On Chomsky Hierarchy of Palindromic Languages*

Pál Dömösi[†], Szilárd Fazekas[‡], and Masami Ito[§]

Abstract

The characterization of the structure of palindromic regular and palindromic context-free languages is described by S. Horváth, J. Karhumäki, and J. Kleijn in 1987. In this paper alternative proofs are given for these characterizations.

Keywords: palindromic formal languages, combinatorics of words and languages

1 Introduction

The study of combinatorial properties of words is a well established field and its results show up in a variety of contexts in computer science and related disciplines. In particular, formal language theory has a rich connection with combinatorics on words, even at the most basic level. Consider, for example, the various pumping lemmata for the different language classes of the Chomsky hierarchy, where applicability of said lemmata boils down in most cases to showing that the resulting words, which are rich in repetitions, cannot be elements of a certain language. After repetitions, the most studied special words are arguably the palindromes. These are sequences, which are equal to their mirror image. Apart from their combinatorial appeal, palindromes come up frequently in the context of algorithms for DNA sequences or when studying string operations inspired by biological processes, e.g., hairpin completion [2], palindromic completion [10], pseudopalindromic completion [3], etc. Said string operations are often considered as language generating formalisms, either by applying them to all words in a given language or by applying them iteratively to words. One of the main questions, when considering the languages arising from these operations, is how they relate to the classes defined by the Chomsky hierarchy. In order to investigate that, one usually needs to refer

*The second author was supported by Akita University, Dept. of Information Science and Engineering

†Institute of Mathematics and Informatics, College of Nyíregyháza, H-4400 Nyíregyháza, Sóstói út 31/B, Hungary, E-mail: domosi@nyf.hu

‡Department of Information Science and Engineering, Akita University, Akita, Tegatagakuen City 1-1, 010-8502, Japan, E-mail: szilard.fazekas@gmail.com

§Department of Mathematics, Kyoto Sangyo University, Kyoto 603, Japan E-mail: ito@cc.kyoto-su.ac.jp

DOI: 10.14232/actacyb.22.3.2016.10

to the characterization of palindromic languages, i.e., languages in which all words are palindromes.

Characterization of palindromic regular and context-free languages was given in [7]. Regular palindromic languages have a simple characterization, which is the basis (essentially using the same idea) of the characterizations of pseudopalindromic and k-palindromic languages and the decidability results rooted in them [3].

In this paper we give alternative proofs of these characterizations. Due to the previously mentioned resurgence of interest in (pseudo-)palindromic languages, we think that it is important to have clear and, where possible, effective proofs for these results readily available. The paper by Horváth et al. is correct, and it conveys the main idea characterizing palindromic languages. However, the proofs omit several (tedious) details and explicit constructions. The latter and the fact that the availability of the paper is unfortunately rather limited, are the two main reasons which prompted us to write the present work. While our line of thought is similar to the original work of Horváth et al., we make use of results discovered since then (e.g. about bounded languages) to make the proofs simpler yet complete with details. We also present some explicit constructions in the proofs, which lead to a normal form of context-free grammars generating palindromic languages. As the proofs progress, we will point out differences between our work and the arguments in [7].

2 Preliminaries

A *word* (over Σ) is a finite sequence of elements of some finite non-empty set Σ. We call the set Σ an *alphabet,* the elements of Σ *letters.* If u and v are words over an alphabet Σ, then their *catenation* uv is also a word over Σ. In particular, for every word u over Σ, $u\lambda = \lambda u = u$, where λ denotes the *empty word.* Two words u, v are said to be *conjugates* if there exists a word w with $uw = wv$. For a word w, we define the powers of w inductively, $w^0 = \lambda$ and $w^n = w^{n-1}w$, where w^n is the n-th *power* of w. A nonempty word w is called *primitive* if it is not a power of another word, i.e., $w = v^k$ implies $v = w$ and $k = 1$. Otherwise we call it a *nonprimitive word.* Thus λ is also considered a nonprimitive word.

The *length* $|w|$ of a word w is the number of letters in w, where each letter is counted as many times as it occurs. Thus $|\lambda| = 0$. By the *free monoid* Σ^* *generated by* Σ we mean the set of all words (including the *empty word* λ) having catenation as multiplication. We set $\Sigma^+ = \Sigma^* \setminus \{\lambda\}$, where the subsemigroup Σ^+ of Σ^* is said to be the *free semigroup generated by* Σ. Subsets of Σ^* are referred to as *languages* over Σ. Denote by $|H|$ the *cardinality* of H for every set H. A language L is said to be *slender* if there exists a nonnegative integer c, such that for all integers $n \geq 0$ we have $|\{w \in L : |w| = n\}| \leq c$.

