数理解析研究所講究録
第 1599 巻 2008 年 103-110

Some Additional Remarks on Grammatical Characterizations of Alternating PDAs*

Etsuro Moriya (早稲田大学・教育学部 守屋 悦朗)
Department of Mathematics, School of Education
Waseda University, Shinjuku-ku, Tokyo, 169-8050
and
Friedrich Otto
Fachbereich Elektrotechnik/Informatik, Universität Kassel
34109 Kassel, Germany

Abstract

We study several extensions of the notion of alternation from context-free grammars to context-sensitive and arbitrary phrase-structure grammars. Thereby new grammatical characterizations are obtained for the class of languages that are accepted by alternating pushdown automata.

In this paper we consider two different ways for defining derivation relation in alternating phrase-structure grammars to prove that they are in fact equivalent in a weak sense. Some other results stated in this paper have been reported in [11] without proof.

1 Introduction

Alternation is a powerful concept that was first introduced by Chandra and Stockmeyer [1, 2] for general Turing machines and then by Ladner, Lipton, and Stockmeyer [6, 7] for pushdown automata. Thereafter this notion has been studied for a variety of other devices. In particular, in [9] one of the authors introduced the concept of *alternating context-free grammars* (ACFG for short) by distinguishing between *existential* and *universal variables* (nonterminals) with the aim of deriving a grammatical characterization for the class of languages that are accepted by alternating pushdown automata (APDA for short).

As no such characterization was obtained in [9], further studies of the notion of alternation for context-free grammars and pushdown automata followed (see, e.g., [3, 10, 12]). Also Okhotin's *conjunctive grammars* [14] can be interpreted as a variant of ACFGs in which the effect of universal steps is localized. In [4] the class of languages that are accepted by APDAs was finally characterized through *linear-erasing* ACFGs. Further, inspired by the notion of *context-free grammar with states* of Kasai [5], the *state-alternating context-free grammar* (sACFG for short) was introduced in [10] by distinguishing between *existential* and *universal states*. Thus, while in an ACFG the variable on the lefthand side of a production determines whether this production is to be used in an existential or a universal fashion, it is the states that make this distinction in an sACFG. For each ACFG G, an sACFG G' can be constructed such that G and G' generate the same language, but it is still open whether or not the converse is true. At least for linear context-free grammars, and therewith in particular for

*This paper is intended to summarize the authors' original paper [13] by omitting the proofs for all the results.

right-linear (that is, regular) grammars, it has been shown that the two notions of alternation have the same expressive power. Actually, both types of alternating right-linear grammars just generate the regular languages. Further, it turned out that sACFGs working in leftmost derivation mode generate exactly those languages that are accepted by APDAs [10]. In this way another grammatical characterization for this class of languages was obtained.

In [12] the authors studied a different way of defining the notion of alternation for pushdown automata. Instead of distinguishing between existential and universal states as in [6, 7], here the pushdown symbols are used for this purpose. The *stateless* variant of this so-called *stack-alternating pushdown automaton* accepts exactly those languages that are generated by ACFGs in leftmost derivation mode [12]. However, in general *stack-alternating pushdown automata* are equivalent in expressive power to the original variant of the APDA. It is known that the class of languages these automata accept coincides with the deterministic time complexity class $\mathsf{ETIME} = \bigcup_{c>0} \mathsf{DTIME}(c^n)$ as well as with the alternating space complexity class ALINSPACE, that is, the class of languages that are accepted by *alternating linear bounded automata* (ALBA) [2, 7]. As in the classical (non-alternating) setting pushdown automata correspond to context-free grammars and linear bounded automata correspond to context-sensitive grammars, the above results raise the question about the expressive power of alternating context-sensitive grammars.

In this paper we carry the notion of alternation over to general phrase structure and context-sensitive grammars. In fact, we consider both types of alternation for grammars mentioned above. By distinguishing between existential and universal variables we obtain the *alternating phrase-structure grammars* (APSG) and the *alternating context-sensitive grammars* (ACSG). By considering grammars with states, for which we distingush between existential and universal states, we obtain the *state-alternating phrase-structure grammars* (sAPSG) and the *state-alternating context-sensitive grammars* (sACSG). For state-alternating grammars it is rather straightforward to define the notion of derivation. However, for the other type of alternating grammars there are various different ways for defining the corresponding derivation relation. We will consider two such definitions, and we will prove that they are in fact equivalent in a weak sense, that is, for a fixed alternating grammar the two definitions yield different languages, but to each alternating grammar working with the one notion of derivation, there is another grammar of the same type that is working with the other notion of derivation, and that generates the same language. In addition, we will consider two modes of derivation: *leftmost* derivations and *unrestricted* derivations.

