International Journal of Foundations of Computer Science
Vol. 19, No. 3 (2008) 597–615

World Scientific
www.worldscientific.com

CONJUNCTIVE GRAMMARS GENERATE NON-REGULAR UNARY LANGUAGES

ARTUR JEŻ

Institute of Computer Science, University of Wroclaw, ul. Joliot-Curie 15
Wroclaw, 53-407, Poland
aje@ii.uni.wroc.pl

Received 30 September 2007
Accepted 14 February 2008
Communicated by Tero Harju and Juhani Karhumäki

Conjunctive grammars, introduced by Okhotin, extend context-free grammars by an additional operation of intersection in the body of any production of the grammar. Several theorems and algorithms for context-free grammars generalize to the conjunctive case. Okhotin posed nine open problems concerning those grammars. One of them was a question, whether a conjunctive grammars over a unary alphabet generate only regular languages. We give a negative answer, contrary to the conjectured positive one, by constructing a conjunctive grammar for the language $\{a^{4^n} : n \in \mathbb{N}\}$. We also generalize this result: for every set of natural numbers L we show that $\{a^n : n \in L\}$ is a conjunctive unary language, whenever the set of representations in base-k system of elements of L is regular, for arbitrary k.

Keywords: Language equations; conjunctive grammars; unary languages; regular languages.

1. Introduction

1.1. *Background*

Okhotin [1] introduced conjunctive grammars as a simple and powerful extension of context-free grammars. Informally speaking, conjunctive grammars allow intersections in the body of any rule of the grammar. More formally, conjunctive grammar is a quadruple $\langle \Sigma, N, P, S \rangle$ where Σ is a finite alphabet, N is a set of nonterminal symbols, $S \in N$ is a starting symbol and P is a set of productions of the form:

$$A \to \alpha_1 \& \alpha_2 \& \ldots \& \alpha_k, \quad \text{where } \alpha_i \in (\Sigma \cup N)^* . \tag{1}$$

Informally speaking, word w is derived by rule (1) if and only if (iff) it is derived from every string α_i for $i = 1, \ldots, k$.

We can also give semantics of a conjunctive grammars with resolved language equations that use union, intersection and concatenation. Language generated by conjunctive grammar is a component of the least solution of such equations.

The usage of intersection allows us to define many natural languages that are not context-free. On the other hand [1] conjunctive languages are computationally easy, that is they are in class $DTIME(n^3) \cap DSPACE(n)$.

Conjunctive grammars inherit many natural techniques and properties of context-free grammars: existence of the Chomsky normal form, parsing using a modification of CYK algorithm *etc.* On the other hand there is no Pumping Lemma for conjunctive grammars, they do not have bounded growth property, non-emptiness is undecidable. No technique for showing that a language is not conjunctive is known. We are not capable of separating conjunctive languages from context-sensitive languages.

For detailed results on conjunctive grammars see Okhotin [1], for shorter overview [2]. Work on the Boolean grammars [3], which extend conjunctive grammars by use of negation, is also suggested.

Okhotin [4] gathered nine most important open problems for conjunctive and Boolean grammars. One of them was a question, whether unary conjunctive languages are always regular. This holds for context-free grammars, and the same result was conjectured for conjunctive grammars. We disprove this conjecture by giving conjunctive grammar for a language $\{a^{4^n} : n \in \mathbb{N}\}$.

The set $\{4^n : n \in \mathbb{N}\}$ written in binary is a regular language. This leads to a natural question, what is the relation between regular (over arbitrary base-k alphabet) languages and unary conjunctive languages. We prove that every regular language (written in some base-k system) interpreted as a set of numbers can be represented by a conjunctive grammar over a unary alphabet.

1.2. *Outline of the paper*

We first briefly introduce the definition of conjunctive grammars in Section 2, we also give their semantics and facts about language equations. Then we present a conjunctive grammar for the language $\{a^{4^n} : n \in \mathbb{N}\}$, as an example for our technique, see Section 3. In Section 4 we consider the question of the number of nonterminals required to generate non-regular language. After those preliminaries we state and prove the main result of the paper in Section 5—for any set of natural numbers L, such that the representation in base-k system of L is regular, language $\{a^n : n \in L\}$ is a conjunctive unary language. In Section 6 we summarize the results and state open problems.

2. Definitions and Notation

2.1. *Language equations*

We gather the basic definitions and facts about language equations in this section.

Definition 1. *Let $(X_1, \ldots, X_n)$ be language variables. A resolved system of language equations is a system of a form*

$$X_i = \varphi_i(X_1, \ldots, X_n) \quad \text{for } i = 1, \ldots, n\,,$$

where each φ_i is an expression using language variables $X_1, \ldots, X_n$, constant languages and language operations.

We abbreviate such systems into vector form $(\ldots, X_i, \ldots) = \varphi(\ldots, X_i, \ldots)$, where $\varphi = (\ldots, \varphi_i, \ldots)$ is a vector of expressions. Note, that a solution of system $X = \varphi(X)$ is just a fixpoint of φ operator. Since fixpoints of operators are vectors of languages and we want to compare different fixpoints, we write $(\ldots, A_i, \ldots) \subseteq (\ldots, B_i, \ldots)$, meaning, that $A_i \subseteq B_i$ for $i = 1, \ldots, n$.

Language operation θ is *monotone* if

$$(\ldots, X_i, \ldots) \subseteq (\ldots, Y_i, \ldots) \quad \text{implies} \quad \theta(\ldots, X_i, \ldots) \subseteq \theta(\ldots, Y_i, \ldots) .$$

A sequence of languages L_i converges to L if

$$\forall_w \exists_n \forall_{m>n} (w \in L_m \Leftrightarrow w \in L)$$

and a sequence of vectors of languages converges if it converges on every coordinate. Language operation θ is *continuous* if for converging sequence of vectors of languages

$$(\ldots, {L^{(n)}}_i, \ldots) \to (\ldots, L_i, \ldots) \quad \text{implies} \quad \theta(\ldots, {L^{(n)}}_i, \ldots) \to \theta(\ldots, L_i, \ldots) .$$

For example intersection, union and concatenation are continuous and monotone. Also composition of continuous (monotone) operations is continuous (monotone).

Lemma 2. *Let φ be an operator using only monotone and continuous operations. Then it has a least fixpoint $(\ldots, S_i, \ldots)$ given by $\bigcup_{j=0}^{\infty} \varphi^j(\ldots, \emptyset, \ldots)$. If a vector of languages $(\ldots, X_i, \ldots)$ satisfies*

$$\varphi(\ldots, X_i, \ldots) \subseteq (\ldots, X_i, \ldots) \subseteq (\ldots, S_i, \ldots)$$

then $(\ldots, X_i, \ldots) = (\ldots, S_i, \ldots)$.

Proof. Consider $\varphi^j(\ldots, \emptyset, \ldots)$. We prove by induction on j, that $\varphi^j(\ldots, \emptyset, \ldots) \subseteq \varphi^{j+1}(\ldots, \emptyset, \ldots)$. Clearly this holds for $j = 0$ as $(\ldots, \emptyset, \ldots) \subseteq \varphi(\ldots, \emptyset, \ldots)$. Then by monotonicity of φ:

$$\varphi^{j+1}(\ldots, \emptyset, \ldots) = \varphi(\varphi^j(\ldots, \emptyset, \ldots)) \subseteq \varphi(\varphi^{j+1}(\ldots, \emptyset, \ldots)) = \varphi^{j+2}(\ldots, \emptyset, \ldots) .$$

Therefore $\bigcup_{j=0}^{\infty} \varphi^j(\ldots, \emptyset, \ldots) = \lim_{j\to\infty} \varphi^j(\ldots, \emptyset, \ldots)$. This, together with the continuity of φ, allows us to show that $\bigcup_{j=0}^{\infty} \varphi^j(\ldots, \emptyset, \ldots)$ is a fixpoint of φ:

$$\varphi(\lim_{j\to\infty} \varphi^j(\ldots, \emptyset, \ldots)) = \lim_{j\to\infty} \varphi(\varphi^j(\ldots, \emptyset, \ldots)) = \lim_{j\to\infty} \varphi^j(\ldots, \emptyset, \ldots) .$$

