LL RANDOM CONTEXT GRAMMARS

Lukáš Vrábel
Doctoral Degree Programme (4), FIT BUT
E-mail: xvrabe01@stud.fit.vutbr.cz

Supervised by: Alexander Meduna
E-mail: meduna@fit.vutbr.cz

Abstract: The present paper investigates LL versions of random context grammars. It proves that they generate the family of LL context-free languages, thus solving the question whether we can build a more powerful LL parser using this model. A formulation of two open problems closes the paper.

Keywords: regulated rewriting, context-free grammars, random context grammars, LL versions, generative power, parsing

1 INTRODUCTION

Over its history, formal language theory has always systematically and intensively investigated regulated grammars (see [4], Chapter 3 in [10], and Chapter V in [11] for an overview of this investigation). Indisputably, random context grammars are central to this investigation as demonstrated by a great number of studies solely dedicated to them (see [2–4, 7–9, 12–15]).

Recall that *random context grammars* are based upon context-free rules just like context-free grammars. However, each rule is extended by a set of *permitting symbols* and a set of *forbidding symbols*. A rule like this can rewrite a nonterminal if each of its permitting symbols occurs in the current sentential form while any of its forbidding symbols does not occur there. *LL random context grammars*, introduced in this paper, represent ordinary random context grammars restricted by analogy with LL requirements placed upon LL context-free grammars. That is, by analogy with LL context-free grammars, (1) LL random context grammars always rewrite the leftmost nonterminal in the current sentential form during every derivation step, and (2) if there are two or more applicable rules with the same nonterminal on their left-hand sides, then the sets of all terminals that can begin a string obtained by a derivation started by using these rules are disjoint.

Recall that although random context grammars generate the family of recursively enumerable languages (see Theorem 1.2.3 in [4]), random context grammars that work in the leftmost way generate only the family of context-free languages (see Theorem 1.4.1 in [4]). Of course, it is only natural to ask whether LL random context grammars are more powerful than LL context-free grammars (CFG). In case of affirmitive answer, we could build a deterministic parser running in linear time that would be more powerful than a LL parser based on context-free grammar. Unfortunately, this paper describes transformations that convert any LL random context grammar to an equivalent LL context-free grammar and conversely, thus proving the opposite.

2 PRELIMINARIES AND DEFINITIONS

In this paper, we assume that the reader is familiar with the theory of formal languages (see [4, 5]), including the theory of parsing (see [1, 6]). For a set Q, $\mathrm{card}(Q)$ denotes the cardinality of Q, and 2^Q denotes the power set of Q. For an alphabet (finite nonempty set) V, V^* represents the free monoid generated by V under the operation of concatenation. The unit of V^* is denoted by ε. Set $V^+ = V^* - \{\varepsilon\}$; algebraically, V^+ is thus the free semigroup generated by V under the operation of concatenation. For $x \in V^*$, $|x|$ denotes the length of x, and $\mathrm{alph}(x)$ denotes the set of symbols occurring in x.

RANDOM CONTEXT GRAMMARS

Since we pay principal attention to random context grammars working in the leftmost way, we directly define them so that they always rewrite the leftmost nonterminal in the current sentential form. Furthermore, in what follows, by a random context grammar, we always mean a random context grammar working in this leftmost way.

Definition 1 (see [4]). A *random context grammar* (an *RCG* for short) is a quadruple $G = \big(N, T, P, S\big)$ where N and T are two disjoint alphabets, $S \in N$, and $P \subseteq N \times \big(N \cup T\big)^* \times 2^N \times 2^N$ is a finite relation.

Set $V = N \cup T$. Each $(A, x, U, W) \in P$ is written as $\lfloor A \rightarrow x, U, W \rfloor$ throughout this paper. For $\lfloor A \rightarrow x, U, W \rfloor \in P$, U and W are called the *permitting context* and the *forbidding context*, respectively. □

Next, we define the leftmost direct derivation relation and the generated language.