Actually, it will turn out that for phrase-structure and for context-sensitive grammars, the state-alternating variant is equivalent to the alternating variant. This equivalence is valid for both leftmost derivations and unrestricted derivations. With respect to unrestricted derivations APSGs just give another characterization for the class RE of recursively enumerable languages. However, with respect to leftmost derivations, they have the same generative power as sACFGs. This can be interpreted as the counterpart to the corresponding result for non-alternating grammars, which states that in leftmost mode general phrase-structure grammars can only generate context-free languages [8]. Our second main result states that with respect to unrestricted derivations ACSGs generate exactly those languages that are accepted by alternating linear bounded automata. As ALBAs and APDAs accept the same languages, we see that APSGs (working in leftmost mode) and ACSGs (working in unrestricted mode) give new grammatical characterizations for the class of languages that are accepted by APDAs. Finally, when working in leftmost mode, ACSGs generate a subclass of this class of languages. It remains open, however, whether this is a proper subclass. These facts should be compared to the fact that no inclusion relation is known between the class of languages generated by sACFGs (or ACFGs) in leftmost mode and the class of languages generated by sACFGs (or ACFGs) in unrestricted mode.

In Section 2 the basic definitions of alternating and state-alternating grammars are given, and two different ways of defining the notion of *derivation* for alternating grammars are considered. In the remaining sections we state without proof relationships among various types of alternating grammars and state-alternating grammars.

2 Two Types of Alternating Grammars

An *alternating phrase-structure grammar* is a grammar $G = (V, U, \Sigma, P, S)$, where V is a set of variables (or nonterminals), $U \subseteq V$ is a set of *universal* variables, while the variables in $V \setminus U$ are called *existential*, Σ is a set of terminals, S is the start symbol, and P is a set of productions, where $(\ell, r) \in P$ implies that $\ell, r \in (V \cup \Sigma)^*$, and ℓ contains at least one variable. If $|\ell| \leq |r|$ holds for all productions $(\ell, r) \in P$, then G is called an *alternating context-sensitive grammar*, and if $\ell \in V$ holds for all productions $(\ell, r) \in P$, then G is called an *alternating context-free grammar.* By APSG (ACSG, ACFG) we denote the class of all alternating phrase-structure(context-sensitive, context-free) grammars.

It remains to specify the way in which derivations are performed by an alternating grammar G. In particular, we must determine a way to distinguish between existential and universal derivation steps. There are various options.

First of all we can use a specific nonterminal occurring in a sentential form α to determine whether α itself is existential or universal. For example, we could use the leftmost variable occurring in α for that, that is, if $\alpha = xA\beta$, where $x \in \Sigma^*$, $A \in V$, and $\beta \in (V \cup \Sigma)^*$, then we call α an *existential sentential form* if $A \in V \setminus U$, and we call α a *universal sentential form* if $A \in U$. To apply a derivation step to α, we nondeterministically choose a substring ℓ of α that occurs as the left-hand side of one or more rules of P. Now if α is existential, then one of these rules is chosen, and $\alpha = \gamma\ell\delta$ is rewritten into $\gamma r\delta$, where $(\ell, r) \in P$ is the rule chosen. If α is universal, then let $(\ell, r_1), \ldots, (\ell, r_m)$ be those rules of P with left-hand side ℓ. Now all these productions are applied simultaneously, thus giving a finite number of successor sentential forms $\gamma r_1\delta, \ldots, \gamma r_m\delta$. In this way a derivation is not a linear chain, but it has the form of a tree. A terminal word w can be derived from G, if there exists a finite derivation tree in the above sense such that the root is labelled with the start symbol S and all leaves are labelled with w. Observe that in this way the rules themselves are neither existential nor universal, but that it purely depends on the type of the leftmost variable in the actual sentential form whether the next derivation step is existential or universal. Below we will use the notation $\Rightarrow_G^c$ to denote this derivation relation. By $L^c(G)$ we denote the language that is generated by G using this relation.