It is easy to see that $\lim_{j\to\infty} \varphi^j(\ldots, \emptyset, \ldots)$ is the least fixpoint: consider any fixpoint $(\ldots, A_i, \ldots)$ of φ. By induction $\varphi^j(\ldots, \emptyset, \ldots) \subseteq (\ldots, A_i, \ldots)$: clearly $(\ldots, \emptyset, \ldots) \subseteq (\ldots, A_i, \ldots)$ and for induction step

$$\varphi^{j+1}(\ldots, \emptyset, \ldots) = \varphi(\varphi^j(\ldots, \emptyset, \ldots)) \subseteq \varphi(\ldots, A_i, \ldots) = (\ldots, A_i, \ldots)$$

Consider now $(\ldots, X_i, \ldots)$ from the statement of the lemma. Then

$$(\ldots, \emptyset, \ldots) \subseteq (\ldots, X_i, \ldots) \subseteq (\ldots, S_i, \ldots) .$$

Since φ is monotone:

$$\varphi^k(\ldots,\emptyset,\ldots) \subseteq \varphi^k(\ldots,X_i,\ldots) \subseteq \varphi^k(\ldots,S_i,\ldots)\,.$$

As φ is continuous:

$$\bigcup_{k=0}^{\infty} \varphi^k(\ldots,\emptyset,\ldots) \subseteq \bigcup_{k=0}^{\infty} \varphi^k(\ldots,X_i,\ldots) \subseteq \bigcup_{k=0}^{\infty} \varphi^k(\ldots,S_i,\ldots)\,.$$

Since $\varphi(\ldots,X_i,\ldots) \subseteq (\ldots,X_i,\ldots)$ and by the definition of $(\ldots,S_i,\ldots)$:

$$(\ldots,S_i,\ldots) \subseteq (\ldots,X_i,\ldots) \subseteq (\ldots,S_i,\ldots)\,. \qquad \square$$

2.2. *Conjunctive grammars*

Definition 3 (Okhotin [1]) *A conjunctive grammar is a quadruple $G = \langle \Sigma, N, P, S\rangle$, in which Σ and N are disjoint finite non-empty sets of terminal and nonterminal symbols and P is a finite set of grammar rules of the form*

$$A \to \alpha_1 \& \ldots \& \alpha_n \quad (\text{where } A \in N,\ n \geq 1 \text{ and } \alpha_1,\ldots,\alpha_n \in (\Sigma \cup N)^*) \tag{2}$$

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

There are many ways of defining the semantics of conjunctive grammar, here we choose the formalism of language equations:

Definition 4 (Okhotin [5]) *For every conjunctive grammar $\langle \Sigma, N, P, S\rangle$, the associated system of language equations is a system of equations in variables N, in which variables assume values of languages over Σ, and which contains an equation:*

$$A = \bigcup_{A \to \alpha_1 \& \ldots \& \alpha_m \in P} \bigcap_{i=1}^{m} \alpha_i \tag{3}$$

for every variable A. Symbols $a \in \Sigma$ in such a system define a language $\{a\}$, while empty strings denote a language $\{\epsilon\}$. A solution of such a system is a vector of languages $(\ldots,L_C,\ldots)_{C\in N}$, such that the substitution of L_C for C, for all $C \in N$, turns each equation form the system into an equality.

Note, that language equations emerging from conjunctive grammars use only monotone and continuous operations.

In fact, we can go in the other direction easily—for every system of language equations of the form

$$A = \bigcup_{i} \bigcap_{j=1}^{m} \alpha_{i,j} \quad , \text{where } \alpha_{i,j} \in (A \cup \Sigma)^* \tag{4}$$

with variables $A \in N$ there exists a conjunctive grammar G with nonterminals N, such that the given system is a system of language equations associated with G. Therefore in the rest of the papers we use language equations of the form (4) instead of conjunctive grammars.

Example 5. *Let us consider conjunctive grammar* $\langle \Sigma, N, P, S \rangle$ *with* $\Sigma = \{a, b, c\}$, $N = \{S, B, C, E, A\}$. *The rules, corresponding language equations and their least solutions are as follows:*

$$\begin{array}{lll}
S \to (AE)\&(BC) & L_S = (L_A L_E) \cap (L_B L_C) & \{a^n b^n c^n : n \in \mathbb{N}\}, \\
A \to aA \mid \epsilon & L_A = \{a\} L_A \cup \{\epsilon\} & a^*, \\
B \to aBb \mid \epsilon & L_B = \{a\} L_B \{b\} \cup \{\epsilon\} & \{a^n b^n : n \in \mathbb{N}\}, \\
C \to Cc \mid \epsilon & L_C = \{c\} L_C \cup \{\epsilon\} & c^*, \\
E \to bEc \mid \epsilon & L_E = \{b\} L_E \{c\} \cup \{\epsilon\} & \{b^n c^n : n \in \mathbb{N}\}.
\end{array}$$

This technical lemma will be useful later, it asserts, that merging the variables does not decrease the least solution of the system of language equations.

Lemma 6. *Let a system of continuous and monotone language equations in variables* $X, Y, \{X_i\}_{i=1}^n$ *be given by*

$$\begin{array}{c}
X = \varphi_X(X, Y, \ldots, X_i, \ldots), \quad Y = \varphi_Y(X, Y, \ldots, X_i, \ldots) \\
(\ldots, X_i, \ldots) = \varphi(X, Y, \ldots, X_i, \ldots) \, .
\end{array}$$

And let $(S_X, S_Y, \ldots, S_i, \ldots)$ *be its least solution. Consider a system given by*

$$X = (\varphi_X \cup \varphi_Y)(X, X, \ldots, X_i, \ldots), \quad (\ldots, X_i, \ldots) = \varphi(X, X, \ldots, X_i, \ldots) \, .$$

and its least solution $(P_X, \ldots, P_i, \ldots)$. *Then* $P_X \supseteq S_X \cup S_Y$ *and* $(\ldots, P_i, \ldots) \supseteq (\ldots, S_i, \ldots)$.

Proof. The idea is very simple—at each iteration of the fixpoint operator, we get a larger set in the latter system. More formally, we prove by induction on j, that

$$(\varphi_X, \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots) \subseteq (\varphi_X \cup \varphi_Y, \varphi_X \cup \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)$$

Clearly this holds for $j = 0$, as both sides are $(\ldots, \emptyset, \ldots)$. For the induction step: the coordinates k except the first two leading ones: by the induction assumption

$$((\varphi_X \cup \varphi_Y, \varphi_X \cup \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)) \supseteq ((\varphi_X, \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots))$$

hence by monotonicity of φ_k:

$$\varphi_k((\varphi_X \cup \varphi_Y, \varphi_X \cup \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)) \supseteq \varphi_k((\varphi_X, \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)) \, .$$

For the leading coordinate:

$$\begin{array}{l}
(\varphi_X \cup \varphi_Y)((\varphi_X \cup \varphi_Y, \varphi_X \cup \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)) \supseteq \\
\qquad \supseteq (\varphi_X \cup \varphi_Y)((\varphi_X \cup \varphi_Y, \varphi_X \cup \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)) \supseteq \\
\qquad\qquad \supseteq (\varphi_X \cup \varphi_Y)((\varphi_X, \varphi_Y, \varphi)^j(\ldots, \emptyset, \ldots)) \, ,
\end{array}$$

the same goes for the second coordinate. Going to the limit:

$$(S_X, S_Y, \ldots, S_i, \ldots) \subseteq (P_X, P_Y, \ldots, P_i, \ldots) \, . \qquad \square$$

2.3. *Unary languages and sets of natural numbers*

In case of unary alphabet we identify word a^n with number n and work with sets of integers rather than with languages. The allowed operations are union, intersection and "concatenation", which interpreted in terms of numbers is an addition:

$$XY := \{x + y : x \in X, y \in Y\} . \tag{5}$$

Still we use words 'grammar' and 'language', as this is the main interest of this paper.