Definition 2 (see [4]). Let $G = (N,\ T,\ P,\ S)$ be an RCG. A rule $\lfloor A \rightarrow x, U, W \rfloor \in P$ is *applicable* to $y \in V^*$ if and only if $y = uAv$, where $u \in T^*$ and $v \in V^*$, and $\lfloor A \rightarrow x, U, W \rfloor \in P, U \subseteq \mathrm{alph}(v)$ and $W \cap \mathrm{alph}(v) = \emptyset$

The *leftmost direct derivation relation* over V^*, symbolically denoted by $\Rightarrow_G$, is defined as follows: $y \Rightarrow_G w\ [r]$ if and only if $y = uAv$, $w = uxv$, and there is $r = \lfloor A \rightarrow x, U, W \rfloor \in P$ that is applicable to y.

In the standard way, based on $\Rightarrow_G$, we define $\Rightarrow_G^k$ for $k \geq 0$ and $\Rightarrow_G^*$. The *language of G* is denoted by $L(G)$ and defined as $L(G) = \big\{ w \in T^* \mid S \Rightarrow_G^* w \big\}$ □

By analogy with the Predict set in CFGs, we introduce such a set to RCGs. It is then used to define LL RCGs. Notice that in contrary to CFGs, the applicability of a random context rule $\lfloor A \rightarrow x, U, W \rfloor$ depends not only on the presence of A in the current sentential form but also on the presence and absence of symbols from U and W, respectively. This has to be properly reflected in the definition.

Definition 3. Let $G = (N, T, P, S)$ be an RCG. For every $r = \lfloor A \rightarrow x, U, W \rfloor \in P$, define $\mathrm{Predict}(r) \subseteq T$ as follows: $a \in \mathrm{Predict}(r)$ if and only if $S \Rightarrow_G^* uAv \Rightarrow_G uxv \Rightarrow_G^* uaw$ where $v, x, w \in V^*$, $u \in T^*$, and r is applicable to uAv. □

Based on the above definition, we now define LL RCGs.

Definition 4. Let $G = (N, T, P, S)$ be an RCG. G is an *LL RCG* if it satisfies the following condition: for any $p = \lfloor A \rightarrow x, U, W \rfloor, r = \lfloor A \rightarrow x', U', W' \rfloor \in P$ such that $p \neq r$, if $\mathrm{Predict}(p) \cap \mathrm{Predict}(r) \neq \emptyset$, then for all w such that $S \Rightarrow_G^* w$, either p is applicable to w or r is applicable to w, but not both. □

3 PROOF OF THE RESULT

In this section, we prove that LL RCGs characterize the family of LL context-free languages. First, we show how to transform any LL RCG into an equivalent LL CFG.

Lemma 1. *For every LL RCG G, there is an LL CFG H such that $L(H) = L(G)$.*

Proof. Let $G = (N,\ T,\ P,\ S)$ be an LL RCG. In what follows, symbols $\langle$ and $\rangle$ are used to clearly unite more symbols into a single compound symbol. Construct the CFG $H = \big(N', T, P', \langle S, \emptyset \rangle\big)$ in the following way. Initially, set $N' = \big\{\langle A, Q\rangle \mid A \in N, Q \subseteq N\big\}$ and $P' = \emptyset$. Without any loss of generality, we assume that $N' \cap (N \cup T) = \emptyset$. Now, for each

$$\lfloor A \rightarrow y_0 Y_1 y_1 Y_2 y_2 \cdots Y_h y_h, U, W \rfloor \in P$$

where $y_i \in T^*$, $Y_j \in N$, for all i and j, $0 \leq i \leq h$, $1 \leq j \leq h$, for some $h \geq 0$, and for each $\langle A, Q\rangle \in N'$ such that $U \subseteq Q$ and $W \cap Q = \emptyset$, add the following rule to P':

$$\begin{array}{rcl} \langle A, Q\rangle & \rightarrow & y_0 \langle Y_1, Q \cup \{Y_2, Y_3, \ldots, Y_h\}\rangle y_1 \\ & & \langle Y_2, Q \cup \{Y_3, \ldots, Y_h\}\rangle y_2 \\ & & \quad\vdots \\ & & \langle Y_h, Q\rangle y_h \end{array}$$

Before proving that $L(H) = L(G)$, let us give an insight into the construction. As G always rewrites the leftmost occurrence of a nonterminal, we use compound nonterminals of the form $\langle A, Q\rangle$ in H, where A is a nonterminal, and Q is a set of nonterminals that appear to the right of this occurrence of A. When simulating rules from P, the check for the presence and absence of symbols is accomplished by using Q. Also, when rewriting A in $\langle A, Q\rangle$ to some y, the compound nonterminals from N' are generated instead of nonterminals from N.