For a nonempty word $w = x_1 \cdots x_n$, where $x_1, \ldots, x_n \in \Sigma$, we denote its *reverse,* $x_n \cdots x_1$, by w^R. Moreover, by definition, let $\lambda = \lambda^R$, where λ denotes the empty word of Σ^*. We say that a word w is a *palindrome* (or *palindromic*) if $w = w^R$. Further, we call a language $L \subseteq \Sigma^*$ *palindromic* if all of its elements are palindromes.

A language $L \subseteq \Sigma^*$ is called a *paired loop language* if it is of the form $L = \{uv^n wx^n y | n \geq 0\}$ for some words $u, v, w, x, y \in \Sigma^*$.

Finally, as usual, we write a *generative grammar* G into the form $G = (V, \Sigma, S, P)$, where V and Σ are disjoint nonempty finite sets, the *set of nonterminals*, and *the set of terminals*, $S \in V$ is the *start symbol*, and $P \subset (V \cup \Sigma)^* VV \times (V \cup \Sigma)^*$ is the finite set of *derivation rules*. For every *sentential form* $W \in (V \cup \Sigma)^*$, $L_G(W)$ denotes the *language generated by* W, and $L(G)$ $(= L_G(S))$ denotes the language *generated by* G. Our results are related to well-known classes of the Chomsky hierarchy, that of context-free languages and regular languages. Apart from those two, we will use the notion of *linear grammars* (languages). For all three classes, $P \subset V \times \alpha$, where $\alpha = (V \cup \Sigma)^*$ for context-free grammars, $\alpha = \Sigma^*(V \cup \{\lambda\})\Sigma^*$ for linear grammars, and $\alpha = \Sigma^*(V \cup \{\lambda\})$ for regular grammars.

We shall use the following classical results.

Theorem 1. *[1] Let L be a regular language. Then there is a constant n such that if z is any word in L, and $|z| \geq n$, we may write $z = uvw$ in such a way that $|uv| \leq n, |v| \geq 1$, and for all $i \geq 0$, $uv^i w$ is in L. Furthermore, n is no greater than the number of states of the finite automaton with minimal states accepting L.*

Theorem 2. *The family of context-free languages is closed under the inverse homomorphism.*

Theorem 3. *[1] The language $L \subseteq \Sigma^*$ is context-free if and only if for every regular language $R \subseteq \Sigma^*$, $L \cap R$ is context-free.*

Theorem 4. *[6] Given an alphabet Σ, a nonempty word $w \in \Sigma^+$, each context-free language $L \subseteq w^*$ is regular having the form*

$$\cup_{i=1}^{k} w^{m_i}(w^{n_i})^* \text{ for some } m_1, n_1, \ldots, m_k, n_k \geq 0. \tag{1}$$

Theorem 5. *[8, 9, 12] Every slender context-free language is a finite disjoint union of paired loop languages.*

The following statement is well-known.

Proposition 1. *Given a context-free grammar $G = (V, \Sigma, S, P)$, a sentential form $W \in (V \cup \Sigma)^*$, the language $S_G(W)$ is also context-free.*

Theorem 6. *[13] Given a positive integer i, a pair $u, v \in \Sigma^+$, let $uv = p^i$ for some primitive word $p \in \Sigma^+$. Then $vu = q^i$ for a primitive word q.*

Theorem 7. *[11] If $uv = vq, u \in \Sigma^+, v, q \in \Sigma^*$, then $u = wz, v = (wz)^k w, q = zw$ for some $w \in \Sigma^*, z \in \Sigma^+$ and $k \geq 0$.*

Theorem 8. *[11] The words $u, v \in \Sigma^*$ are conjugates if and only if there are words $p, q \in \Sigma^*$ with $u = pq$ and $v = qp$.*

Theorem 9. *[4] Let $u, v \in \Sigma^*$. $u, v \in w^+$ for some $w \in \Sigma^+$ if and only if there are $i, j \geq 0$ so that u^i and v^j have a common prefix (suffix) of length $|u| + |v| - gcd(|u|, |v|)$.*

We shall use the following direct consequence of this result.

Theorem 10. *If two non-empty words p^i and q^j share a prefix of length $|p| + |q|$, then there exists a word r such that $p, q \in r^+$.*

3 Results

We start with alternative proofs of some results of S. Horváth, J. Karhumäki, J. Kleijn [7].

First we turn to consider regular languages. We present a proof which is shorter than the one in [7] and does not make direct reference to the underlying finite automata and is instead based solely on the pumping lemma for regular languages and combinatorial results. The following is a simple result, and essentially the same idea has been used for instance for the characterization of pseudopalindromic regular languages [3].