In many technical proofs we inspect the multiset of non-zero digits in base-k notation of natural numbers (for fixed k). Therefore we introduce notation of $\mathrm{Dig}_k(n)$—the multiset of non-zero digit of n and $\Sigma\mathrm{Dig}_k(n)$ for the sum of those digits. Formally for $n = \sum_{i=0}^{m} k^i d_i$, where $d_i \in \{0, 1 \ldots, k-1\}$ and $d_m \neq 0$:

$$\mathrm{Dig}_k(n) = \{d_i \mid i \in \{0, 1, \ldots m\}\ d_i \neq 0\} \quad \text{and} \quad \Sigma\mathrm{Dig}_k(n) = \sum_{i=0}^{m} d_i .$$

If the value of k is clear from the context we omit it and write Dig and ΣDig instead of Dig_k and $\Sigma\mathrm{Dig}_k$.

Fact 7. *If equation* $\sum_i n_i = \sum_i m_i$ *holds for natural numbers* $\{n_i, m_i\}$ *then for every* k $\sum_i(\Sigma\mathrm{Dig}_k(n_i)) \equiv_{k-1} \sum_i(\Sigma\mathrm{Dig}_k(m_i))$.

Proof. Consider the process of calculating the sum $\sum_i n_i$ symbol after symbol in base-k positional notation. Then if there is a carry the sum of digits in one column decreases by k and the sum of symbols in the other increases by 1 in the following column. Hence

$$\sum_i \Sigma\mathrm{Dig}_k(n_i) \equiv_{k-1} \Sigma\mathrm{Dig}_k(\sum_i n_i) = \Sigma\mathrm{Dig}_k(\sum_i m_i) \equiv_{k-1} \sum_i \Sigma\mathrm{Dig}_k(m_i). \qquad \square$$

We also extend the those notions to sets of numbers in the usual manner, that is

$$\mathrm{Dig}_k(A) = \{\mathrm{Dig}_k(n) \mid n \in A\} \quad \text{and} \quad \Sigma\mathrm{Dig}_k(A) = \{\Sigma\mathrm{Dig}_k(n) \mid n \in A\} .$$

3. Toy Example

Let us define the following sets of integers (here $\mathbb{N} = \{0, 1, \ldots\}$ denotes the set of natural numbers):

$$A_i = \{i \cdot 4^n : n \in \mathbb{N}\}, \text{ for } i = 1, 2, 3 \qquad A_{12} = \{6 \cdot 4^n : n \in \mathbb{N}\} \tag{6}$$

The indices reflect the fact that these sets consist of numbers that written in base-4 positional system begin with digits $1, 2, 3, 12$, respectively and have only 0'es afterwards. We show that those sets are the least solution of the equations:

$$B_1 = (B_2B_2 \cap B_1B_3) \cup \{1\} , \tag{7}$$

$$B_2 = (B_{12}B_2 \cap B_1B_1) \cup \{2\} , \tag{8}$$

$$B_3 = (B_{12}B_{12} \cap B_1B_2) \cup \{3\} , \tag{9}$$

$$B_{12} = (B_3B_3 \cap B_1B_2) . \tag{10}$$

Note, that none of those sets is regular.

Lemma 8. *Every solution* (S_1, S_2, S_3, S_{12}) *of* (7)–(10) *satisfies:*

$$(A_1, A_2, A_3, A_{12}) \subseteq (S_1, S_2, S_3, S_{12}). \tag{11}$$

Proof. We prove by induction on m that for $i \in \{1, 2, 3, 12\}$ if $m \in A_i$ then $m \in S_i$.

Induction basis: consider $m = 1, 2, 3$ then $m \in A_m$. By (7)–(9) $m \in S_m$.

Induction step: let $m = 4^{n+1} \in A_1$. By induction assumption $2 \cdot 4^n \in S_2$ and hence $(2 \cdot 4^n) + (2 \cdot 4^n) = 4^{n+1} \in S_2 S_2$. Also by induction assumption $4^n \in S_1$ and $3 \cdot 4^n \in S_3$, hence $4^n + (3 \cdot 4^n) = 4^{n+1} \in S_1 S_3$, and so $4^{n+1} \in S_2 S_2 \cap S_1 S_3$ and by (7) we conclude that $4^{n+1} \in S_1$.

If $m = 6 \cdot 4^n \in A_{12}$ then by induction assumption $3 \cdot 4^n \in S_3$, $2 \cdot 4^n \in S_2$ and $4^{n+1} = 4 \cdot 4^n \in S_1$. Hence $6 \cdot 4^n \in S_3 S_3 \cap S_1 S_2$ and by (10) we get $6 \cdot 4^n \in S_{12}$.

If $m = 2 \cdot 4^{n+1} \in A_2$ we use the fact that $2 \cdot 4^n \in S_2$, $6 \cdot 4^n \in S_{12}$ and $4^{n+1} \in S_1$. Hence $2 \cdot 4^{n+1} \in S_1 S_1 \cap S_{12} S_2$ and by (8) $2 \cdot 4^{n+1} \in S_2$.

If $m = 3 \cdot 4^{n+1} \in A_3$ then by $2 \cdot 4^{n+1} \in S_2$, $6 \cdot 4^n \in S_{12}$ and $4^{n+1} \in S_1$ hence $3 \cdot 4^{n+1} \in S_{12} S_{12} \cap S_1 S_2$ and by (9) $3 \cdot 4^{n+1} \in S_3$. This ends induction step. □

Lemma 9. *Sets* (A_1, A_2, A_3, A_{12}) *satisfy*

$$A_1 \supseteq (A_2 A_2 \cap A_1 A_3) \cup \{1\}\,, \tag{12}$$

$$A_2 \supseteq (A_{12} A_2 \cap A_1 A_1) \cup \{2\}\,, \tag{13}$$

$$A_3 \supseteq (A_{12} A_{12} \cap A_1 A_2) \cup \{3\}\,, \tag{14}$$

$$A_{12} \supseteq (A_3 A_3 \cap A_1 A_2)\,. \tag{15}$$

Proof. The idea of the proof is quite natural—to exploit the base-4 positional notation. For each equation we consider any number that can appear on the right-hand side. Then we analyse the positions of its non-zero digits, as the set on the left-hand side constitutes of numbers with non-zero digits on leading position (or two leading positions in case of A_{12}). This analysis is performed by case inspection of possible ways of obtaining an element on the right-hand side of the equation. In each such case we use the fact that this number is built by arithmetic operations from number with only one (two) non-zero digits and deal with the possible alignments of non-zero digits. It is easy to identify the good cases and to show that the bad ones are eliminated by the intersection.

Consider first (12). Let m belong to the right-hand side of (12). If $m = 1$ then $m \in A_1$ by definition. Let $m \neq 1$ and therefore $m \in A_2 A_2 \cap A_1 A_3$. There are numbers $k, l \in A_2$ and $m = k + l$. Either $k = l$ and $\mathrm{Dig}(m) = \{1\}$ or $k \neq l$ and $\mathrm{Dig}(m) = \{2, 2\}$. On the other hand $m \in A_1 A_3$, so there are $k' \in A_1$, $l' \in A_3$ such that $m = l' + k'$. Either $\mathrm{Dig}(m) = \{1\}$, if $l' = 3k'$ or $\mathrm{Dig}(m) = \{1, 3\}$. Since $m \in A_2 A_2 \cap A_1 A_3$ we conclude that $\mathrm{Dig}(m) = \{1\}$ and therefore $m \in A_1$.

Consider (13). Let m belong to the right-hand side of (13). If $m = 2$ then $m \in A_2$. So let $m \neq 2$, in particular $m \in A_{12} A_2 \cap A_1 A_1$. There are numbers

$k, l \in A_1$ and $m = k + l$. Note, that $\Sigma\mathrm{Dig}(k) + \Sigma\mathrm{Dig}(l) = 2$. On the other hand there are $k' \in A_{12}$, $l' \in A_2$ such that $m = l' + k'$. Here $\Sigma\mathrm{Dig}(l') + \Sigma\mathrm{Dig}(k') = 5$. The equation $k + l = k' + l'$ implies, that there is a carry in $k' + l'$, hence $k' = 3l'$ and so $k' + l' = 4l' \in A_2$.