The proof of the identity $L(H) = L(G)$ is divided into two claims. First, Claim 1 shows how derivations of G are simulated by H. Then, Claim 2 demonstrates the converse—that is, it shows how G simulates derivations of H.

Set $V = N \cup T$ and $V' = N' \cup T$. Define the homomorphism τ from V'^* to V^* as $\tau(\langle A, Q\rangle) = A$ for all $A \in N$ and $Q \subseteq N$, and $\tau(a) = a$ for all $a \in T$.

Claim 1. *If $S \Rightarrow_G^k x$, where $x \in V^*$ and $k \geq 0$, then $\langle S, \emptyset\rangle \Rightarrow_H^* x'$, where $\tau(x') = x$ and x' is of the form*

$$x' = x_0 \langle X_1, \{X_2, X_3, \ldots, X_n\}\rangle x_1 \langle X_2, \{X_3, \ldots, X_n\}\rangle x_2 \cdots \langle X_n, \emptyset\rangle x_n$$

where $X_i \in N$ for $i = 1, 2, \ldots, n$ and $x_j \in T^*$ for $j = 0, 1, \ldots, n$, for some $n \geq 0$.

Proof. This claim is established by induction on $k \geq 0$.

Basis. Let $k = 0$. Then, for $S \Rightarrow_G^0 S$, $\langle S, \emptyset\rangle \Rightarrow_H^0 \langle S, \emptyset\rangle$, so the basis holds.

Induction Hypothesis. Suppose that there exists $k \geq 0$ such that the claim holds for all derivations of length ℓ, where $0 \leq \ell \leq k$.

Induction Step. Consider any derivation of the form $S \Rightarrow_G^{k+1} w$, where $w \in V^*$. Since $k+1 \geq 1$, this derivation can be expressed as $S \Rightarrow_G^k x \Rightarrow_G w\ [r]$, for some $x \in V^+$ and $r \in P$. By the induction hypothesis, $\langle S, \emptyset\rangle \Rightarrow_H^* x'$, where $\tau(x') = x$ and x' is of the form

$$x' = x_0 \langle X_1, \{X_2, X_3, \ldots, X_n\}\rangle x_1 \langle X_2, \{X_3, \ldots, X_n\}\rangle x_2 \cdots \langle X_n, \emptyset\rangle x_n$$

where $X_i \in N$ for $i = 1, 2, \ldots, n$ and $x_j \in T^*$ for $j = 0, 1, \ldots, n$, for some $n \geq 1$. As $x \Rightarrow_G w\ [r]$, $x = x_0 X_1 x_1 X_2 x_2 \cdots X_n x_n$, $r = \lfloor X_1 \rightarrow y, U, W \rfloor$, $U \subseteq \{X_2, X_3, \ldots, X_n\}$, $W \cap \{X_2, X_3, \ldots, X_n\} = \emptyset$, and $w = x_0 y x_1 X_2 x_2 \cdots X_n x_n$. By the construction of H, there is $r' = \big(\langle X_1, \{X_2, X_3, \ldots, X_n\}\rangle \rightarrow y'\big) \in P'$ where $\tau(y') = y$. Then,

$$x' \Rightarrow_H x_0 y' x_1 \langle X_2, \{X_3, \ldots, X_n\}\rangle x_2 \cdots \langle X_n, \emptyset\rangle x_n\ [r']$$

Since $w' = x_0 y' x_1 \langle X_2, \{X_3, \ldots, X_n\}\rangle x_2 \cdots \langle X_n, \emptyset\rangle x_n$ is of the required form and $\tau(w') = w$, the induction step is completed. □

Claim 2. *If $\langle S, \emptyset\rangle \Rightarrow_H^k x$, where $x \in V'^*$ and $k \geq 0$, then $S \Rightarrow_G^* \tau(x)$ and x is of the form*

$$x = x_0\langle X_1, \{X_2, X_3, \ldots, X_n\}\rangle x_1 \langle X_2, \{X_3, \ldots, X_n\}\rangle x_2 \cdots \langle X_n, \emptyset\rangle x_n$$

where $X_i \in N$ for $i = 1, 2, \ldots, n$ and $x_j \in T^*$ for $j = 0, 1, \ldots, n$, for some $n \geq 0$.