Theorem 11. *[7] A regular language $L \subseteq \Sigma^*$ is palindromic if and only if it is a union of finitely many languages of the form*

$$L_p = \{p\}, L_{q,r,s} = qr(sr)^* q^R, (p, q, r, s \in \Sigma^*), \tag{2}$$

where p, r and s are palindromes.

Proof. Clearly, any finite union of languages in (2) is both palindromic and regular. Conversely, let L be a palindromic regular language and n be the language-specific constant from Theorem 1. Naturally, there are finitely many words shorter than n, those will form the languages L_p. For any suitably long word $w \in L$, according to Theorem 1, we have a factorization $w = qvz$, with $0 < |qv| \leq n$ and $v \neq \lambda$, such that $qv^iz \in L$, for any $i \geq 0$. The two cases being symmetric, we may assume $|q| \leq |z|$, i.e., $z = xq^R$, for some $x \in \Sigma^*$, with v^ix being a palindrome. This gives us $x = r(v^R)^j$, for some r with $v^R = sr$ and some $j \geq 0$. But, for large enough i, v^ix ends in $sx = (v^Rv^R)^Rx = (r^Rs^R)^2r(v^R)^j$ and it starts with v^{j+2}, so we instantly get $v = r^Rs$ and thus $s = s^R$. It also follows, that $v^R = s^Rr$ and $v^R = s^Rr^R$, hence r is a palindrome, too. Then, our original word w can be written as $qr(sr)^{j+k}q^R$. A similar decomposition, according to Theorem 1 is bound to exist for all words longer than n. All parts of the decomposition, q, r and s are shorter than n, therefore there are finitely many triplets like this.

□

Next we prove the following simple observation.

Proposition 2. *Given a pair of positive integers i, j, let $p, r, u, w \in \Sigma^*, v \in \Sigma^+$ be arbitrary with $|p| \leq |u|, |r| \leq |w|$ and let $q \in \Sigma^+$ be a primitive word having $|v^j| \geq |v| + 3|q|$ such that $pq^ir = uv^jw$. Then there exists a positive integer k such that v and q^k conjugate.*

Proof. By our assumptions, there exists a pair of factorizations $u = pu'$, $w = v'q$ such that $q^i = u'v^jv'$. Because $|v^j| \geq |v| + 3|q|$, $|u'v'| = |q^i| - |v^j| \leq |q^i| - |v| - 3|q| < |q^{i-3}|$, there are a positive integer n, a suffix q_2 and a prefix q_3 of q such that $v^j = q_2q^nq_3$. Hence $v^j = q_2(q_1q_2)^nq_3 = (q_2q_1)^nq_2q_3$ for some decomposition $q = q_1q_2$ and prefix q_3 of q. By our conditions, $|v^j| - |q_3| \geq |v| + 3|q| - |q_3| \geq |v| + 2|q| > |v| + |q|$. Therefore, applying Theorem 10, we obtain $v, q_2q_1 \in z^+$ for some primitive word $z \in \Sigma^+$. By Theorem 6, q_2q_1 is also primitive. Therefore, $z = q_2q_1$. Hence $v = (q_2q_1)^k$ for some $k > 0$. Then Theorem 8 implies that v and q^k conjugate. $\square$

Now we continue with palindromic context-free languages. The line of thought is similar to the one in [7]. The main differences are as follows. The original proof of Theorem 12 is very succinct and only hints at the constructions needed to transform context-free grammars generating palindromic languages into linear grammars. We develop the result in detail. Afterwards, we show that for a linear grammar generating a palindromic language, one can find a "normal form", called palindromic grammar in [7]. Again, the original proof provides the combinatorial arguments to show that this is possible, but does not give an explicit construction. We present such a construction in the proofs of Lemmas 4 and 5. The technical details might at times be somewhat difficult to follow due to the proliferation of notation. To remedy that as much as possible, we decomposed the proofs in several lemmas.

Lemma 1. *Let $G = (V, \Sigma, S, P)$ be a context-free grammar, such that $L(G)$ is palindromic. Then, for any rule of the form $X \to pAqBr \in P$, with $p, q, r \in \Sigma^*$, $X, A, B \in V$, and $|L_G(A)| > 1$, $|L_G(B)| > 1$, we have that both $L_G(A)$ and $L_G(B)$ are slender context-free languages.*

Proof. Without loss of generality we can assume that V is reduced, i.e., for every $X \in V$, $L_G(X) \neq \emptyset$.