Consider (14). Let m belong to the right-hand side of (14). If $m = 3$ then $m \in A_3$. Otherwise $m \neq 3$ and $m \in A_{12}A_{12} \cap A_1A_2$. Then $m \in A_{12}A_{12}$ and so there are $k, l \in A_{12}$ and $m = k + l$. Calculating the sums of the digits yields $\Sigma\mathrm{Dig}(k) + \Sigma\mathrm{Dig}(l) = 6$. On the other hand $m \in A_1A_2$, so there are $k' \in A_1$, $l' \in A_2$ such that $m = l' + k'$. Here the sum of digits is $\Sigma\mathrm{Dig}(k') + \Sigma\mathrm{Dig}(l') = 3$ and since $k + l = k' + l'$ there is a carry in $k + l$, but this is possible only when $k = l$ and in such case $k + l \in A_3$.

Consider (15). Let m belong to the right-hand side of (15), that is $m \in A_3A_3 \cap A_1A_2$. There are numbers $k, l \in A_3$ such that $m = k + l$, therefore $\Sigma\mathrm{Dig}(k) + \Sigma\mathrm{Dig}(l) = 6$. On the other hand $m \in A_1A_2$ and there are $k' \in A_1$, $l' \in A_2$ such that $m = l' + k'$. Here $\Sigma\mathrm{Dig}(k') + \Sigma\mathrm{Dig}(l') = 3$ and since $k + l = k' + l'$ there is a carry in $k + l$. This is possible only when $k = l$ and then $k + l \in A_{12}$. □

Theorem 10. *Sets A_1, A_2, A_3, A_{12} are the least solution of* (7)–(10).

Proof. By Lemma 2 it is enough to show that (A_1, A_2, A_3, A_{12}) are included in every solution and that $\varphi(A_1, A_2, A_3, A_{12}) \subseteq (A_1, A_2, A_3, A_{12})$. The former was shown in Lemma 8 and the latter in Lemma 9. □

4. Number of Nonterminals Required

We say that the conjunctive grammar is in Chomsky normal form, if for every production

$$A \to \alpha_1 \& \alpha_2 \& \ldots \& \alpha_k$$

- each α_i consists of two nonterminals or it is a single terminal symbol,
- there are no productions with ϵ on the right-hand side, except for S
- if there is a production $S \to \epsilon$ then S does not appear on the right-hand side of any production.

The language equations described in the previous section is equivalent to a conjunctive grammar that uses four nonterminals. It is easily converted to Chomsky normal form—we introduce two new nonterminals for languages $\{1\}$ and $\{2\}$, respectively. Hence grammar for language $\{4^n : n \in \mathbb{N}\}$ in Chomsky normal form requires at most six nonterminals.

As context-free grammars can generate only regular languages, it is an interesting question, how complicated the conjunctive grammars have to be to generate non-regular language? For example—how many nonterminals are required? How many of them must generate non-regular languages? How many intersections are

needed? Putting this question in the other direction, are there any natural sufficient conditions for a conjunctive grammar to generate regular language?

We are able to reduce the number of nonterminals in our construction to three, but we sacrifice Chomsky normal form and introduce also concatenations of three nonterminals in productions. This can be seen as trade-off between number of nonterminals and length of concatenations. Consider language equations:

$$B_1 = (B_{2,12}B_{2,12} \cap B_1B_3) \cup \{1\}\,, \tag{16}$$

$$B_{2,12} = \Big((B_{2,12}B_{2,12} \cap B_1B_1) \cup \{2\}\Big) \cup \Big((B_3B_3 \cap B_{2,12}B_{2,12})\Big)\,, \tag{17}$$

$$B_3 = (B_{2,12}B_{2,12} \cap B_1B_1B_1) \cup \{3\}\,. \tag{18}$$

These are basically the same equations as (7)–(10), except that variables B_2 and B_{12} are identified (or merged) and B_2B_1 in (9) was changed to $B_1B_1B_1$.

Theorem 11. *The least solution of* (16)–(18) *is* $(A_1,\ A_2 \cup A_{12},\ A_3)$.

Proof. The proof is a modification of the proof of Theorem 10. The main idea is to think of nonterminal $B_{2,12}$ that corresponds to the set $A_2 \cup A_{12}$ as two nonterminals: B_2 and B_{12}, corresponding to sets A_2 and A_{12}, respectively.

Firstly it is easy to check that replacing B_2B_1 with $B_1B_1B_1$ in (9) requires only small changes of proofs of Lemma 8 and Lemma 9.

Let $(S_1, S_{2,12}, S_3)$ be the least solution of (16)–(18). We want to show, that $(A_1, A_2 \cup A_{12}, A_3) \subseteq (S_1, S_{2,12}, S_3)$, in analogy to Lemma 8. It is enough to show, that $S_{2,12}$ is a superset of both A_2 and A_{12}. This follows from Lemma 6.

Now we show that:

$$A_1 \supseteq \Big((A_2 \cup A_{12})(A_2 \cup A_{12}) \cap A_1A_3\Big) \cup \{1\}\,, \tag{19}$$

$$A_2 \supseteq \Big((A_2 \cup A_{12})(A_2 \cup A_{12}) \cap A_1A_1\Big) \cup \{2\}\,, \tag{20}$$

$$A_{12} \supseteq A_3A_3 \cap (A_2 \cup A_{12})(A_2 \cup A_{12})\,, \tag{21}$$

$$A_3 \supseteq \Big((A_2 \cup A_{12})(A_2 \cup A_{12}) \cap A_1A_1A_1\Big) \cup \{3\}\,. \tag{22}$$

These equations are similar to (12)–(15), apart that on the right-hand side each A_2 and A_{12} was replaced by $A_2 \cup A_{12}$. We show, that each $A_2 \cup A_{12}$ can be replaced by exactly one from the pair A_2 or A_{12} (in fact the one that was originally in the equation in question) and keep the value of the right-hand side constant.

Consider (19). Let $n_1 \in A_1$, $n_2 \in A_3$, $m_1, m_2 \in A_{2,12}$. As $n_1 + n_2 = m_1 + m_2$ then by Fact 7: $\Sigma\mathrm{Dig}(n_1) + \Sigma\mathrm{Dig}(n_2) \equiv_3 \Sigma\mathrm{Dig}(m_1) + \Sigma\mathrm{Dig}(m_2)$. As $\Sigma\mathrm{Dig}(n_1) + \Sigma\mathrm{Dig}(n_2) = 4$ we get that $\Sigma\mathrm{Dig}(m_1) + \Sigma\mathrm{Dig}(m_2) = 4$, as they clearly cannot exceed 6 or be 1. We conclude that $\Sigma\mathrm{Dig}(m_1) = \Sigma\mathrm{Dig}(m_2) = 2$ and hence we can replace each $A_{2,12}$ by A_2.

Consider (20). We use the same type of argument as in the previous case. Let $n_1, n_2 \in A_1$ and $m_1, m_2 \in A_2 \cup A_{12}$ such that $n_1 + n_2 = m_1 + m_2$. Thus $\Sigma\mathrm{Dig}(n_1) + \Sigma\mathrm{Dig}(n_2) = 2$. On the other hand $m_1, m_2 \in A_2 \cup A_{12}$, hence $\Sigma\mathrm{Dig}(m_1), \Sigma\mathrm{Dig}(m_2) \in$

$\{2,3\}$. By Fact 7: $\Sigma\mathrm{Dig}(m_1)+\Sigma\mathrm{Dig}(m_2)=5$, as this sum clearly cannot exceed 8 or be 2. Therefore exactly one from m_1, m_2 is in A_2 and the other in A_{12}. Therefore we can replace $(A_2 \cup A_{12})(A_2 \cup A_{12})$ by A_2A_{12}.

Consider (21). Let $n_1, n_2 \in A_3$ and $m_1, m_2 \in A_2 \cup A_{12}$ such that $n_1+n_2 = m_1+m_2$. Then $\Sigma\mathrm{Dig}(n_1)+\Sigma\mathrm{Dig}(n_2)=6$ and $\Sigma\mathrm{Dig}(m_1)+\Sigma\mathrm{Dig}(m_2) \in \{4,5,6\}$, and therefore by Fact 7: $\Sigma\mathrm{Dig}(m_1)+\Sigma\mathrm{Dig}(m_2)=6$ which implies that $m_1, m_2 \in A_{12}$. Hence we can remove A_2 from this equation.

