Alexander Okhotin

Notes on dual concatenation

Turku Centre for Computer Science

TUCS Technical Report
No 843, October 2007

Notes on dual concatenation

Alexander Okhotin
Department of Mathematics, University of Turku, *and*
Turku Centre for Computer Science
Turku FIN–20015, Finland, *and*
Academy of Finland
alexander.okhotin@utu.fi

TUCS Technical Report

No 843, October 2007

Abstract

The concatenation of formal languages has a logical dual (A. Okhotin, "The dual of concatenation", *Theoret. Comput. Sci.*, 320 (2005), 425–447), which is particularly important in the study of Boolean operations in formal language theory. In this paper, the closure or nonclosure of common language families under dual concatenation with finite, co-finite and regular languages is determined. In addition, language equations with union, linear concatenation and dual concatenation with co-finite constants are shown to be almost equal in power to linear conjunctive grammars.

Keywords: Concatenation; conjunctive grammars; language equations; trellis automata.

TUCS Laboratory
Discrete Mathematics for Information Technology

1 Introduction

The idea of writing words one after another is common to all languages. In mathematical models of languages it is represented by the operation of *concatenation.*

Many variants of this operation have been considered recently, inspired in part by processes occurring in genome evolution, which are modelled as operations on words representing DNA strands. In particular, a few such operations were introduced and studied in the context of formal language theory by Dassow et al. [2] and by Dassow, Mitrana and A. Salomaa [3]. An important unifying work on many variants of concatenation is the theory of *trajectory-based operations* proposed by Mateescu, Rozenberg and A. Salomaa [12] and further developed by Domaratzki [4, 5].

These operations resemble concatenation in that the symbols of one or two words are combined in a certain order to form another word, but the rules under which the symbols are combined are different from sequential composition. Once such a word operation is defined, it is extended to languages similarly to the standard concatenation, as a union of applications of this operation to individual words.

This paper concerns with another operation closely related to concatenation, which, on the contrary, maintains sequential composition of words, but defines a different operation on languages on its basis. Consider that the concatenation is

$$\begin{aligned} K \cdot L = \{w \mid &\text{there exists a factorization } w = uv, \\ &\qquad\qquad\text{such that } u \in K \text{ and } v \in L\}, \end{aligned}$$

where the condition of membership uses an existential quantifier and conjunction. Replacing them with a universal quantifier and disjunction, the following logical dual of concatenation is obtained [17]:

$$\begin{aligned} K \odot L = \{w \mid &\text{for every factorization } w = uv \\ &\qquad\text{it holds that } u \in K \text{ or } v \in L\} = \overline{\overline{K} \cdot \overline{L}}. \end{aligned}$$

Note that there is no such thing as dual concatenation of two words: the underlying operation on words is the standard concatenation, and it is the way the words in two languages determine the words in their dual concatenation which has been dualized.

This paper continues the study of this operation with the question of the closure of common language families under *dual concatenation with co-finite, finite and regular languages*, investigated in Sections 3–5. Linear context-free and context-free languages are not closed under any of them. Deterministic context-free languages are closed under dual concatenation with regular languages from the right, while dually concatenating finite or co-finite languages from the left leads out of this class. Finite intersections of context-free languages are closed under dual concatenation with co-finite languages, but not with finite languages. Finally, conjunctive languages [13] are closed under dual concatenation with arbitrary regular languages.

The second result of this paper is about *language equations.* Language equations have been occasionally studied since the 1960s, and in the last years they became an object of a systematic research. In particular, language equations with different variants of concatenation were studied by Kari [8] and by Domaratzki and K. Salomaa [6]. The research on equations with all Boolean operations led to unexpected characterizations of computability [14], as well as to the notion of a Boolean grammar [16], which completes the formalism of context-free grammars by allowing the use of all Boolean operations. The power of dual concatenation in language equations is a natural topic [17], and the findings of this paper are used to contribute a new result: in Section 6 it is shown that equations using union, concatenation with singletons and dual concatenations with co-finite constants can specify linear conjunctive languages [13, 18] almost precisely.

2 Language families

Let us define the families of languages used in this paper. We will consider, among others, regular languages, linear context-free languages, context-free languages, linear conjunctive languages and conjunctive languages. The latter are generated by *conjunctive grammars* [13], which are an extension of context-free grammars with an explicit conjunction operation; the rest of the families are defined by restricted subclasses of conjunctive grammars, so let us give the most general definition.

Definition 1. *A conjunctive grammar [13] is a quadruple $G = (\Sigma, N, P, S)$, in which Σ and N are disjoint finite nonempty sets of terminal and nonterminal symbols, respectively; P is a finite set of grammar rules, each of the form*

$$A \to \alpha_1 \& \ldots \& \alpha_n \quad (\textit{where } A \in N,\ n \geqslant 1 \textit{ and } \alpha_1, \ldots, \alpha_n \in (\Sigma \cup N)^*) \qquad (1)$$

$S \in N$ is a nonterminal designated as the start symbol.