We will show that for every $q_1, q_2 \in \Sigma^*$, with $A \overset{*}{\underset{G}{\Rightarrow}} q_1, A \overset{*}{\underset{G}{\Rightarrow}} q_2$, we have that $q_1 \neq q_2$ implies $|q_1| \neq |q_2|$. Similarly, for every $r_1, r_2 \in \Sigma^*$, with $B \overset{*}{\underset{G}{\Rightarrow}} r_1, B \overset{*}{\underset{G}{\Rightarrow}} r_2$, we have $r_1 \neq r_2$ implies $|r_1| \neq |r_2|$. Because G is reduced, there are $u, y \in \Sigma^*$ having $S \overset{*}{\underset{G}{\Rightarrow}} uXy$. Therefore, $A \overset{*}{\underset{G}{\Rightarrow}} q_1$ and $A \overset{*}{\underset{G}{\Rightarrow}} q_2$ imply that for every $r' \in L_G(B)$, $upq_1qr'ry, upq_2qr'ry \in L(G)$, i.e., both of them are palindromes. This is impossible if $|q_1| = |q_2|$ with $q_1 \neq q_2$, unless $q_1 = xz_1x'$ and $q_2 = x''z_2x'''$, where z_1 and z_2 are palindromes and $upx = (x'qr'ry)^R, upx'' = (x'''qr'ry)^R$. However, then for any $r'' \in L_G(B)$ different from r', one of the words $upq_1qr''ry, upq_2qr''ry$ will not be a palindrome, but should be in $L(G)$, a contradiction.

Similarly, $B \overset{*}{\underset{G}{\Rightarrow}} r_1$ and $B \overset{*}{\underset{G}{\Rightarrow}} r_2$ imply that for every $q' \in L_G(A)$, we have $upq'qr_1ry$, $upq'qr_2ry \in L(G)$, i.e., both of them are palindromes. This is impossible if $|r_1| = |r_2|$ and $r_1 \neq r_2$, and $|L_G(A)| > 1$. This means, that both $L_G(A)$ and $L_G(B)$ are slender context-free. $\square$

Lemma 2. *Let L_1 and L_2 be paired loop languages. If L_1L_2 is palindromic, then L_1L_2 can be generated by a linear grammar.*

Proof. The words in L_1L_2 are of the form $u_1v_1^iw_1x_1^iu_2v_2^jw_2x_2^ju_3$ and we assume they are palindromes for any $i, j \geq 0$.

If one of the words v_1, x_1, v_2, x_2 is empty, then we can generate L_1L_2 with linear rules, e.g., if x_1 is empty then we can generate $u_1v_1^iw_1$, $i \geq 0$, by linear rules $X \to u_1A$, $A \to v_1A$, $A \to w_1u_2B$ and the rest of the word by linear rules $B \to Cu_3$, $C \to v_2Cx_2$, $C \to w_2$.

Therefore, if one of v_1, x_1, v_2, x_2 is empty then we are ready, so let us assume that none of them are λ.

W.l.o.g. we may assume that $|u_1| \geq |u_3|$. Choose $j \geq 2$ such that:

- $|x_2^ju_3| - |u_1| \leq 2|x_2|$,
- $|u_1v_1^2| \leq |x_2^ju_3|$ and
- $|v_2^j| \geq 2|v_1|$.

Choose i such that $|u_1v_1^i| \geq |u_2v_2^jw_2x_2^ju_3|$. As the word is a palindrome, this means that $(u_2v_2^jw_2x_2^ju_3)^Rt = u_1v_1^i$, for some possibly empty word t. By Theorem 9, we get that the primitive roots of v_1, v_2^R, x_2^R are all conjugates of some primitive word z and $(u_2v_2^jw_2x_2)^R$ is a factor of z^k, for large enough k. If we choose j and i such that $|v_2^ju_3| > |u_1v_1^iw_1x_1^i|$ and $|x_1^i| > 2|x_2|$, then again from Theorem 9, we get that the primitive root of x_1 is also a conjugate of z. Moreover, if we choose i such that either v_1 or x_1 is in the middle of the word, then we get that there exist some palindromes z_1, z_2 such that z_1z_2 is a conjugate of z. This means that for any i, j we have $u_1v_1^iw_1x_1^iu_2v_2^jw_2x_2^ju_3 \in u_3^R(z_1z_2)^+z_1u_3$. As $|v_1|$, $|x_1|$, $|v_2|$ and $|x_2|$ are all multiples of $|z_1z_2|$, we get that L can be generated by a linear grammar with derivation rules of the form $S \to u_3^Rz_1Xu_3$ and $X \to (z_2z_1)^{n_1}X$, $X \to (z_2z_1)^{n_2}X$, $X \to (z_2z_1)^m$, for some positive integers m, n_1, n_2, such that $n_1 \cdot |z| = |v_1x_1|$, $n_2 \cdot |z| = |v_2x_2|$ and $m \cdot |z| = |w_1| + |u_2| + |w_2| + (|u_1| - |u_3| - |z_1|)$. □