Consider (22) and let $n_1, n_2, n_3 \in A_1$, $m_1, m_2 \in A_2 \cup A_{12}$ such that $n_1+n_2+n_3 = m_1+m_2$. Then $\Sigma\mathrm{Dig}(n_1)+\Sigma\mathrm{Dig}(n_2)+\Sigma\mathrm{Dig}(n_3)=3$ and $\Sigma\mathrm{Dig}(m_1)+\Sigma\mathrm{Dig}(m_2) \in \{4,5,6\}$, hence by Fact 7 we obtain $\Sigma\mathrm{Dig}(m_1)+\Sigma\mathrm{Dig}(m_2)=6$ which implies that $m_1, m_2 \in A_{12}$. Hence we can remove the A_2 from (22).

Hence without the change of the right-hand side we can transform the (19)–(22) into (12)–(15). And the latter are satisfied by the Lemma 9, thus (19)–(22) are satisfied as well. Then by Lemma 2 the theorem follows. □

5. Languages Regular in Base-k Notation

We give a major generalization of Theorem 10. Let $\Sigma_k = \{0,\ldots,k-1\}$. We deal with languages $\{a^n : n \in L\}$, where L is some regular language over Σ_k. From the following on we consider regular languages over Σ_k for some k that do not have words with leading 0, since this is meaningless in case of numbers. Still, as for regular language L language $L' = (0^*)^{-1}L$ is regular. Both L and L' represent the same set of numbers. We consider only the languages without leading 0.

Definition 12. *Let $w \in \Sigma_k^*$ be a word. We define its unary representation as*

$$f_k(w) = \{a^n : w \text{ read as base-}k \text{ number is } n\}\,. \tag{23}$$

We also apply f_k to languages with an obvious meaning: $f_k(S) = \{f_k(n) \mid n \in S\}$.

Fact 13. *For every $k = l^n$, $n > 0$ and every unary language L language $f_k^{-1}(L)$ is regular iff language $f_l^{-1}(L)$ is regular.*

It is technically much easier to deal with larger values of k, as for small k there is only a small set of digits and hence it is harder to cut out unwanted results in intersection. In the following we use 'big enough' k, that is we focus on $k \geq 9$. We claim, that for regular L language $f_k(L)$ is unary conjunctive.

As we use positional notation extensively, it is convenient to think of number as string of digits. Hence we write $n = ijw$, meaning that w begins with digit i, then digit j follows and then a string of digits, represented by a string w.

5.1. *Languages with two leading digits fixed*

Theorem 14. *Let $k > 8$ be a natural number. For every $i \in \{1,\ldots,k-1\}$ there is a conjunctive grammar over unary alphabet generating language $\{i \cdot k^n : n \in \mathbb{N}\}$.*

For every $i, j \in \{1, \ldots, k-1\}$ there is a conjunctive grammar over unary alphabet generating language $\{(ki+j) \cdot k^n : n \in \mathbb{N}\}$.

Proof. We introduce variables $B_{i,j}$ for $i = 1, \ldots k-1$ and $j = 0, \ldots, k-1$, with intention that $B_{i,j}$ defines language of numbers beginning with digits i, j and then only zeroes in base-k positional system. We show that sets

$$L_{i,j} = \{(k \cdot i + j) \cdot k^n : n \in \mathbb{N}\} \quad \text{for } j \neq 0 \quad L_{i,0} = \{i \cdot k^n : n \in \mathbb{N}\} .$$

are the least solution of the system

$$B_{1,j} = \bigcap_{n=1}^{2} B_{k-n,0} B_{j+n,0} \cup \{1 : j = 0\} \qquad \text{for } j = 0, 1, 2 \tag{24}$$

$$B_{i,j} = \bigcap_{n=1}^{2} B_{i-1,k-n} B_{j+n,0} \cup \{i : j = 0\} \qquad \text{for } j = 0, 1, 2 \text{ and } i > 1 \tag{25}$$

$$B_{i,j} = \bigcap_{n=1}^{2} B_{i,j-n} B_{n,0} \cap B_{i,0} B_{j,0} \qquad \text{for } j > 2 \tag{26}$$

The proof follows by Lemma 15 and Lemma 16. □

Lemma 15. *Every solution $(\ldots, S_{i,j}, \ldots)$ of (24)–(26) satisfies $(\ldots, S_{i,j}, \ldots) \supseteq (\ldots, L_{i,j}, \ldots)$.*

Proof. It is enough to show, that for every number n and every pair of indices (i, j), if $n \in L_{i,j}$ then $n \in S_{i,j}$. We prove it by induction on n. The idea is to use the fact, that all the smaller numbers are already proved to be in the proper $S_{i',j'}$ and therefore we may use them to construct the new number.

Induction basis: $n \in L_{n,0}$ for $n < k$ and this is the only set it is in, by the definition of $L_{i,j}$. By (24) and (25) we conclude that also $n \in S_{n,0}$.

Induction step: let $n = ij0^m \geq k$ thus $n \in L_{i,j}$, by the definition. The equations defining $S_{i,j}$ are different, depending on i, j, therefore we have to inspect three different cases, for three different types of equations defining $S_{i,j}$.

If $i = 1$ and $j < 3$ then $S_{1,j}$ is the solution to the (24). By induction assumption $(k-1)0^m \in S_{k-1,0}$ and $(j+1)0^m \in S_{j+1,0}$, therefore we may use them on the right-hand side of (24). Adding those two numbers yields $(k-1)0^m + (j+1)0^m = 1j0^m \in S_{k-1,0} S_{j+1,0}$. We deal with the second conjunct: by the induction assumption $(k-2)0^m \in S_{k-2,0}$ and $(j+2)0^m \in S_{j+2,0}$, therefore we can use them on the right-hand side of (24). By adding those two numbers we obtain $(k-2)0^m + (j+2)0^m = 1j0^m \in S_{k-2,0} S_{j+2,0}$. Joining those two results: $n = 1j0^m \in S_{k-1,0} S_{j+1,0} \cap S_{k-2,0} S_{j+2,0} \subseteq S_{1,j}$, by (24).

The second case, for $i > 1$ and $j < 3$ is associated with (25). We focus on the first conjunct first. By induction assumption $(i-1)(k-1)0^m \in S_{i-1,k-1}$ and $(j+1)0^m \in S_{j+1,0}$ and thus we may use them on the right-hand side of the equation. By adding we obtain $(i-1)(k-1)0^m + (j+1)0^m = ij0^m = n \in S_{i-1,k-1} S_{j+1,0}$. We move

to the second conjunct: by the induction assumption $(i-1)(k-2)0^m \in S_{i-1,k-2}$ and $(j+2)0^m \in S_{j+2,0}$, we use them for adding on the right-hand side of (25). Adding $(i-1)(k-2)0^m + (j+2)0^m = ij0^m = n \in S_{i-1,k-2}S_{j+2,0}$. By comparing the obtained results one gets $ij0^m = n \in S_{i-1,k-2}S_{j+2,0} \cap S_{i-1,k-1}S_{j+1,0} = S_{i,j}$, by (25).

The last remaining case is for $j > 2$, described in (26). By induction assumption $i(j-1)0^m \in S_{i,j-1}$ and $10^m \in S_{1,0}$ thus we may use them on the right-hand side of (26). Adding those two numbers: $i(j-1)0^m + 10^m = ij0^m = n \in S_{i,j-1}S_{1,0}$. Also by induction assumption $i(j-2)0^m \in S_{i,j-2}$ and $20^m \in S_{2,0}$, we use those numbers in adding on the right-hand side of (26): $i(j-2)0^m + 20^m = ij0^m = n \in S_{i,j-2}S_{2,0}$. Again by induction assumption $i00^m \in S_{i,0}$ and $j0^m \in S_{j,0}$, we add them as part of the right-hand side of (26): $i00^m + j0^m = ij0^m = n \in S_{i,0}S_{j,0}$. Hence $ij0^m = n \in S_{i,j}$, by (26). □

Lemma 16. *Languages* $(\ldots, L_{i,j}, \ldots)$ *satisfy* $\varphi(\ldots, L_{i,j}, \ldots) \subseteq (\ldots, L_{i,j}, \ldots)$.