Informally, a rule (1) states that if a word is generated by each α_i, then it is generated by A. This semantics can be formalized using term rewriting, which generalizes Chomsky's word rewriting.

Definition 2. *Given a grammar $G = (\Sigma, N, P, S)$, consider terms built from symbols in $\Sigma \cup N$ using conjunction and concatenation. The relation $\Longrightarrow$ of immediate derivability on the set of such terms*

1. *Using a rule $A \to \alpha_1 \& \ldots \& \alpha_n$, a subterm A of any term $\varphi(A)$ can be rewritten as follows: $\varphi(A) \Longrightarrow \varphi(\alpha_1 \& \ldots \& \alpha_n)$.*

2. *A conjunction of several identical terminal words can be rewritten by one such word: $\varphi(w \& \ldots \& w) \Longrightarrow \varphi(w)$, for every $w \in \Sigma^*$.*

The language generated by a term $\mathcal{A}$ is the set of all words over Σ derivable from it in a finite number of steps: $L_G(\alpha) = \{w \mid w \in \Sigma^*,\ \mathcal{A} \Longrightarrow^* w\}$. The language generated by the grammar is $L(G) = L_G(S) = \{w \mid w \in \Sigma^*,\ S \Longrightarrow^* w\}$.

Let us give a example of a conjunctive grammar for a language that is known to be inexpressible as a finite intersection of context-free languages, see Wotschke [19].

Example 1 ([13, 18])**.** *The following conjunctive grammar generates the language $\{wcw \mid w \in \{a,b\}^*\}$:*

$$\begin{array}{lcl}
S & \to & C\&D \\
C & \to & aCa \mid aCb \mid bCa \mid bCb \mid c \\
D & \to & aA\&aD \mid bB\&bD \mid cE \\
A & \to & aAa \mid aAb \mid bAa \mid bAb \mid cEa \\
B & \to & aBa \mid aBb \mid bBa \mid bBb \mid cEb \\
E & \to & aE \mid bE \mid \varepsilon
\end{array}$$

The key part of this grammar is the nonterminal D, which generates $\{uczu \mid u, z \in \{a,b\}^*\}$. Its rules match a symbol in the left part to the corresponding symbol in the right part using A or B; the recursive reference of D to aD or bD makes the remaining symbols be compared in the same way.

It remains to define subclasses of conjunctive grammars we shall consider. A conjunctive grammar is called *context-free*, if every every rule (1) contains a single conjunct, that is, $n = 1$. A conjunctive grammar is *linear conjunctive*, if each α_i in every rule (1) is in $\Sigma^* \cup \Sigma^* N \Sigma^*$. A conjunctive grammar is *linear context-free*, if it is at the same time linear conjunctive and context-free. We shall also consider finite intersections of context-free and of linear context-free languages, which were studied by Liu and Weiner [10], by Wotschke [19] and by Kutrib, Malcher and Wotschke [9]; these families are properly contained in conjunctive and linear conjunctive languages, respectively. Finally, regular languages are those recognized by finite automata, while deterministic context-free languages are recognized by deterministic pushdown automata, see Harrison [7].

The largest family to be mentioned is the family defined by Boolean grammars [16]. Boolean grammars further extend conjunctive grammars by allowing negation in their rules, which are of the form $A \to \alpha_1 \& \ldots \& \alpha_m \& \neg\beta_1 \& \ldots \& \neg\beta_n$. Their semantics is defined using language equations by interpreting logical connectives as Boolean operations on languages [16]. This family can express dual concatenation directly, as a composition of concatenation and complementation [17]. For more information about this family the reader is referred to a recent survey [18].

3 Dual concatenation with co-finite languages

Dual concatenation with co-finite languages is dual to concatenation with finite languages. It is known [17] that the language Σ^* is a zero for dual concatenation, that is, $L \odot \Sigma^* = \Sigma^* \odot L = \Sigma^*$ for every $L \subseteq \Sigma^*$. The language Σ^+ is an identity: $L \odot \Sigma^+ = \Sigma^+ \odot L = L$. It is also known that $L \odot \overline{a} = La \cup \overline{\Sigma^* a}$; in other words, dual concatenation with a co-singleton is the same as concatenation with a singleton plus some regular garbage.