Alternatively, we can use a distinguished occurrence of a variable in the left-hand side of a rule to declare that rule as being existential or universal. Of course, this must be done in a consistent way, that is, for all rules with the same left-hand side, the same variable occurrence must be chosen. Then for $\alpha = \gamma\ell\delta$, if ℓ is existential, then α is rewritten into $\gamma r\delta$, where $(\ell, r) \in P$ is one of the rules with left-hand side ℓ, and if ℓ is universal, then α is rewritten simultaneously into $\gamma r_1\delta, \ldots, \gamma r_m\delta$, where $(\ell, r_1), \ldots, (\ell, r_m)$ are all the rules in P with left-hand side ℓ. Here we use the following convention: ℓ is universal (or existential, resp.) if the leftmost variable occurring in ℓ is universal (or existential, resp.). We will use the notation $\Rightarrow_G$ to denote this derivation relation. By $L(G)$ we denote the language that is generated by G using this derivation relation.

The following example demonstrates that the derivation relations $\Rightarrow_G^c$ and $\Rightarrow_G$ will in general yield different languages.

Example 2.1. Let $G = (\{S, A, B\}, \{B\}, \{a, b, c\}, P, S)$ with $P = \{S \to AB, A \to a, A \to ab, B \to c, B \to bc\}$. Then with respect to $\Rightarrow^c_G$, G generates the language $L^c(G) = \{ac, abc, abbc\}$, while with respect to $\Rightarrow_G$, we only obtain the language $L(G) = \{abc\}$. The reason is the fact that with respect to $\Rightarrow^c_G$, the rules with left-hand side B can be applied in existential fashion as long as the variable A is still present in the actual sentential form.

However we have the following results [13].

Proposition 2.2. *For each alternating phrase-structure grammar G, there exists an alternating phrase-structure grammar G' such that $L(G') = L^c(G)$.*

Proposition 2.3. *For each alternating phrase-structure grammar G, there exists an alternating phrase-structure grammar G' such that $L^c(G') = L(G)$.*

These results also hold for the special case of alternating context-sensitive grammars.

Proposition 2.4. *For each alternating context-sensitive grammar G, there exists an alternating context-sensitive grammar G' such that $L(G') = L^c(G)$.*

Proposition 2.5. *For each alternating context-sensitive grammar G, there exists an alternating context-sensitive grammar G' such that $L^c(G') = L(G)$.*

Thus, we see that for context-sensitive as well as for general phrase-structure grammars, both definitions of alternation yield the same expressive power. Therefore we restrict our attention in the rest of this paper to alternating grammars for which the leftmost variable occurring in the lefthand side of a production determines whether the production itself is existential or universal.

In addition to the unrestricted derivation mode, we are also interested in the so-called *leftmost* derivation mode. A derivation step $\alpha = \gamma\ell\delta \Rightarrow_G \beta$, respectively $\alpha = \gamma\ell\delta \Rightarrow_G (\gamma r_1\delta, \ldots, \gamma r_m\delta)$, is called *leftmost* if $\gamma \in \Sigma^*$, that is, this step involves the leftmost variable occurrence in α. By $L_{\mathsf{lm}}(G)$ we denote the language consisting of all terminal words that G generates by leftmost derivations. It is obvious that with respect to leftmost derivations the above two definitions of the derivation process of an alternating grammar coincide, if in both definitions the leftmost variable occurrence is chosen.

In [8] it is shown that the language $L_{\mathsf{lm}}(G)$ is context-free if $G = (V, \Sigma, S, P)$ is a phrase-structure grammar such that each rule $(\ell \to r) \in P$ has the structure

$$\ell = x_0A_1x_2 \cdots x_{n-1}A_nx_n \to x_0\beta_1x_2 \cdots x_{n-1}\beta_nx_n = r$$

for some $n \geq 1$, where $x_0, x_i \in \Sigma^*$, $A_i \in V$, and $\beta_i \in (V \cup \Sigma)^*$ for all $1 \leq i \leq n$. To obtain a corresponding result, we restrict our attention to alternating phrase-structure grammars $G = (V, U, \Sigma, P, S)$ that satisfy the following condition, when we consider leftmost derivations: each rule $(\ell \to r) \in P$ has the form $\ell = xA\alpha \to x\beta = r$, where $x \in \Sigma^*$, $A \in V$, and $\alpha, \beta \in (V \cup \Sigma)^*$. It is likely that this restriction limits the expressive power of alternating grammars, but this question remains to be studied in detail. Obviously, this restriction contains the above restriction as a special case, and it is satisfied by all grammars for which the lefthand side of each production begins with a nonterminal.