Proof. This proof is just an elaboration of the proof of Lemma 9. As in Lemma 9 we exploit the base-k notation. Numbers occurring on the right-hand side of the $\varphi_{i',j'}(\ldots, L_{i,j}, \ldots)$ have very small amount of non-zero digits. We inspect case by case the relative alignment of those digits and prove, that many possibilities are cancelled out by the intersection and the only one remaining are those in the set $L_{i',j'}$.

Consider first (24) and a number that belongs to the right-hand side after substituting $L_{i',j'}$ for variables $B_{i',j'}$. If this number is 1 then it clearly belongs to the left-hand side. Otherwise let $n \in L_{k-1,0}$, $n' \in L_{j+1,0}$ and $m \in L_{k-2,0}$, $m' \in L_{j+2,0}$ such that $n+n' = m+m'$. If there is a carry in $n+n'$ then $n+n' \in L_{i,j}$, also a carry in $m+m'$ implies $m+m' \in L_{1,j}$. So consider the case, when there is no carry in both $n+n'$ and $m+m'$. Hence $\mathrm{Dig}(n+n') = \{j+1, k-1\}$ and $\mathrm{Dig}(m+m') = \{j+2, k-2\}$. Since $j+1 < j+2 \leq k-2 < k-1$ we conclude that $n+n' \neq m+m'$. And so the only possible case is when $n+n' = m+m' \in L_{1,j}$.

Consider the second equation in question, that is (25) and any number belonging to the right-hand side after substituting $L_{i',j'}$ for variables $B_{i',j'}$. If the number in question is i, then it belongs to $L_{i,0}$ and thus to the left-hand side. Let $n \in L_{i-1,k-1}$, $n' \in L_{j+1,0}$ and $m \in L_{i-1,k-2}$, $m' \in L_{j+2,0}$ such that $n+n' = m+m'$. Since $\Sigma\mathrm{Dig}(n) + \Sigma\mathrm{Dig}(n') = i+j+k-1 = \Sigma\mathrm{Dig}(m) + \Sigma\mathrm{Dig}(m')$, there is either no carry of digits in both sums or there is a carry in both sums (in both cases it is not possible to have two carries). If there is no carry and we have three non-zero digits in both sums then $\mathrm{Dig}(n+n') = \{i-1, k-1, j+1\}$ and $\mathrm{Dig}(m+m') = \{i-1, k-2, j+2\}$. They must equal and so $\{k-1, j+1\} = \{k-2, j+2\}$, which is not possible, since $j+1 < j+2 \leq k-2 < k-1$. If there is no carry and there are two non-zero digits then in $n+n'$ the leading digit is $i+j$ and in $m+m'$ it is $i+j+1$, contradiction. If a carry in one of the sums $n+n$ or $m+m'$ results in $n+n' \in L_{i,j}$ or $m+m' \in L_{i,j}$ then we are done. Those good add-ups are when in $n+n'$ digit $(j+1)$

adds up with $(k-1)$ or in $m+m'$ digit $(j+2)$ adds up with $(k-2)$. So we may restrict ourselves to the cases, when the digits add up in different configurations. This is possible only when $(j+1)$ adds up with $(i-1)$ in $n+n'$ and $(j+2)$ adds up with $(i-1)$ in $m+m'$. Then in $n+n'$ the second digit is $i+j-k$ and in $m+m'$ the digit is $i+j+1-k$, contradiction. This ends the case inspection for this equation.

Consider the last equation, (26). We inspect the numbers that can appear on the right-hand side of this equation after substituting sets $L_{i',j'}$ for variables $B_{i',j'}$. Let $n \in L_{i,j-1}$, $n' \in L_{1,0}$ and $m \in L_{i,j-2}$, $m' \in L_{2,0}$ and $p \in L_{i,0}$, $p' \in L_{j,0}$ such that $n+n' = m+m' = p+p'$. Again, $\Sigma\mathrm{Dig}(n)+\Sigma\mathrm{Dig}(n') = \Sigma\mathrm{Dig}(m)+\Sigma\mathrm{Dig}(m') = \Sigma\mathrm{Dig}(p)+\Sigma\mathrm{Dig}(p')$ and so either there is no carry in all of the sums or exactly one carry in each sum (clearly $p+p'$ cannot have two carries). Since digits $i, j-1, 1$ from $n+n'$ are all non-zero, then $n+n'$ has at least two non-zero digits. Since $p+p'$ has at most two non-zero digits, then there are exactly two non-zero digits in the result. Suppose that there was a carry and we ended up with two non-zero digits. Then $p+p'$ has 1 as its first digit and $(i+j-k)$ as the second an then only 0's. On the other hand $n+n'$ has 1 as the first digit, 0 as the second and $j-1>0$ as the third, contradiction. And so there is no carry. If as a result of adding at least one of the sums $n+n'$, $m+m'$ or $p+p'$ is in $L_{i,j}$, then we are done. So we can deal only with the case when all the sums are outside $L_{i,j}$. Then $\mathrm{Dig}(n+n') = \{i+1, j-1\}$, $\mathrm{Dig}(m+m') = \{i+2, j-2\}$ and $\mathrm{Dig}(p+p') = \{i, j\}$. Contradiction. This ends this case and concludes the proof. □

5.2. *Any regular language*

We now define the resolved language equations for fixed regular language $L \subseteq \Sigma_k^* \setminus 0\Sigma_k^*$. Let $M = \langle \Sigma_k, Q, \delta, F, q_0 \rangle$ be a (non-deterministic) automaton recognizing L reading it from the right to the left. For technical reasons we choose $k > 8$. The set of variables is

$$N = \{A_{i,j,q}, A_{i,j} : 1 \leq i < k,\ 0 \leq j < k,\ q \in Q\} \cup \{S\}\ . \tag{27}$$

We intend to construct a system of language equations with the least solution

$$L(A_{i,j}) = \{n\ :\ f_k^{-1}(n) = ij0^\ell \text{ for some natural } \ell\}\ , \tag{28}$$

$$L(A_{i,j,q}) = \{n : f_k^{-1}(n) = ijw,\ \delta(q_0, w, q)\}\ , \tag{29}$$

$$L(S) = f_k(L)\ . \tag{30}$$

We denote sets defined by (29) as $L_{i,j,q}$ and sets defined by (28) as $L_{i,j}$.

By Theorem 14 sets $L_{i,j}$ can be defined by resolved language equations, and so we focus on equations for $A_{i,j,q}$.

$$A_{i,j,q} = \bigcup_{\substack{(x,q'):\\ \delta(q',x,q)}} \bigcap_{n=0}^{3} A_{i,n}A_{j-n,x,q'} \cup \{ij : q_0 = q\} \qquad \text{for } j > 3,\ i \neq 0 \tag{31}$$

$$A_{i,j,q} = \bigcup_{\substack{(x,q'):\\ \delta(q',x,q)}} \bigcap_{n=1}^{4} A_{i-1,j+n}A_{k-n,x,q'} \cup \{ij : q_0 = q\} \qquad \text{for } j < 4,\ i \neq 0, 1 \tag{32}$$

$$A_{1,j,q} = \bigcup_{\substack{(x,q'):\\ \delta(q',x,q)}} \bigcap_{n=1}^{4} A_{k-n,0}A_{j+n,x,q'} \cup \{1j : q_0 = q\} \qquad \text{for } j < 4 \tag{33}$$

$$S = (L \cap \Sigma_k) \cup \bigcup_{\substack{q,i,j:\\ \delta(q,ij)\cap F\neq\emptyset}} A_{i,j,q} \tag{34}$$

We prove that $(\ldots, L_{i,j,q}, \ldots)$ is the least solution of (31)–(33).

Lemma 17. *For $k > 8$ the least solution $(\ldots, X_{i,j,q}, \ldots)$ of (31)–(33) satisfies*

$$(\ldots, L_{i,j,q}, \ldots) \subseteq (\ldots, X_{i,j,q_i}, \ldots) .$$

Proof. We prove by induction on n that for every $n > 1$ and every $i, j \in \Sigma_k$, $q \in Q$ if $n \in L_{i,j,q}$ then also $n \in X_{i,j,q}$. When $n = ij$ then this is obvious, as the last set in the union in (31)–(32) is $\{ij \mid q_0 = q\}$.