Theorem 12. *[7] Every palindromic context-free language is linear.*

Proof. Let $G = (V, \Sigma, S, P)$ be a context-free grammar generating the palindromic language L. Without loss of generality we can assume that V is reduced, i.e., for every $X \in V$, $L_G(X) \neq \emptyset$. In particular, we may assume for every $X \in V$, $|L_G(X)| = \infty$. Indeed, if $|L_G(X)| < \infty$, then we can eliminate the derivation rules

$$Y \to W_1XW_2X \cdots W_nXW_{n+1}, X \to W \in P,$$

$W, W_1, W_2, \ldots, W_{n+1} \in ((V \setminus \{X\}) \cup \Sigma)^*$ by new derivation rules of the form

$$Y \to W_1w_1W_2w_2 \cdots w_nW_{n+1}, w_1, \ldots, w_n \in L_G(X).$$

It can also be assumed that for every $X \to W \in P$, there are at most two (not necessarily different) nonterminals appearing in W. Indeed, if

$X \to u_1 A_1 \cdots u_n A_n u_{n+1} \in P$ with $X, A_1, \ldots, A_n \in V, u_1, \ldots, u_n \in \Sigma^*, n > 2$ then we can eliminate this derivation rule by the following new derivation rules using some new nonterminals $A'_1, \ldots, A'_{n-1}$:

$$X \to u_1 A_1 u_2 A'_2, A'_2 \to A_2 u_3 A'_3, \ldots, A'_{n-2} \to A_{n-2} u_{n-1} A'_{n-1}, A'_{n-1} \to A_{n-1} u_n.$$

Next we show that the derivation rules of the form $X \to pAqBr$ with $p, q, r \in \Sigma^*, A, B \in V$ can be eliminated.

Since we assumed $L_G(A)$ and $L_G(B)$ are infinite languages, by Lemma 1 both of them are slender context-free languages, hence so are $\{p\} \cdot L_G(A) \cdot \{q\}$ and $L_G(B) \cdot \{r\}$. Using Theorem 5, we get that $L_G(pAqBr)$ is a concatenation of two paired loop languages and it is palindromic. From here, applying Lemma 2 gives that $L_G(pAqBr)$ can be generated by linear derivation rules.

Thus we receive that $L(G)$ can be generated by a linear grammar. $\square$

Lemma 3. *Given an alphabet* Σ, *words* $v, z \in \Sigma^*$, *a non-empty word* $w \in \Sigma^+$, *each context-free language* $L \subseteq vw^*z$ *is regular having the form*

$$v(\cup_{i=1}^{k} w^{m_i}(w^{n_i})^*)z \text{ for some } m_1, n_1, \ldots, m_k, n_k \geq 0. \tag{3}$$

Proof. Let a, b, c distinct symbols and consider a homomorphism $\psi : \{a, b, c\} \to \Sigma^*$ with $\psi(a) = v, \psi(b) = w, \psi(c) = z$. Then $\psi^{-1}(L) \cap ab^*c = \{ab^k c \mid vw^k z \in L, k \geq 0\}$. On the other hand, using that ab^*c is obviously a regular language, Theorem 2 and Theorem 3 imply that $\psi^{-1}(L) \cap ab^*c$ is also context-free. Let $\psi' : \{a, b, c\} \to b^*$ be a homomorphism with $\psi'(a) = \psi'(c) = \lambda$ and $\psi'(b) = b$. By Theorem 2, $\psi'(\psi^{-1}(L) \cap ab^*c)$ is also context-free. On the other hand, $\psi'(\psi^{-1}(L) \cap ab^*c) = \{b^k \mid vw^k z \in L, k \geq 0\}$, therefore, by Theorem 4, it is regular which can be written into the form $\cup_{i=1}^{k} b^{m_i}(b^{n_i})^*$ for some $m_1, n_1, \ldots, m_k, n_k \geq 0$. This implies that L is regular having the form as in (3). $\square$

Given a grammar $G = (V, \Sigma, S, P)$, we say that a nonterminal $X \in V$ is *non-balanced* if there are $p, q \in \Sigma^*$ with $|p| \neq |q|$ such that $X \overset{*}{\underset{G}{\Rightarrow}} pXq$. Otherwise, we say that X is *balanced*. We will show that for each palindromic context-free language, there exists a linear grammar in a palindromic normal form. The proof requires two steps: first we show that such languages can be generated by grammars with balanced nonterminals, and then we show that any grammar with balanced nonterminals can be effectively transformed into a grammar in palindromic normal form.