We denote the class of languages generated by grammars of type X in leftmost derivation mode by $\mathcal{L}_{\mathsf{lm}}(\mathsf{X})$, while $\mathcal{L}(\mathsf{X})$ is used to denote the class of languages generated by these grammars in unrestricted derivation mode.

In [10] also the *state-alternating context-free grammar* (sACFG) was introduced. Analogously, we define the *state-alternating phrase-structure grammar* as an 8-tuple $G = (Q, U, V, \Sigma, P, S, q_0, F)$, where Q is a finite set of states, $U \subseteq Q$ is a set of *universal states*, while the states in $Q \setminus U$ are called *existential states*, V is a finite set of variables, Σ is a set of terminals, $S \in V$ is the start symbol, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is a set of final states. Finally, P is a finite set of productions of the form $(p, \ell) \rightarrow (q, r)$, where $p, q \in Q$, $\ell \in (V \cup \Sigma)^* \cdot V \cdot (V \cup \Sigma)^*$, and $r \in (V \cup \Sigma)^*$. The *derivation relation* $\Rightarrow_G^*$ is defined on the set $Q \times (V \cup \Sigma)^*$ of *extended sentential forms*. Let $p \in Q$ and $\alpha \in (V \cup \Sigma)^*$. If p is an existential state, that is, $p \in Q \setminus U$, then $(p, \alpha) \Rightarrow_G (q, \alpha_1 r \alpha_2)$, if $\alpha = \alpha_1 \ell \alpha_2$, and there exists a production of the form $(p, \ell) \rightarrow (q, r)$. If p is a universal state, α has the factorization $\alpha = \alpha_1 \ell \alpha_2$, and $(p, \ell) \rightarrow (q_i, r_i)$ $(1 \leq i \leq k)$ are all the productions with lefthand side (p, ℓ), then $(p, \alpha) \Rightarrow_G ((q_1, \alpha_1 r_1 \alpha_2), \ldots, (q_k, \alpha_1 r_k \alpha_2))$, that is, all these productions are applied in parallel to the chosen occurrence of the substring ℓ, and following this step all these sentential forms are rewritten further, independently of each other. In this way a derivation tree is obtained.

The language $L(G)$ that is generated by G consists of all words $w \in \Sigma^*$ for which there exists a derivation tree such that the root is labelled with (q_0, S) and all leaves are labelled with pairs of the form (p, w) with $p \in F$. Note that the labels of different leaves may differ in their first components.

If $|\ell| \leq |r|$ holds for all productions $(p, \ell) \rightarrow (q, r)$ of P, then G is called a *state-alternating context-sensitive grammar*, and if $\ell \in V$ for all productions $(p, \ell) \rightarrow (q, r)$ of P, then G is a *state-alternating context-free grammar*. By sACFG, sACSG, and sAPSG we denote the classes of state-alternating context-free, context-sensitive, and general phrase-structure grammars, respectively. As before we are interested in the expressive power of these grammars with respect to the leftmost and the unrestricted derivation modes. It is known that the class of languages $\mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG})$ coincides with the class of languages that are accepted by alternating pushdown automata ([10] Theorem 6.4).

3 Alternation Versus State-Alternation

First we consider the generative power of alternating grammars with respect to the leftmost derivation mode. Recall that we require that each production of an alternating grammar is of the form $(xA\alpha \rightarrow x\beta)$, where $x \in \Sigma^*$, $A \in V$, and $\alpha, \beta \in (V \cup \Sigma)^*$. For state-alternating grammars, we require analogously that each production is of the form $((p, xA\alpha) \rightarrow (q, x\beta))$, where p and q are states.

Lemma 3.1. $\mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{sACSG})$.

An analogous result holds for alternating phrase-structure grammars.

Lemma 3.2. $\mathcal{L}_{\mathsf{lm}}(\mathsf{APSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{sAPSG})$.

However, for APSGs we even have the following result.

Lemma 3.3. $\mathcal{L}_{\mathsf{lm}}(\mathsf{APSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG})$.

Next we see that also the converse of Lemma 3.1 holds.

Lemma 3.4. $\mathcal{L}_{\mathsf{lm}}(\mathsf{sACSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG})$.