The inductive step is similar to the inductive step in Lemma 8: for a fixed number n from $L_{i,j,q}$ we point smaller numbers form sets $L_{i,j,q}$ and $L_{i,j}$ that substituted into the right-hand side generate n.

Let $n = ijw$ and $w = xw'$. Let p be a state such that $\delta(q_0, w', p)$ and $\delta(p, x, q)$. We want to prove, that $n \in L_{i,j,q}$. As the equations defining $A_{i,j,q}$ are different for different i, j, our proofs splits depending on the value of i and j.

Suppose $j > 3$, by induction assumption $(j-m)xw' \in X_{j-m,x,p}$ and $im0^{|w'|+1} \in L_{i,m}$ for $m = 0, \ldots, 3$. Adding the appropriate numbers according to the equation:

$$\begin{aligned}
jxw' + i00^{|w'|+1} &= ijw \in X_{j,x,p}L_{i,0} ,\\
(j-1)xw' + i10^{|w'|+1} &= ijw \in X_{j-1,x,p}L_{i,1} ,\\
(j-2)xw' + i20^{|w'|+1} &= ijw \in X_{j-2,x,p}L_{i,2} ,\\
(j-3)xw' + i30^{|w'|+1} &= ijw \in X_{j-3,x,p}L_{i,3}
\end{aligned}$$

Thus $ijw \in \cap_{m=0}^{3} X_{j-m,x,p}L_{i,m}$ and by (31) $ijw \in X_{i,j,q}$.

Suppose $j < 4$ and $i > 1$. Induction assumption gives us $(k-m)xw' \in X_{k-m,x,p}$ and $(i-2)(j+m)0^{|w'|+1} \in L_{i-1,j+m}$ for $m = 1, \ldots, 4$. We add the numbers according

to the equation:

$$(k-1)xw' + (i-1)(j+1)0^{|w'|+1} = ijw \in X_{k-1,x,p}L_{i-1,j+1} ,$$
$$(k-2)xw' + (i-1)(j+2)0^{|w'|+1} = ijw \in X_{k-2,x,p}L_{i-1,j+2} ,$$
$$(k-3)xw' + (i-1)(j+3)0^{|w'|+1} = ijw \in X_{k-3,x,p}L_{i-1,j+3} ,$$
$$(k-4)xw' + (i-1)(j+4)0^{|w'|+1} = ijw \in X_{k-4,x,p}L_{i-1,j+4} .$$

Thus $ijw \in \bigcup_{m=1}^{4} X_{k-m,x,p}L_{j+m,0}$ and by (32) $ijw \in X_{i,j,q}$.

Suppose $j < 4$ and $i = 1$, by induction assumption: $(j+m)xw' \in X_{j+m,x,p}$ and $(k-m)0^{|w'|+1} \in L_{k-m,0}$ for $m = 1, \ldots, 4$. Adding according to the equation:

$$(j+1)xw' + (k-1)0^{|w'|+1} = 1jw \in X_{j+1,x,p}L_{k-1,0} ,$$
$$(j+2)xw' + (k-2)0^{|w'|+1} = 1jw \in X_{j+2,x,p}L_{k-2,0} ,$$
$$(j+3)xw' + (k-3)0^{|w'|+1} = 1jw \in X_{j+3,x,p}L_{k-3,0} ,$$
$$(j+4)xw' + (k-4)0^{|w'|+1} = 1jw \in X_{j+4,x,p}L_{k-4,0} .$$

Hence by (33) $1jw \in X_{1,j,q}$. □

Lemma 18. *For $k > 8$ languages $L_{i,j,q}$ satisfy $(\ldots, L_{i,j,q}, \ldots) \supseteq \varphi(\ldots, L_{i,j,q}, \ldots)$.*

Proof. This Lemma has a proof similar to the proof of Lemma 9. We proceed by induction. We inspect the numbers that can occur as the concatenation on the right-hand side of the equations. We show that either they are of the desired form, or they are not in the intersection—this can be shown by inspecting the positions of different digits in those conjuncts. We intersect four conjuncts. Every conjunct is a sum of two numbers with specified leading digits. We want those numbers to be properly arranged. It can happen, that one number is too big (leading digits are left of the intended position) or to small (leading digits are right of the intended position). As in every conjunct the leading digits are different, for an unintended number to prevail in the intersection we have to combine many different arrangements in different conjuncts. We prove, that this is not possible.

We proceed by induction on number of digits in n. We first prove, that if $n \in \varphi_{i,j,q}(\ldots, L_{i',j',q'}, \ldots)$ then $n = ijw$ for some word w and if n was obtained as a non-trivial sum then $|w| \geq 1$. Then we show that $\delta(q_0, w, q)$.

We begin with (31). Suppose that n belongs to the right-hand side after substituting $L_{i,j,q}, L_{i,j}$ for $A_{i,j,q}, A_{i,j}$. Consider the possible positions of the two first digits of each summand. Notice, that if j is on the position one to the right of i, then the two first digits are ij and there is at least one more digit. In this case we are done, so we deal only with other cases. The Table 1 summarizes the results, it has some drawbacks:

- some digits sum up to k or more and influence another digit by a carry,
- in the second column i may be or may be not on the same position as x, but we deal with those two cases together,

Table 1. The possible leading digits of numbers resulting from adding in (31).

	i and j are on the same position	j is leading	i is leading
$A_{i,0}A_{j,xq'}$	$(i+j), x$	$j, x\langle+i\rangle$	$i, 0$
$A_{i,1}A_{j-1,xq'}$	$(i+j-1), (x+1)$	$(j-1), x\langle+i\rangle$	$i, 1$
$A_{i,2}A_{j-2,xq'}$	$(i+j-2), (x+2)$	$(j-2), x\langle+i\rangle$	$i, 2$
$A_{i,3}A_{j-3,xq'}$	$(i+j-3), (x+3)$	$(j-3), x\langle+i\rangle$	$i, 3$

- in the second column there may be an add up to k somewhere to the right, and hence we can add 1 to x.

This possibilities in the second point were marked in the table by writing $\langle+i\rangle$.

For the intersection to be non-empty we have to choose four items from the Table 1, each in a different row. We show, that this is not possible. We say that some choices *fit*, if the digits included in the table are the same for those choices. As there are three columns, there have to be two fitting choices in one column.

No two elements in the third column fit. They have fixed leading digits and they clearly are different.

Suppose that two elements from the first column fit. We want to show that in both numbers the position of x is the same and therefore those numbers cannot be equal, as there can be no carry to the position of x and the digits on this position are different. If $i + j - z \geq k$ (perhaps by additional 1 carried from the previous position) then the first digit is 1. In the second element the first digit can be 1 (if there is a carry of 1) or at least $i + j - z'$, but the latter is not possible, since $i + j - z' > 1$. Hence either in both choices there is a carry of one to the leading position or in both choices there is no such carry. In both cases the position of x is the same in the numbers involved. Contradiction.

It is not possible to choose three fitting elements from the second column: as previously we may argue, that either in all of them in the leading digit (that is with $j - z$ for some z) there is an adding to k and a carry to the (newly created) position or in all of them there is no adding to k in the leading position. Suppose there is a carry in a leading position. Since $j < k$ then this is possible only for the first row. In particular it is not possible to have two choices when there is a carry in the leading position. Suppose that there is no carry from the leading position. Since there are three fitting choices, in one of them we must increase the value of the second digit by at least 2. But the maximal value carried from the previous position is 1, contradiction. As a consequence if there are two fitting choices from the second column then the first digit in the result is in range $(j-1, j)$.

And so if there are four fitting choices, then exactly one of them is in the first column, one in the third column and two in the middle column. The third column always begins with i. In the first column the leading digit is at least $i + j - 3 > i$ or it is 1. Hence $i = 1$. And so the choices in the second column begin with 1 as well. Hence $j < 3$, which is a contradiction.

Table 2. The possible leading digits of numbers resulting from adding in (32).