Lemma 4. *Every palindromic context-free language can be generated by a* $G = (V, \Sigma, S, P)$, *such that each non-terminal in* V *is balanced.*

Proof. Consider an arbitrary palindromic context-free language L. By Theorem 12, we have that L is linear. Thus there exists a linear grammar $G = (V, \Sigma, S, P)$, such that $L(G) = L$. Without loss of generality, we may assume that G is reduced, moreover, $P \subseteq \{X \to aYb \mid X \in V, Y \in V \cup \{\lambda\}, a, b \in \Sigma \cup \{\lambda\}, ab \neq \lambda\}$. Indeed, if $X \to paYbq \in P$ with $p, q \in \Sigma^*, pq \in \Sigma^+, a, b \in \Sigma \cup \{\lambda\}, ab \neq \lambda, Y \in V \cup \{\lambda\}$, then we can eliminate the derivation rule $X \to paYbq \in P$ by introducing a new nonterminal symbol Z and the new derivation rules $X \to pZq, Z \to aYb$. Thus we get in finite-many steps that all derivation rules have the form $X \to aYb, X \in V, a, b \in \Sigma \cup \{\lambda\}, Y \in V \cup \{\lambda\}$.

Clearly, then

$$L = \cup\{\{p\}L_G(X)\{q\} \mid \ S \overset{*}{\underset{G}{\Rightarrow}} pXq,\ X \in V, p, q \in \Sigma^*, |p|, |q| \leq |V|\}. \tag{4}$$

Consider a non-balanced nonterminal X, as above. Let us assume X appears in a derivation at some point as $S \Rightarrow uXv$. Then, because $X \Rightarrow pXq$, we get $S \Rightarrow up^iXq^iv$, for all $i \geq 1$. Without loss of generality, we may assume $|u| \leq |v|$, that is, since the derived word will be a palindrome, $v = wu^R$, for some $w \in \Sigma^*$. Now, to keep arguments simple, let X stand for any word in $L_G(X)$. So, we know that p^iXq^iw is a palindrome for any positive i. For large enough i, this gives us that $w^R = p^jp_1$, for some $j \geq 0$ and $p_1 \in \Sigma^*$ prefix of p, hence $p^iXq^ip_1^R(p^R)^j$ is a palindrome. Again, if i was big enough for $|p^i| > |q^2p_1^R(p^R)^j|$, then by Theorem 9, we get that for a decomposition q_1q_2 of q^R, its conjugate q_2q_1 has the same primitive root as p, i.e., there exists some primitive word $z \in \Sigma^+$, $m, n \geq 1$, such that $q_2q_1 = z^m$ and $p = z^n$. Rewriting $p^iXq^ip_1^R(p^R)^j$ with these powers of z, we have $z^{ni}X(q_2^Rq_1^R)^ip_1(z^R)^{nj} = z^{ni}Xq_2^R(q_1^Rq_2^R)^{i-1}q_1^Rp_1(z^R)^{nj} = z^{ni}Xq_2^R(z^R)^{m(i-1)}q_1^Rp_1(z^R)^{nj}$ is a palindrome, therefore $z^{n(i-j)}Xq_2^R(z^R)^{m(i-1)}q_1^Rp_1$ is, as well. This means $p_1^Rq_1z^2$ is a prefix of $z^{n(i-j)}$, and we can apply Theorem 9 again to get that, since z is primitive, $p_1^Rq_1 = z^k$, for some integer k. Since p_1^R is a suffix of $p^R = (z^R)^n$ and q_1 is a suffix of z^m, there exist non-negative integers i_1, i_2 and z'_r suffix of z^R, z' suffix of z, such that $z'_r(z^R)^{i_1}z'z^{i_2} = z^k$. From here, there is some prefix z''_r of z^R, with $z''_rz'_r = z^R$, $z'_rz''_r = z$, so both z''_r and z'_r are palindromes and so are $p_1 = z'_r(z''_rz'_r)^{i_1}$ and $q_1 = (z''_rz'_r)^{k-i_1-1}z''_r$. But $q_2q_1 = z^m = (z'_rz''_r)^m$, so $q_2 = z'_r(z''_rz'_r)^{m-k+i_1+1}$. From here, $z^{ni}X(q_2^Rq_1^R)^ip_1(z^R)^{nj} = (z'_rz''_r)^{ni}X(z'_rz''_r)^{mi}z'_r(z''_rz'_r)^{i_1}(z''_rz'_r)^{nj} = (z'_rz''_r)^{ni}X(z'_rz''_r)^{mi+i_1+nj}z'_r$ is a palindrome for all $i \geq 1$. As our original assumption was $|p| \neq |q|$, i.e., $m \neq n$, for a large enough i, the word X will be entirely to the left or right from the center of a palindrome of the form $(z'_rz''_r)^{j_1}X(z'_rz''_r)^{j_2}z'_r$. Since $z'_rz''_r$ is primitive, the center of the palindrome has to be exactly z'_r or z''_r, and this means that $X \in (z'_rz''_r)^+$. Then, the language $L_G(X)$ is isomorphic to a unary context-free language, hence it is regular with rules of the form $X \to (z'_rz''_r)^{m+n}X$. This way, in our original grammar we can replace all rules with X on the left with balanced rules $X \to (z'_rz''_r)^{\frac{m+n}{2}}X(z'_rz''_r)^{\frac{m+n}{2}}$ and $X \to \lambda$, or if $m+n$ is odd, with rules $X \to (z'_rz''_r)^{m+n}X(z'_rz''_r)^{m+n}$ and $X \to (z'_rz''_r)^{m+n}|\lambda$.