Let us consider dual concatenation with co-finite languages in general, starting from the language $\overline{\{u, v\}}$, with $u, v \in \Sigma^*$. If one of these two words is a prefix of the other, the dual concatenation with such a co-finite language represents intersection:

Lemma 1. *Let Σ be an alphabet, let $x \in \Sigma^+$ be a nonempty word. Then, for every $L \subseteq \Sigma^*$,*

$$L \odot \overline{\{\varepsilon, x\}} = L \cap (Lx \cup \overline{\Sigma^* x}).$$

Proof. $L \odot \overline{\{\varepsilon, x\}} = \overline{\overline{L} \cdot \{\varepsilon, x\}} = \overline{\overline{L} \cup \overline{L}x} = L \cap \overline{\overline{L}x} = L \cap (L \odot \overline{x}) = L \cap (Lx \cup \overline{\Sigma^* x})$. □

This equality can be regarded as a formal dual of $L \cdot \{\varepsilon, x\} = L \cup Lx$. It can be used to define intersection of any two languages known not to contain words ending with a particular symbol as follows:

Lemma 2. *Let Σ be an alphabet, let $\dagger \notin \Sigma$, let $K, L \subseteq (\Sigma \cup \{\dagger\})^* \setminus (\Sigma \cup \{\dagger\})^* \dagger$. Then*

$$(L \cup K\dagger) \odot \overline{\{\varepsilon, \dagger\}} = (L \cap K)\dagger \cup L \quad \textit{and therefore}$$
$$\big((L \cup K\dagger) \odot \overline{\{\varepsilon, \dagger\}}\big) \cdot \{\dagger\}^{-1} = L \cap K.$$

Proof. By Lemma 1, $(L \cup K\dagger) \odot \overline{\{\varepsilon, \dagger\}} = (L \cup K\dagger) \cap \big(L\dagger \cup K\dagger\dagger \cup \overline{(\Sigma \cup \{\dagger\})^*\dagger}\big) = \cancel{(L \cap L\dagger)} \cup \cancel{(L \cap K\dagger\dagger)} \cup (L \cap \overline{(\Sigma \cup \{\dagger\})^*\dagger}) \cup (K\dagger \cap L\dagger) \cup \cancel{(K\dagger \cap K\dagger\dagger)} \cup \cancel{(K\dagger \cap \overline{(\Sigma \cup \{\dagger\})^*\dagger})} = L \cup (K \cap L)\dagger$, where four intersections are empty because no words ending with † are in K and in L. The second statement follows. □

This construction leads to the following theorem in the style of the theory of abstract families of languages, see Mateescu and A. Salomaa [11].

Theorem 1. *Every family of languages closed under (1) union, (2) concatenation with singletons, (3) quotient with singletons and (4) dual concatenation with co-finite languages is closed under intersection.*

This result allows us to identify some language families not closed under dual concatenation with co-finite languages. These are the linear context-free and the context-free languages: since both are closed under union, concatenation with singletons and quotient with singletons, the hypothetical closure under dual concatenation with co-finite languages would imply that they are closed under intersection, which is known to be untrue.

It was shown how dual concatenation with a co-finite language can express intersection. Let us now see how intersection can in turn be used to represent dual concatenation with a co-finite constant. First factorize dual concatenation as follows:

Lemma 3. *Let $\Sigma = \{a_1, \ldots, a_m\}$ be an alphabet, let $K, L \subseteq \Sigma^*$ with $\varepsilon \notin K$, and consider the decomposition $K = K_1a_1 \cup \ldots \cup K_ma_m$. Then*

$$L \odot \overline{K} = L \odot \overline{K_1a_1 \cup \ldots \cup K_ma_m} = (L \odot \overline{K_1})a_1 \cup \ldots \cup (L \odot \overline{K_m})a_m \cup \{\varepsilon\}.$$

Proof. Transform the left-hand side as follows: $L \odot \overline{\bigcup_{i=1}^m K_ia_i} = \overline{\overline{L} \cdot \bigcup_{i=1}^m K_ia_i} = \overline{\bigcup_{i=1}^m \overline{L}K_ia_i}$. Pushing the complementation downwards, we obtain $\bigcap_{i=1}^m L \odot \overline{K_i} \odot \overline{a_i}$, which is equal to $\bigcap_{i=1}^m \big((L \odot \overline{K_i})a_i \cup \overline{\Sigma^*a_i}\big)$. Using the distributivity of union and intersection, this can be represented as a union of 2^m intersections of m terms each. Any intersection that includes $(L \odot \overline{K_i})a_i$ and $(L \odot \overline{K_j})a_j$ for $i \neq j$ is bound to be empty, and therefore there are only $m + 1$ potentially nonempty intersections: $\overline{\Sigma^*a_1} \cap \ldots \cap \overline{\Sigma^*a_{i-1}} \cap (L \odot \overline{K_i})a_i \cap \overline{\Sigma^*a_{i+1}} \cap \ldots \cap \overline{\Sigma^*a_m} = (L \odot \overline{K_i})a_i$ for $i = 1, \ldots, m$ and $\overline{\Sigma^*a_1} \cap \ldots \cap \overline{\Sigma^*a_m} = \{\varepsilon\}$. Their union is exactly the expression in the right-hand side of the proposed equality. □

Now dual concatenation with every co-finite language can be decomposed as follows:

Lemma 4. *For every finite language $K \subset \Sigma^*$ there exists and can be effectively constructed an expression $\varphi(X)$ using union, intersection, concatenation with singletons and constants $\{\varepsilon\}$, Σ^*, such that $\varphi(L) = L \odot \overline{K}$ for every language $L \subseteq \Sigma^*$.*

Proof. Induction on the least nonnegative integer, for which no words of this or greater length are in K.