Proof. This claim is established by induction on $k \geq 0$.

Basis. Let $k = 0$. Then, for $\langle S, \emptyset\rangle \Rightarrow_H^0 \langle S, \emptyset\rangle$, $S \Rightarrow_G^0 S$, so the basis holds.

Induction Hypothesis. Suppose that there exists $k \geq 0$ such that the claim holds for all derivations of length ℓ, where $0 \leq \ell \leq k$.

Induction Step. Consider any derivation of the form $\langle S, \emptyset\rangle \Rightarrow_H^{k+1} w$, where $w \in V'^*$. Since $k+1 \geq 1$, this derivation can be expressed as $\langle S, \emptyset\rangle \Rightarrow_H^k x \Rightarrow_H w\ [r']$, for some $x \in V^+$ and $r' \in P'$. By the induction hypothesis, $S \Rightarrow_G^* \tau(x)$ and x is of the form

$$x = x_0\langle X_1, \{X_2, X_3, \ldots, X_n\}\rangle x_1 \langle X_2, \{X_3, \ldots, X_n\}\rangle x_2 \cdots \langle X_n, \emptyset\rangle x_n$$

where $X_i \in N$ for $i = 1, 2, \ldots, n$ and $x_j \in T^*$ for $j = 0, 1, \ldots, n$, for some $n \geq 0$. As $x \Rightarrow_H w\ [r']$,

$$r' = (\langle X_1, \{X_2, X_3, \ldots, X_n\}\rangle \rightarrow y') \in P'$$

where $y' \in V'^*$, and there is $r = \lfloor X_1 \rightarrow y, U, W\rfloor \in P$, where $U \subseteq \{X_2, X_3, \ldots, X_n\}$, $W \cap \{X_2, X_3, \ldots, X_n\} = \emptyset$, and $\tau(y') = y$. Then,

$$x_0X_1x_1X_2x_2 \cdots X_nx_n \Rightarrow_G x_0yx_1X_2x_2 \cdots X_nx_n\ [r]$$

Since $x_0yx_1X_2x_2 \cdots X_nx_n$ is of the required form and it equals $\tau(w)$, induction step is completed. □

Consider Claim 1 with $x \in T^*$. Then, $S \Rightarrow_G^* x$ implies that $S \Rightarrow_H^* x$, so $L(G) \subseteq L(H)$. Consider Claim 2 with $x \in T^*$. Then, $\langle S, \emptyset\rangle \Rightarrow_H^* x$ implies that $S \Rightarrow_G^* x$, so $L(H) \subseteq L(G)$. Hence, $L(H) = L(G)$.

Finally, we argue that H is an LL CFG. For the sake of contradiction, suppose that H is not an LL CFG—that is, assume that there are $p' = (X \rightarrow y_1) \in P'$ and $r' = (X \rightarrow y_2) \in P'$ such that $y_1 \neq y_2$ and $\text{Predict}(p) \cap \text{Predict}(r) \neq \emptyset$. Let a be a symbol from $\text{Predict}(p') \cap \text{Predict}(r')$. By the construction of P', X is of the form $X = \langle A, Q\rangle$, for some $A \in N$ and $Q \subseteq N$, and there are $p = \lfloor A \rightarrow \tau(y_1), U_1, W_1\rfloor \in P$ and $r = \lfloor A \rightarrow \tau(y_2), U_2, W_2\rfloor \in P$ such that $U_1 \subseteq Q$, $U_2 \subseteq Q$, $W_1 \cap Q = \emptyset$, and $W_2 \cap Q = \emptyset$. Since $a \in \text{Predict}(p') \cap \text{Predict}(r')$,

$$\langle S, \emptyset\rangle \Rightarrow_H^* u\langle A, Q\rangle v \Rightarrow_H uy_1v\ [p'] \Rightarrow_H^* uaw_1, \text{and}$$

$$\langle S, \emptyset\rangle \Rightarrow_H^* u\langle A, Q\rangle v \Rightarrow_H uy_2v\ [r'] \Rightarrow_H^* uaw_2$$