□

Lemma 5. *Every palindromic context-free language can be generated by a grammar* $G = (V, \Sigma, S, P)$ *having* $P \subseteq \{X \to aYa \mid X, Y \in V, a \in \Sigma\} \cup \{X \to a \mid X \in V, a \in \Sigma\} \cup \{X \to \lambda\}$.

Proof. Now we may assume that V contains only balanced nonterminals, i.e., for every derivation, $X \overset{*}{\underset{G}{\Rightarrow}} uXx$, where $X \in V$, $u, x \in \Sigma^*$, $|u| = |x|$. Then, for every

$X \in V$, $p, q \in \Sigma^*$, $S \overset{*}{\underset{G}{\Rightarrow}} pXq$ implies $||p| - |q|| < |V|$. This obviously holds for derivations of less than $|V|$ steps, as in each step we add at most one letter to either side. Assume the contrary for a longer derivation:

$$X_0 \underset{G}{\Rightarrow} x_1 X_1 y_1 \underset{G}{\Rightarrow} \cdots \underset{G}{\Rightarrow} x_{n-1} X_{n-1} y_{n-1} \cdots y_1 \underset{G}{\Rightarrow} x_1 \cdots x_n X_n y_n \cdots y_1, \tag{5}$$

where $X_0 = S$, $x_1, \ldots, x_n, y_1, \ldots, y_n \in \Sigma \cup \{\lambda\}$ and $n > |V|$. Then, there exist $0 \leq i < j \leq n$, such that $X_i = X_j$, but X_i is balanced, so $|x_i \cdots x_j| = |y_j \cdots y_i|$, therefore we can remove them from both sides and get that $||x_1 \cdots x_n| - |y_n \cdots y_1|| = ||x_1 \cdots x_{i-1} x_{j+1} \cdots x_n| - |y_n \cdots y_{j+1} y_{j-1} \cdots y_{i+1}||$. Repeating this until we get a derivation with at most $|V|$ steps, gives us $||x_1 \cdots x_n| - |y_n \cdots y_1|| \leq |V|$.

Now, to every derivation, we assign two queues (first-in-first-out storages), called *left store* and *right store*. Either both of them are empty, or one of them is empty and the other one contains a non-empty terminal string of length less than $|V|$.

At the start, both stores are empty. This status does not change as long as the applied derivation rules are of the form $X \to aYa$, $X, Y \in V, a \in \Sigma \cup \{\lambda\}$. If the applied derivation rule has the form $X \to aY, X, Y \in V, a \in \Sigma$, then there are two cases: if the left store is empty, then we drop the terminal letter a onto the top of the right store; otherwise we delete the terminal letter contained at the bottom of the left store. In the second case, the bottom of the left store should contain the same terminal letter a. Otherwise the generated word will not be a palindrome. Similarly, if the applied derivation rule has the form $X \to Yb, X, Y \in V, b \in \Sigma$, then we have two cases: if the right store is empty, then we drop the terminal letter b onto the top of the left store; otherwise we delete the terminal letter contained at the bottom of the right store. In the second case again, the bottom of the right store should contain the same terminal letter b. Otherwise the generated word will not be a palindrome.

If the applied derivation rule has the form $X \to aYb, X, Y \in V, a, b \in \Sigma$, then we have the following possibilities: if one of the stores is not empty, then our procedure works as in the previous cases (like, in order, applying a derivation rule $X \to aZ, a \in \Sigma, X, Z \in V$, and then a derivation rule $Z \to Yb, b \in \Sigma, Z, Y \in V$); if both stores are empty then $a = b$ should hold, otherwise the generated string will not be a palindrome. After applying the considered derivation rule $X \to aYb, X, Y \in V, a, b \in \Sigma$, the contents of the stores remain the same.

We will construct our grammar such that a derivation rule of the form $X \to a, a \in \Sigma \cup \{\lambda\}, X \in V$ can be applied only if either one of the stores contains the letter a or both stores are empty.