	$i-1$ and $k-z$ are on the same position	$k-z$ is leading	$i-1$ is leading
$A_{i-1,j+1}A_{k-1,x,q'}$	$(k+i-2),(1+j+x)$	$(k-1),x\langle+i\rangle$	$i-1,j+1$
$A_{i-1,j+2}A_{k-2,x,q'}$	$(k+i-3),(2+j+x)$	$(k-2),x\langle+i\rangle$	$i-1,j+2$
$A_{i-1,j+3}A_{k-3,x,q'}$	$(k+i-4),(3+j+x)$	$(k-3),x\langle+i\rangle$	$i-1,j+3$
$A_{i-1,j+4}A_{k-4,x,q'}$	$(k+i-5),(4+j+x)$	$(k-4),x\langle+i\rangle$	$i-1,j+4$

Table 3. The possible leading digits of numbers resulting from adding in (33).

	$k-z$ is leading	$j+z$ is leading
$A_{k-1,0}A_{j+1,x,q'}$	$(k-1),0\langle+j+1\rangle$	$(j+1),x\langle+k-1\rangle$
$A_{k-2,0}A_{j+2,x,q'}$	$(k-2),0\langle+j+2\rangle$	$(j+2),x\langle+k-2\rangle$
$A_{k-3,0}A_{j+3,x,q'}$	$(k-3),0\langle+j+3\rangle$	$(j+3),x\langle+k-3\rangle$
$A_{k-4,0}A_{j+4,x,q'}$	$(k-4),0\langle+j+4\rangle$	$(j+4),x\langle+k-4\rangle$

We move to (32). We again inspect the positions of the two leading digits in the summands on the right-hand side of the equation after substituting $L_{i'j',q'}$ and $L_{i',j'}$. If in at least one number the $j+m$ is on position of leading $k-m$ then the sum equals ijw for some w. Hence we consider only other arrangements of digits. The Table 2 summarizes the possible first two digits: as before we may argue, that if there are some fitting entries in some column then on their leading position digits sum up to k in all choices or in all choices they do not sum up to k.

We cannot have two choices from the third column (the second digits do not match). We can have at most two from the second column (to obtain three we would have to carry at least 2 to the first digit in one of them and this is not possible). For the same reason there can be at most two choices from the first column. But if there are two choices from the first column then we cannot match the positions with x. Hence there is at the most one choice from the first column.

And so we have one choice from the first column, one from the third and two from the second. Since the third and the second column match, then $i \geq k-3$. But in such a case in the first column we have at least $k+k-3-5 > k$, as $k > 8$. Hence the leading digit is 1. Contradiction.

Consider (33) and any number that can be a result of adding on the right-hand side after substituting $L_{i'j',q}$ and $L_{i',i'}$ for the variables. If in at least one case $j+m$ is on the same position as $k-m$, then the result begins with digits $1j$ and we are done. So we have to cope only with cases, when the leading digits are arranged in other way. The Table 3 summarizes the possible alignment of first two digits: in the first column there are no two fitting choices—since the second digit in this column is at most $j+4 < k$ and there is no carry to the first digit. And clearly the first digits are different. So we have to choose at least three elements from the second column. And this is not possible as the carry from the previous position to the leading position is at most 1 and we require 2 in such case. Contradiction.

We now take the indices denoting states of the automaton into our consideration. Consider equation (31) and some w belonging to the right-hand side. We have already proved that $w = ijw'$. If $w' = \epsilon$ then w is a part of $\{ij \mid q = q_0\}$, hence we deal with L_{i,j,q_0} and w is on the left-hand side as well. If $|w'| > 0$ then consider $A_{i,0}A_{j,x,q'}$ and $jxw'' \in L_{j,x,q'}$ that was used in derivation of w. Note that $w' = xw''$. By definition of $L_{j,x,q'}$ we obtain $\delta(q_0, w'', q')$ and by (31) we obtain $\delta(q', x, q)$, thus $\delta(q_0, xw'', q)$. Since $w = ijxw''$ therefore it belongs to the left-hand side.

Consider equation (32) and some w belonging to the right-hand side. We have already proved that $w = ijw'$. Consider $A_{i-1,j+1}A_{k-1,x,q'}$. If $w' = \epsilon$ then it is present on the right-hand side due to $\{ij \mid q = q_0\}$ and then ij is an element of L_{i,j,q_0} on the left-hand side. If $|w'| > 0$ then consider $(k-1)xw'' \in L_{k-1,x,q'}$ that was used in derivation of w. Note that $w' = xw''$. By definition of $L_{k-1,x,q'}$ we obtain $\delta(q_0, w'', q')$ and by (32) we obtain $\delta(q', x, q)$, thus $\delta(q_0, xw'', q)$. As $w = ijxw''$ it belongs to the left-hand side.

Consider equation (33) and some w belonging to the right-hand side. We have already proved that $w = 1jw'$. If $w' = \epsilon$ then it is on the right-hand side by $\{1j \mid q = q_0\}$ and is such case it is an element of L_{1,j,q_0} on the left-hand side. If $|w'| > 0$ consider $A_{k-1,0}A_{j+1,x,q'}$ and $(j+1)xw'' \in L_{j+1,x,q'}$ that was used in derivation of w. Note that $w' = xw''$. By definition of $L_{j+1,x,q'}$ we obtain $\delta(q_0, w'', q')$ and by (33) we obtain $\delta(q', x, q)$, and so $\delta(q_0, xw'', q)$. But $w = 1jxw''$, therefore it belongs to the left-hand side. □

We conclude with:

Theorem 19. *For every natural $k > 1$ and every regular $L \subseteq \Sigma_k^*$ language $f_k(L)$ is a conjunctive unary language.*

Proof. First note that language $L' = L \setminus 0\Sigma_k^*$ is regular as well and $f_k(L) = f_k(L')$, hence without losing generality we may assume that $L \subset \Sigma_k^* \setminus 0\Sigma_k^*$.

By Lemma 13 it is enough to consider $k > 8$.

Lemma 17 and Lemma 18 guarantee that the sets $(\ldots, L_{i,j,q_i}, \ldots)$ fulfil the assumptions of Lemma 2. Now the only thing left is to see that (34) defines S properly. If $w \in L$ then either $|w| = 1$ and hence $w \in S \cap \Sigma_k$ or $|w| \geq 2$ and hence $w = ijw'$. Let q be such that $\delta(q_0, w', q)$ and $\delta(q, ij) \in F$. Clearly such state exists, since $w \in L$ and so automata recognizing L has some intermediate state q. But this means that $w \in L_{i,j,q}$ and $S \supset L_{i,j,q}$. □

6. Conclusions and Open Problems

The main result of this paper is an example of a conjunctive grammar over unary alphabet generating non-regular language. This grammar has six nonterminal symbols in Chomsky normal form. Number of nonterminals could be reduced to three if we consider a grammar that is not in a Chomsky normal form. It remains an open question, how many nonterminals, intersection *etc.* are required to generate

a non-regular language. In particular, can we give natural sufficient conditions for a conjunctive grammar to generate a regular language? Also, no non-trivial algorithm for recognizing conjunctive languages over unary alphabet is known. An obvious modification of the CYK algorithm requires quadratic time and linear space. Can those bounds be lowered? Closure under complementation of conjunctive languages (both in general and and in case of unary alphabet) remains unknown, with conjectured negative answer.

The second important result is a generalization of the previous one: for every regular language $R \subseteq \{0, ..., k-1\}^*$ treated as set of base-k numbers language $\{a^n : \exists_{w \in R}\, w$ is a base-k notation of $n\}$ is a conjunctive unary language.

Acknowledgments

The author would like to thank T. Jurdziński and K. Loryś for introducing to the subject and helpful discussion, S. Bala and M. Bieńkowski for helpful discussion and A. Okhotin, who suggested the study of generalization to unary representations of all regular languages (Theorem 19).

Research was supported by MNiSW grant number N206 024 31/3826, 2006-2008.

Bibliography

1. A. Okhotin, Conjunctive grammars, *Journal of Automata, Languages and Combinatorics.* **6:4**(2001) 519–535.
2. A. Okhotin, An overview of conjunctive grammars. *Formal Language Theory Column. Bulletin of the EATCS.* **79**(2003) 145–163.
3. A. Okhotin, Boolean grammars, *Information and Computation.* **194:1**(2004) 19–48.
4. A. Okhotin, Nine open problems on conjunctive and boolean grammars. *TUCS Technical Report.* **794**(2006).
5. A. Okhotin, Conjunctive grammars and systems of language equations, *Programming and Computer Software.* **28**(2002) 243–249.