for some $u \in T^*$, $v \in V'^*$ such that $\text{alph}\big(\tau(v)\big) = Q$ (see Claim 2), and $w_1, w_2 \in V'^*$. Then, by Claim 2,

$$S \Rightarrow_G^* uA\tau(v) \Rightarrow_G u\tau(y_1v)\ [p] \Rightarrow_G^* ua\tau(w_1), \text{and}$$

$$S \Rightarrow_G^* uA\tau(v) \Rightarrow_G u\tau(y_1v)\ [r] \Rightarrow_G^* ua\tau(w_2)$$

However, then $a \in \text{Predict}(p)$ and $a \in \text{Predict}(r)$, so $\text{Predict}(p) \cap \text{Predict}(r) \neq \emptyset$. Since both p and r have the same left-hand side and are applicable to $uA\tau(v)$, we have a contradiction with the fact that G is an LL RCG. Hence, H is an LL CFG, and the lemma holds. □

Let **LLCF** and **LLRC** denote the families of languages generated by LL CFGs and LL RCGs, respectively. The following theorem represents the main result of this paper.

Theorem 1. LLRC = LLCF

Proof. This theorem follows directly from Lemma 1 and the fact that every LL CFG can be trivialy converted to an equivalent LL RCG with empty permitting and forbidding context for each rule. □

4 CONCLUDING REMARKS

In this paper, we prooved that LL random context grammars generate the family of LL context-free languages. We propose two open problem areas as suggested topics of future investigations related to the topic of LL RCG.

I. Observe that for a single random context rule from P, the construction providede in Lemma 1 introduces several rules to P'. Given an LL CFG G, is there an algorithm which converts G into an equivalent LL RCG that contains fewer rules than G? In the case of an affirmative answer to this question, we might create a more efficient parser.

II. Is the LL property of LL RCGs decidable? Affirmitive answer is necessary in order to verify the input grammar of the parser.

REFERENCES

[1] A. V. Aho, M. S. Lam, R. Sethi, and J. D. Ullman. *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, Boston, 2nd edition, 2006.

[2] H. Bordihn and M. Holzer. Random context in regulated rewriting versus cooperating distributed grammar systems. In *LATA'08: Proceedings of the 2nd International Conference on Language and Automata Theory and Applications*, pages 125–136. Springer, 2008.

[3] A. B. Cremers, H. A. Maurer, and O. Mayer. A note on leftmost restricted random context grammars. *Information Processing Letters*, 2(2):31–33, 1973.

[4] J. Dassow and G. Păun. *Regulated Rewriting in Formal Language Theory*. Springer, New York, 1989.

[5] A. Meduna. *Automata and Languages: Theory and Applications*. Springer, London, 2000.

[6] A. Meduna. *Elements of Compiler Design*. Auerbach Publications, Boston, 2007.

[7] A. Meduna and M. Švec. *Grammars with Context Conditions and Their Applications*. Wiley, New Jersey, 2005.

[8] A. Meduna and P. Zemek. One-sided random context grammars. *Acta Informatica*, 48(3):149–163, 2011.

[9] G. Păun. A variant of random context grammars: semi-conditional grammars. *Theoretical Computer Science*, 41(1):1–17, 1985.

[10] G. Rozenberg and A. Salomaa, editors. *Handbook of Formal Languages, Vol. 2: Linear Modeling: Background and Application*. Springer, New York, 1997.

[11] A. Salomaa. *Formal Languages*. Academic Press, London, 1973.

[12] A. van der Walt and S. Ewert. A shrinking lemma for random forbidding context languages. *Theoretical Computer Science*, 237(1-2):149–158, 2000.

[13] A. van der Walt and S. Ewert. A pumping lemma for random permitting context languages. *Theoretical Computer Science*, 270(1-2):959–967, 2002.

[14] A. P. J. van der Walt. Random context grammars. In *Proceedings of Symposium on Formal Languages*, pages 163–165, 1970.

[15] G. Zetzsche. On erasing productions in random context grammars. In *ICALP'10: Proceedings of the 37th International Colloquium on Automata, Languages and Programming*, pages 175–186. Springer, 2010.