Combining Lemmas 3.1 and 3.4 we obtain the following equivalence.

Theorem 3.5. $\mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG}) = \mathcal{L}_{\mathsf{lm}}(\mathsf{sACSG})$.

The proof above can also be adapted to the case of alternating phrase-structure grammars, which yields the following result.

Lemma 3.6. $\mathcal{L}_{\mathsf{lm}}(\mathsf{sAPSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{APSG})$.

From Lemmas 3.3 and 3.6 and the facts that $\mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{sAPSG})$ and that $\mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG}) = \mathcal{L}(\mathsf{APDA})$ [10] we obtain the following equivalence.

Theorem 3.7. $\mathcal{L}_{\mathsf{lm}}(\mathsf{APSG}) = \mathcal{L}_{\mathsf{lm}}(\mathsf{sAPSG}) = \mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG}) = \mathcal{L}(\mathsf{APDA})$.

As $\mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{APSG})$ holds, Theorem 3.5 and Theorem 3.7 yield the following consequence.

Corollary 3.8. $\mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG}) = \mathcal{L}_{\mathsf{lm}}(\mathsf{sACSG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG}) = \mathcal{L}(\mathsf{APDA})$.

Now we turn to the unrestricted derivation mode. We have the following equalities, where RE denotes the class of recursively enumerable languages.

Corollary 3.9. (a) $\mathcal{L}(\mathsf{ACSG}) = \mathcal{L}(\mathsf{sACSG})$.
(b) $\mathcal{L}(\mathsf{APSG}) = \mathcal{L}(\mathsf{sAPSG}) = \mathsf{RE}$.

4 ACSGs and Alternating Linear Bounded Automata

An *alternating linear bounded automaton*, ALBA for short, M is a linear bounded automaton for which some of its states are distinguished as *universal states*.

It is known that $\mathcal{L}(\mathsf{ALBA}) = \mathcal{L}(\mathsf{APDA})$ [2, 7]. The next lemma shows that ACSGs are of sufficient expressive power to generate all languages that are accepted by ALBAs.

Lemma 4.1. $\mathcal{L}(\mathsf{ALBA}) \subseteq \mathcal{L}(\mathsf{sACSG})$.

Also we have the converse of Lemma 4.1, which can also be proved by an appropriate modification of the standard construction of a linear bounded automaton from a monotone grammar.

Lemma 4.2. $\mathcal{L}(\mathsf{sACSG}) \subseteq \mathcal{L}(\mathsf{ALBA})$.

Thus, we obtain the following theorem.

Theorem 4.3. $\mathcal{L}(\mathsf{sACSG}) = \mathcal{L}(\mathsf{ALBA})$.

By Corollary 3.9 (a) this yields the following consequence.

Corollary 4.4. $\mathcal{L}(\mathsf{ACSG}) = \mathcal{L}(\mathsf{sACSG}) = \mathcal{L}(\mathsf{ALBA}) = \mathcal{L}(\mathsf{APDA}) = \mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG})$.

From Corollaries 3.8 and 4.4, the following inclusion follows.

Corollary 4.5. $\mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG}) \subseteq \mathcal{L}(\mathsf{ACSG})$.

It remains to consider the converse of the above inclusion. By Corollary 4.4 this is equivalent to the question of whether the inclusion $\mathcal{L}_{\mathsf{lm}}(\mathsf{sACFG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG})$ holds. Unfortunately this problem remains unsettled in this paper. As each ε-free sACFG is context-sensitive, at least the following special case holds.

Corollary 4.6. $\mathcal{L}_{\mathsf{lm}}(\varepsilon\text{-free } \mathsf{sACFG}) \subseteq \mathcal{L}_{\mathsf{lm}}(\mathsf{ACSG})$.

The diagram in Figure 1 depicts the inclusion relations among the classes of languages we have discussed in this paper.