Basis, $K = \varnothing$**.** Then $L \odot \overline{K} = \Sigma^*$ and $\varphi(X)$ can be defined as Σ^*.

Induction step, the case $\varepsilon \notin K$. Let the longest word in K be of length ℓ. Represent K as $K_1a_1 \cup \ldots \cup K_ma_m$, where $\Sigma = \{a_1, \ldots, a_m\}$. Then, by Lemma 3, $L \odot \overline{K} = \bigcup_{i=1}^m (L \odot \overline{K_i})a_i \cup \{\varepsilon\}$. Since the longest word in each K_i is of length at most $\ell - 1$, by the induction hypothesis, there is an expression $\varphi_i(X)$, such that $\varphi_i(L) = L \odot \overline{K_i}$. It is then sufficient to define $\varphi(X) = \varphi_1(X)a_1 \cup \ldots \cup \varphi_m(X)a_m \cup \{\varepsilon\}$.

Induction step, the case $\varepsilon \in K$. Then $L \odot \overline{K} = L \odot \overline{K \setminus \{\varepsilon\}} \cap L$, and the previous case is applicable. The resulting expression is $\varphi(X) = \big(\varphi_1(X)a_1 \cup \ldots \cup \varphi_m(X)a_m \cup \{\varepsilon\}\big) \cap X$. □

Lemma 4 immediately implies the following theorem:

Theorem 2. *Every family of languages containing $\{\varepsilon\}$ and Σ^* and closed under (1) union, (2) intersection and (3) concatenation with singletons is closed under dual concatenation with co-finite languages.*

Consider finite intersections of context-free [10, 19] or linear context-free languages [9]: both families have the closure properties required by Theorem 2, and they are therefore closed under dual concatenation with co-finite languages.

4 Dual concatenation with finite languages

The basic case of dual concatenation with a finite language is *dual concatenation with the empty set.* This operation has first been considered by Birget [1], who attributed it to J.-E. Pin, defined it as

$$\overline{\Sigma^* \overline{L}} = \{w \mid \text{every suffix of } w \text{ is in } L\}$$

and studied its descriptional complexity. In our terminology, this operation is $\varnothing \odot L$. As one can easily see, it is dual to concatenation with Σ^*, which can be represented as follows:

$$\Sigma^* L = \{w \mid \text{some suffix of } w \text{ is in } L\}.$$

The operation $L \odot \varnothing$ can be considered as well: it similarly defines the language of all words w, such that every *prefix* of w is in L. While these operations obviously preserve regularity, applying them to simple nonregular languages yields nontrivial results:

Example 2. *Consider linear context-free languages $L = \{aucxav, bucxbv \mid u, v, x \in \{a,b\}^*,\ |u| = |v|\} \cup c\{a,b\}^*$ and $K = \{ucv \mid u, v \in \{a,b\}^*,\ |u| = |v|\}$. Then $L \odot \varnothing = \{ucxu \mid u, x \in \{a,b\}^*\}$ and $(L \odot \varnothing) \cap K = \{wcw \mid w \in \{a,b\}^*\}$.*

Recall the conjunctive grammar from Example 1: there $L = L_G(aA \cup bB)$, $K = L_G(C)$ and the rules for nonterminal D implement dual concatenation with the empty set. This construction holds in general, providing the closure of the conjunctive languages under dual concatenation with the empty set. For any conjunctive grammar $G = (\Sigma, N, P, S)$, such that $\varepsilon \in L(G)$, the grammar $G' = (\Sigma, N \cup \{S'\}, P \cup \{S' \to aS'\&S \mid a \in \Sigma\} \cup \{S' \to \varepsilon\}, S')$, generates the language $L(G) \odot \varnothing$; a conjunctive grammar for $\varnothing \odot L(G)$ is constructed symmetrically. A more general closure property will be formally proved in the next section.

Example 2 is a good source of nonclosure results. It immediately follows that neither the linear context-free nor the context-free languages are closed under dual concatenation with the empty set. In addition, finite intersections of linear context-free and of context-free languages are also not closed under this operation, because the language $\{wcw \mid w \in \{a,b\}^*\}$, as demonstrated by Wotschke [19], is not representable by such a intersection.

5 Dual concatenation with regular languages

We have already seen that context-free and linear context-free languages, as well as their finite intersections, are not closed under dual concatenation with regular languages, $\odot Reg$. On the other hand, linear conjunctive languages are closed under $\odot Reg$, because they are closed under complementation and under concatenation with regular languages. Let us show that conjunctive languages, which are not known to be closed under dual concatenation or under complementation, are nevertheless closed under $\odot Reg$.

