by a prefix of the form α. This operator is meant to be used together with usual specifications of the structure of the current substring, using conjunction to combine several specifications. For example, consider the rule $A \to BC \,\&\, \lhd D$, which represents any substring of the form BC preceded by a substring of the form D. If the grammar contains additional rules $B \to b$, $C \to c$ and $D \to d$, then the above rule for A specifies that a substring bc of a string $w = dbc\ldots$ has the property A; however, this rule will not produce the same substring occurring in the strings $w' = abc$ or $w'' = adbc$. The other *extended left context operator* $\trianglelefteq\alpha$ represents the form of the left context of the current substring concatenated with the substring itself, so that the rules $A \to B \,\&\, \trianglelefteq E$, $B \to b$, $E \to ab$ state that the substring b occurring in the string $w = ab$ has the property A. Fig. 1 illustrates how context operators refer to a substring and its left context. One can symmetrically define operators for right contexts $\rhd\alpha$ and extended right contexts $\trianglerighteq\alpha$.

Note that the proposed context operators apply to the whole left context, which begins with the first symbol of the entire string. One could argue for using a partial left context instead, that is, any substring ending where the substring being defined begins (illustrated at the bottom of Fig. 1). Such a partial context of the form D can be easily described using the proposed operator as $\lhd\Sigma^*D$, where Σ^* stands for an arbitrary string. On the other hand, the whole left context can be simulated by a partial left context, provided that the entire string begins with a special marker symbol: if a partial context beginning with that symbol is requested, then it can only be the whole left context. Thus, a partial left context operator would be less convenient to use, as well as not any easier to implement than the proposed operator for the whole left context.

In the literature, related ideas have occasionally arisen in connection with parsing, where (extended) right contexts of the form $\trianglerighteq\alpha\Sigma^*$ (in the terminology of this paper) are considered as "lookahead strings" and are used to guide a deterministic parser. If α represents a regular language, these simple forms of contexts occur in LR-regular [7], LL-regular [11] and LL($*$) [30] parsers. Some software tools for engineering parsers, such as those developed by Parr and Fischer [30] and by Ford [9], allow specifying contexts $\trianglerighteq\alpha\Sigma^*$, with α defined within the grammar, and such specifications can be used by a programmer for *ad hoc* adjustment of the behaviour of a deterministic recursive descent parser.

In this paper, the above intuitive definition of grammars with one-sided contexts is formalized in two equivalent ways. The first possibility, pursued in Section 2, is to consider deduction of elementary propositions of the form $A(u\langle v\rangle)$, where