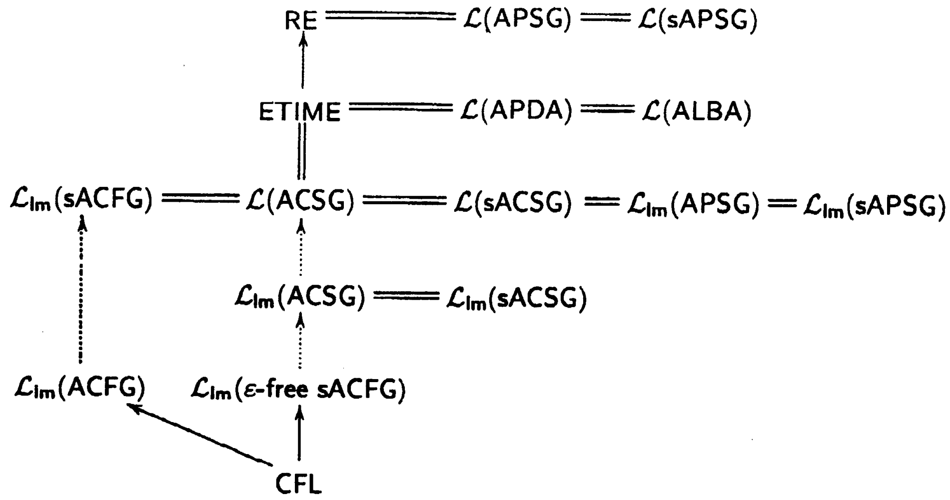

Figure 1: Inclusion relations among language classes defined by various types of alternating grammars. An arrow denotes a proper inclusion, while a dotted arrow denotes an inclusion that is not known to be proper.

5 Concluding Remarks

We have generalized the notion of alternation from context-free grammars to general phrase-structure grammars. Our main result shows that with respect to the leftmost derivation mode alternating phrase-structure grammars are just as expressive as state-alternating context-free grammars, and that alternating context-sensitive grammars working in the unrestricted derivation mode have the same expressive power, too. In this way we have obtained new grammar-based characterizations for the class of languages that are accepted by alternating pushdown automata.

Acknowledgement. The first author was supported in part by Waseda University Grant for Special Research Projects #2006B-073, which he gratefully acknowledges. The authors thank Hartmut Messerschmidt from Universität Kassel for fruitful discussions on the notions and results presented in this paper.

References

[1] A.K. Chandra and L.J. Stockmeyer. Alternation. In: *Proc. 17th FOCS*, pp. 98–108. IEEE Computer Society Press, 1976.

[2] A.K. Chandra, D.C. Kozen, and L.J. Stockmeyer. Alternation. *J. Assoc. Comput. Mach.*, 28: 114–133, 1981.

[3] Z.Z. Chen and S. Toda. Grammatical characterizations of P and PSPACE. *Transactions IEICE*, E 73:1540–1548, 1990.

[4] O.H. Ibarra, T. Jiang, and H. Wang. A characterization of exponential-time languages by alternating context-free grammars. *Theoret. Comput. Sci.*, 99: 301–313, 1992.

[5] T. Kasai. An infinite hierarchy between context-free and context-sensitive languages. *J. Comput. Syst. Sci.*, 4: 492–508, 1970.

[6] R.E. Ladner, R.J. Lipton, and L.J. Stockmeyer. Alternating pushdown automata. In: *Proc. 19th FOCS*, pp. 92–106. IEEE Computer Society Press, 1978.

[7] R.E. Ladner, R.J. Lipton, and L.J. Stockmeyer. Alternating pushdown and stack automata. *SIAM J. Comput.*, 13: 135–155, 1984.

[8] G. Matthews. A note on symmetry in phrase structure grammars. *Inform. Control*, 7: 360–365, 1964.

[9] E. Moriya. A grammatical characterization of alternating pushdown automata. *Theoret. Comput. Sci.*, 67: 75–85, 1989.

[10] E. Moriya, D. Hofbauer, M. Huber, and F. Otto. On state-alternating context-free grammars. *Theoret. Comput. Sci.*, 337: 183–216, 2005.

[11] 守屋悦朗, Alternating CFG の拡張について, 京都大学数理解析研究所講究録 1554, pp.9-15, 2007 年 5 月.

[12] E. Moriya and F. Otto. Two ways of introducing alternation into context-free grammars and pushdown automata. *Transactions IEICE*, E 90-D: 889–894, 2007.

[13] E. Moriya and F. Otto. On alternating non-context-free gammars. *Kasseler Informatik Schriften* 2007/6. Fachbereich Elektrotechnik/Informatik, Universität Kassel, 2007. Available at: `https://kobra.bibliothek.uni-kassel.de/handle/urn:nbn:de:hebis:34-2007110719587`.

[14] A. Okhotin. Conjunctive grammars. *J. Auto. Lang. Comb.*, 6:519–535, 2001.