Theorem 3. *For every conjunctive language L and regular language R, the languages $L \odot R$ and $R \odot L$ are conjunctive. Given a conjunctive grammar for L and a finite automaton for R, conjunctive grammars for $L \odot R$ and $R \odot L$ can be effectively constructed.*

Proof. Given a conjunctive grammar $G = (\Sigma, N, P, S)$ and a finite automaton $A = (\Sigma, Q, q_0, \delta, F)$, construct a conjunctive grammar $\widehat{G} = (\Sigma, N \cup \{T_q \mid q \in Q\}, P \cup P', T_{q_0})$, where P' consists of the following rules:

$$T_q \to aT_{\delta(q,a)} \quad \text{(for all } q \in F \text{ and } a \in \Sigma) \tag{2a}$$

$$T_q \to aT_{\delta(q,a)}\&S \quad \text{(for all } q \notin F \text{ and } a \in \Sigma) \tag{2b}$$

$$T_q \to \varepsilon \quad \text{(for all } q \in F) \tag{2c}$$

$$T_q \to \varepsilon\&S \quad \text{(for all } q \notin F) \tag{2d}$$

It is sufficient to establish the following claim: *For every word $w \in \Sigma^*$ and for every state $q \in Q$, $w \in L_{\widehat{G}}(T_q)$ if and only if for every factorization $w = uv$, $\delta(q,u) \in F$ or $v \in L(G)$.* The proof is an induction on the length of w.

Basis $w = \varepsilon$. Then there exists only one factorization of $w = uv$, the one with $u = v = \varepsilon$. If $q \in F$, then $\varepsilon \in L_{\widehat{G}}(T_q)$ and $\delta(q,u) = q \in F$. If $q \notin F$, then the condition $\delta(q,u) \in F$ is false, and both conditions $w \in L_{\widehat{G}}(T_q)$ and $v \in L(G)$ are equivalent to $\varepsilon \in L(G)$.

Induction step: let $w = aw'$ and denote $q' = \delta(q,a)$.

	Reg	LinCF	co-LinCF	LinConj	DetCF	CF	co-CF	$\cap$CF	Conj	co-Conj	Bool
$\sim L$	$+$	$-$	$-$	$+$	$+$	$-$	$-$	$-$	?	?	$+$
$L_1 \cdot L_2$	$+$	$-$	$-$	$-$	$-$	$+$	$-$	?	$+$	?	$+$
$L_1 \odot L_2$	$+$	$-$	$-$	$-$	$-$	$-$	$+$	$-$	?	$+$	$+$
$\cdot Fin/Fin\cdot$	$+$	$+$	$-$	$+$	$+/-$	$+$	$-$	$+$	$+$	$+$	$+$
$\odot coFin/coFin\odot$	$+$	$-$	$+$	$+$	$+/-$	$-$	$+$	$+$	$+$	$+$	$+$
$\cdot\Sigma^*/\Sigma^*\cdot$	$+$	$+$	$-$	$+$	$+/-$	$+$	$-$	?	$+$	$+$	$+$
$\odot\varnothing/\varnothing\odot$	$+$	$-$	$+$	$+$	$+/-$	$-$	$+$	$-$	$+$	$+$	$+$
$\cdot Reg/Reg\cdot$	$+$	$+$	$-$	$+$	$+/-$	$+$	$-$	?	$+$	$+$	$+$
$\odot Reg/Reg\odot$	$+$	$-$	$+$	$+$	$+/-$	$-$	$+$	$-$	$+$	$+$	$+$

Table 1: Closure and nonclosure under the operations studied and their duals.

$\bigcirc\!\!\!\!\!\Rightarrow$ Suppose $aw' \in L_{\widehat{G}}(T_q)$. Then $w' \in L_{\widehat{G}}(T_{q'})$ and, if $q \notin F$, then also $aw' \in L(G)$. The former, by the induction hypothesis, implies that for every factorization $w' = u'v$, $\delta(q', u') \in F$ or $v \in L(G)$.

Consider any factorization $aw' = uv$; it has to be proved that $\delta(q, u) \in F$ or $v \in L(G)$. If $u = \varepsilon$, then $v = aw'$ and we know that $\delta(q, \varepsilon) = q \in F$ or $aw' \in L(G)$. If $u = au'$, then $\delta(q, u) = \delta(q', u')$ and, as inferred from the induction hypothesis above, $\delta(q', u') \in F$ or $v \in L(G)$, which completes the proof in this direction.

$\bigcirc\!\!\!\!\!\Leftarrow$ Conversely, assume that for every factorization $aw' = uv$ it holds that $\delta(q, u) \in F$ or $v \in L(G)$. It follows that for every factorization $w' = u'v$, $\delta(q', u') \in F$ or $v \in L(G)$: it is sufficient to consider the factorization $aw' = au' \cdot v$ and to note that $\delta(q', u') = \delta(q, au')$. Then, by the induction hypothesis, $w' \in L_{\widehat{G}}(T_{q'})$.

Consider the factorization $aw' = \varepsilon \cdot aw'$, for which we know that $\delta(q, \varepsilon) = q \in F$ or $aw' \in L(G)$. In the former case, $\widehat{G}$ contains the rule (2a), and, by this rule, $aw' \in L_{\widehat{G}}(T_{q'})$. In the latter case, $aw' \in L_{\widehat{G}}(S)$ and the rule (2b) can produce aw'.

Substituting $q = q_0$ into this statement, one obtains $L_{\widehat{G}}(T_{q_0}) = \{u \mid \delta(q_0, u) \in F\} \odot L(G)$, that is, $L(\widehat{G}) = L(A) \odot L(G)$. □

Let us put together the closure properties studied in this paper. The first three rows of Table 1 are already known [7, 17]. The next three pairs of rows refer to the operations investigated in this paper and their formal duals.

The case of deterministic context-free languages needs comments. Since they are closed under complementation and right-concatenation with regular languages [7], they are therefore closed under right-dual-concatenation with regular languages. On the other hand, they are not closed under concatenation of a two-element set from the left [7], and under left-concatenation of Σ^* either:

Proposition 1. *For the deterministic context-free language* $L = \{ca^nb^n \mid n \geqslant 0\} \cup \{cca^nb^{2n} \mid n \geqslant 0\}$ *over the alphabet* $\Sigma = \{a, b, c\}$*, the concatenation* Σ^*L *is not deterministic context-free.*

It is sufficient to note that

$$\{a, b, c\}^*L \cap cc\{a, b\}^* = cc\big(\{a^nb^n \mid n \geqslant 0\} \cup \{a^nb^{2n} \mid n \geqslant 0\}\big),$$

where the latter is not a deterministic context-free language [7, Theorem 11.8.4]. This implies the nonclosure of deterministic context-free languages under dual concatenation with co-finite languages and with $\varnothing$ on the left.

The problem of whether the dual concatenation of any two conjunctive languages is always conjunctive [17] remains open, and is proposed for future research.

6 Language equations with $\cup$, lin$\cdot$ and lin$\odot$

Using the results of Section 3, it will be demonstrated that two particular families of language equations can simulate each other. These are systems of the following form:

$$\begin{cases} X_1 & = & \varphi_1(X_1, \ldots, X_n) \\ & \vdots & \\ X_n & = & \varphi_n(X_1, \ldots, X_n) \end{cases} \tag{3}$$

In the first family, each φ_i may contain union, intersection, concatenation with singletons and $\{\varepsilon\}$. Least solutions of such systems represent linear conjunctive languages [13, 18].

The other kind of systems to be considered allows expressions φ_i to contain union, concatenation with singletons and $\{\varepsilon\}$. Least solutions of such systems define a certain new family of languages. Recalling Lemma 4 above, dual concatenation with co-finite languages can be represented using intersection, and hence these systems can be simulated by systems from the former class, showing the containment of the new language family in the linear conjunctive languages. A reverse simulation will now be demonstrated.

Theorem 4. *Let* Σ *be an alphabet, let* $\dagger \notin \Sigma$ *and let the homomorphism* $h : \Sigma^* \to (\Sigma \cup \{\dagger\})^*$ *be defined by* $h(a) = a\dagger$ *for every* $a \in \Sigma$*. Then for every linear conjunctive grammar* G *there exists and can be effectively constructed a system of language equations of the form* $X_i = \varphi_i(X_1, \ldots, X_n)$ $(1 \leqslant i \leqslant n)$*, where* φ_i *contains (1) union, (2) concatenation with singleton constants and (3) dual concatenation with co-finite constants, such that the first component of its unique solution equals* $h(L(G))$ *modulo intersection with* $(\Sigma\dagger)^* = h(\Sigma^*)$*.*

Proof. Let $G = (\Sigma, N, P, S)$ be a linear conjunctive grammar in the shortened linear normal form [15], that is, all rules in P are of the form $A \to bB \& Cc$ or $A \to a$, where $A, B, C \in N$ and $a, b, c \in \Sigma$. Let $\overrightarrow{N} = \{\overrightarrow{A} \mid A \in N\}$ and $\overleftarrow{N} = \{\overleftarrow{A} \mid A \in N\}$, and construct the following system of language equations in variables $\overrightarrow{N} \cup \overleftarrow{N}$:

$$\overrightarrow{A} = \bigcup_{A \to bB\&Cc \in P} (b\overleftarrow{B} \cup \overrightarrow{C}c\dagger) \odot \overline{\{\varepsilon, \dagger\}} \ \cup \bigcup_{A \to a \in P} a\dagger \tag{4a}$$

$$\overleftarrow{A} = \bigcup_{A \to bB\&Cc \in P} \overline{\{\varepsilon, \dagger\}} \odot (b\overleftarrow{B} \cup \dagger\overrightarrow{C}c) \ \cup \bigcup_{A \to a \in P} \dagger a \tag{4b}$$

This system belongs to a class of *strict systems* which are guaranteed to have a unique solution. Let $(\overrightarrow{A} = \overrightarrow{L}_A, \overleftarrow{A} = \overleftarrow{L}_A)_{A \in N}$ be this solution.

The first claim is that $\overrightarrow{L}_A$ contains no words starting with †. Suppose the contrary, and let $x \in (\Sigma \cup \{\dagger\})^*$ be the shortest word, such that $\dagger x \in \overrightarrow{L}_A$. Then, by (4a), $\dagger x$ is in $(b\overleftarrow{L}_B \cup \overrightarrow{L}_C c\dagger) \odot \overline{\{\varepsilon, \dagger\}}$ for some b, c, B and C. Since $\varepsilon \notin \overline{\{\varepsilon, \dagger\}}$, then, $\dagger x \in b\overleftarrow{L}_B \cup \overrightarrow{L}_C c\dagger$, which implies $\dagger x \in \overrightarrow{L}_C c\dagger$, and thus $\overrightarrow{L}_C$ contains a shorter word starting with † than $\dagger x$. The contradiction obtained establishes the claim. Symmetrically, no words ending with † are in $\overleftarrow{L}_A$.

Define another homomorphism $h' : \Sigma^* \to (\Sigma \cup \{\dagger\})^*$ as $h'(a) = \dagger a$ for each $a \in \Sigma$. It is now claimed that $\overrightarrow{L}_A \cap (\Sigma\dagger)^* = h(L_G(A))$ and $\overleftarrow{L}_A \cap (\dagger\Sigma)^* = h'(L_G(A))$, and from this the statement of the theorem will follow. First, assume $w \in L_G(A)$ and let us show by induction on $|w|$ that $h(w) \in \overrightarrow{L}_A$ and $h'(w) \in \overleftarrow{L}_A$.

Basis $w = a$**.** If $a \in L_G(A)$, then $A \to a \in P$ and the equation (4a) explicitly contains $a\dagger = h(a) \in \overrightarrow{L}_A$. Similarly, (4b) gives $\dagger a = h'(a) \in \overleftarrow{L}_A$.

Induction step. Let $w = buc$, where $b, c \in \Sigma$ and $u \in \Sigma^*$. If $buc \in L_G(A)$, then there is a rule $A \to bB\&Cc \in P$, such that $uc \in L_G(B)$ and $bu \in L_G(C)$. By the induction hypothesis, $h'(uc) \in \overleftarrow{L}_B$ and $h(bu) \in \overrightarrow{L}_C$. Since $h(buc) = bh'(uc)\dagger = h(bu)c\dagger$, we obtain

$$\begin{aligned} h(buc) \in (b\overleftarrow{L}_B \cap \overrightarrow{L}_C c)\dagger \subseteq (b\overleftarrow{L}_B \cap \overrightarrow{L}_C c)\dagger \cup b\overleftarrow{L}_B = \\ = (b\overleftarrow{L}_B \cup \overrightarrow{L}_C c\dagger) \odot \overline{\{\varepsilon, \dagger\}} \subseteq \overrightarrow{L}_A, \end{aligned}$$

where the equality is by Lemma 2, used for $L = b\overleftarrow{L}_B$ and $K = \overrightarrow{L}_C c$ and applicable because $\overleftarrow{L}_B \cap (\Sigma \cup \{\dagger\})^*\dagger = \varnothing$.

The case $h'(buc) \in \overrightarrow{L}_A$ is proved symmetrically.

It remains to prove the converse, that is, if $x \in (\Sigma\dagger)^* \cup (\dagger\Sigma)^*$ is in $\overrightarrow{L}_A$ (or in $\overrightarrow{L}_A$), then $x = h(w)$ ($x = h'(w)$, respectively) for some $w \in L_G(A)$. Induction on $|x|$.

Basis $|x| = 2$**.** The shortest words in $\overrightarrow{L}_A$ are of the form $x = a\dagger$, where $A \to a \in P$. Then $x = h(a)$ and $a \in L_G(A)$. Similarly, if $x = \dagger a \in \overleftarrow{L}_A$, then $x = h'(a)$ and $a \in L_G(A)$.

Induction step. Let $x \in (\Sigma\dagger)^* \cup (\dagger\Sigma)^*$, with $|x| \geqslant 4$, be in $\overleftarrow{L}_A$. By the equation (4a), there exists a rule $A \to bB\&Cc \in P$, such that $x \in (b\overleftarrow{L}_B \cup \overrightarrow{L}_C c\dagger) \odot \overline{\{\varepsilon, \dagger\}}$. Then, according to Lemma 2, $x \in (b\overleftarrow{L}_B \cap \overrightarrow{L}_C c)\dagger \cup b\overleftarrow{L}_B$. The case $x \in b\overleftarrow{L}_B$ is impossible, because then $\overleftarrow{L}_B$ would contain a word ending with †.

Assume $x \in (b\overleftarrow{L}_B \cap \overrightarrow{L}_C c)\dagger$; then $x = b\dagger yc\dagger$, where $y \in (\Sigma\dagger)^*$, $\dagger yc \in \overleftarrow{L}_B$ and $b\dagger y \in \overrightarrow{L}_C$. Let $u \in \Sigma^*$ be a word, such that $h(u) = y$. Since we know that $h'(uc) = \dagger h(u)c \in \overleftarrow{L}_B$ and $h(bu) = b\dagger u \in \overrightarrow{L}_C$, by the induction hypothesis, $uc \in L_G(B)$ and $bu \in L_G(C)$. Then, by the rule $A \to bB\&Cc$, $w = buc \in L_G(A)$.

Finally, note that $\overrightarrow{L}_S \cap h(\Sigma^*) = h(L(G))$, which completes the proof. □

This last result leaves two questions to ponder. First, can these equations specify every linear conjunctive language precisely, without a homomorphic encoding and an intersection with a regular language? If so, these two language families will coincide. Second, do any similar results hold for equations with union, concatenation and dual concatenation? Are they powerful enough to simulate Boolean grammars?

Acknowledgement

Research supported by the Academy of Finland under grant 118540.

References

[1] J.-C. Birget, "The state complexity of $\overline{\Sigma^*\overline{L}}$ and its connection with temporal logic", *Information Processing Letters*, 58:4 (1996), 185–188.

[2] J. Dassow, C. Martín-Vide, G. Păun, A. Rodriguez-Paton, "Conditional concatenation", *Fundamenta Informaticae*, 44:4 (2000), 353–372.

[3] J. Dassow, V. Mitrana, A. Salomaa, "Operations and language generating devices suggested by the genome evolution", *Theoretical Computer Science*, 270:1–2 (2002), 701–738.

[4] M. Domaratzki, *Trajectory-Based Operations*, Ph.D Thesis, Queen's University, Kingston, Ontario, Canada, 2004.

[5] M. Domaratzki, "More words on trajectories", *Bulletin of the EATCS*, 86 (2005), 107–145.

[6] M. Domaratzki, K. Salomaa, "Decidability of trajectory-based equations", *Theoretical Computer Science*, 345:2–3 (2005), 304–330.

[7] M. Harrison, *Introduction to Formal Language Theory*, Addison-Wesley, 1978.

[8] L. Kari, "On language equations with invertible operations", *Theoretical Computer Science*, 132 (1994), 129–150.

[9] M. Kutrib, A. Malcher, D. Wotschke, "The Boolean closure of linear context-free languages", *Developments in Language Theory* (DLT 2004, Auckland, New Zealand, December 13-17, 2004), LNCS 3340, Springer, 2004, 284–295.

[10] L. Y. Liu, P. Weiner, "An infinite hierarchy of intersections of context-free languages", *Mathematical Systems Theory*, 7 (1973), 187–192.

[11] A. Mateescu, A. Salomaa, "Aspects of classical language theory", in: Rozenberg, Salomaa (Eds.), *Handbook of Formal Languages*, Vol. 1, Springer, 1997, 175–251.

[12] A. Mateescu, G. Rozenberg, A. Salomaa, "Shuffle on trajectories: Syntactic constraints", *Theoretical Computer Science*, 197:1–2 (1998), 1–56.

[13] A. Okhotin, "Conjunctive grammars", *Journal of Automata, Languages and Combinatorics*, 6:4 (2001), 519–535.

[14] A. Okhotin, "Decision problems for language equations with Boolean operations", *Automata, Languages and Programming* (ICALP 2003, Eindhoven, The Netherlands, June 30–July 4, 2003), LNCS 2719, 239–251.

[15] A. Okhotin, "On the equivalence of linear conjunctive grammars to trellis automata", *RAIRO Informatique Théorique et Applications*, 38:1 (2004), 69–88.

[16] A. Okhotin, "Boolean grammars", *Information and Computation*, 194:1 (2004), 19–48.

[17] A. Okhotin, "The dual of concatenation", *Theoretical Computer Science*, 345:2–3 (2005), 425–447.

[18] A. Okhotin, "Nine open problems on conjunctive and Boolean grammars". *Bulletin of the EATCS*, 91 (2007), 96–119.

[19] D. Wotschke, "Nondeterminism and Boolean operations in PDAs", *Journal of Computer and System Sciences*, 16:3 (1978), 456–461.

Turku Centre *for* Computer Science

Lemminkäisenkatu 14 A, 20520 Turku, Finland | www.tucs.fi

University of Turku
- Department of Information Technology
- Department of Mathematical Sciences

Åbo Akademi University
- Department of Computer Science
- Institute for Advanced Management Systems Research

Turku School of Economics and Business Administration
- Institute of Information Systems Sciences

ISBN 978–952–12–1962–7
ISSN 1239-1891