In addition, if both stores are empty, and $X \overset{*}{\underset{G}{\Rightarrow}} w$ may hold for the nonterminal X contained on the left-hand side of the applied derivation rule, then w should be a palindrome. In addition, if $|w| < |V|$, then either $w = b$ with $b \in \Sigma \cup \{\lambda\}$, or $w = c_1 \cdots c_t d c_t \cdots c_1$ for some $c_1, \ldots, c_t \in \Sigma, d \in \Sigma \cup \{\lambda\}, 1 \leq t < |V|$. For the second case, we assume the existence of some derivation rules of the form $X \to c_1 Z_1 c_1, Z_1 \to c_2 Z_2 c_2, \ldots, Z_{t-1} \to c_t Z_t c_t, Z_t \to d, Z_1, \ldots, Z_t \in V$.

Having these properties, we formally define the following set of derivation rules, where the (new) nonterminals are supplied by the queues discussed above.

Let $\bar{V} = \{X \in V \mid X \overset{*}{\underset{G}{\Rightarrow}} w, w \in \Sigma^+, |w| < |V|\}$ and define, in order,
$V' = \{X_{\lambda,\lambda} \mid X \in V\} \cup \{X_{a_1\cdots a_k,\lambda} \mid X \in V, a_1, \ldots, a_k \in \Sigma, k < |V|\} \cup \{X_{\lambda,b_1\cdots b_k} \mid X \in V, b_1, \ldots, b_k \in \Sigma, k < |V|\}$
and
$P' = \{X_{a_1\cdots a_k,\lambda} \to aY_{a_1\cdots a_k a,\lambda}a, X_{\lambda,a_1\cdots a_k} \to Y_{\lambda,a_1\cdots a_{k-1}}, X_{\lambda,\lambda} \to aY_{a,\lambda}a \mid X \to Ya \in P, X, Y \in V, a_1, \ldots, a_k, a \in \Sigma, k < |V|\} \cup \{X_{a_1\cdots a_k,\lambda} \to Y_{a_1\cdots a_{k-1},\lambda}, X_{\lambda,a_1\cdots a_k} \to aY_{\lambda,a_1\cdots a_k a}a, X_{\lambda,\lambda} \to aY_{\lambda,a}a \mid X \to aY \in P, X, Y \in V, a_1, \ldots, a_k, a \in \Sigma, k < |V|\} \cup \{X_{a_1\cdots a_k,\lambda} \to bY_{a_1\cdots a_{k-1}b,\lambda}b, X_{\lambda,a_1\cdots a_k} \to aY_{\lambda,a_1\cdots a_{k-1}a}a, X_{\lambda,\lambda} \to aY_{\lambda,\lambda}b \mid X \to aYb \in P, X, Y \in V, a_1, \ldots, a_k, a, b \in \Sigma \cup \{\lambda\}\} \cup \{X_{a_1\cdots a_k,\lambda} \to Y_{a_1\cdots a_k,\lambda}, X_{\lambda,a_1\cdots a_k} \to Y_{\lambda,a_1\cdots a_k}, X_{\lambda,\lambda} \to Y_{\lambda,\lambda} \mid X \to Y \in P, X, Y \in V, a_1, \ldots, a_k, \in \Sigma \cup \{\lambda\}\} \cup \{X_{a,\lambda} \to \lambda, X_{\lambda,a} \to \lambda, X_{\lambda,\lambda} \to a \mid X \to a \in P, X \in V, a \in \Sigma\} \cup \{X_{\lambda,\lambda} \to \lambda \mid X \to \lambda \in P\} \cup \{X_{\lambda,\lambda} \to c_1 Z_{1_{X\,\lambda,\lambda}} c_1, Z_{1_{X\,\lambda,\lambda}} \to c_2 Z_{2_{X\,\lambda,\lambda}} c_2, \ldots, Z_{t-1_{X\,\lambda,\lambda}} \to c_t Z_{t_{X\,\lambda,\lambda}} c_t, Z_{t_{X\,\lambda,\lambda}} \to d \mid X \in \bar{V}, X \overset{*}{\underset{G}{\Rightarrow}} c_1 \cdots c_t d c_t \cdots c_1, c_1, \ldots, c_t \in \Sigma, d \in \Sigma \cup \{\lambda\}\}$.

Thus we get that $L(G) = L(G')$, where $G' = (V', \Sigma, S_{\lambda,\lambda}, P')$, and G' has the desired form. □

Theorem 13. *[7] A context-free language $L \subseteq \Sigma^*$ is palindromic if and only if it is a disjoint union of $|V|$ languages of the form $\{pap^R \mid p \in L_a\}$, where the L_a $(a \in \Sigma \cup \{\lambda\})$ are regular languages (uniquely determined by L).*

Proof. Given an alphabet Σ, for every $a \in \Sigma \cup \{\lambda\}$ consider a regular language L_a. It is clear that $L = \bigcup_{a \in \Sigma \cup \{\lambda\}} \{pap^R : p \in L_a\}$ is palindromic and linear (and thus, it is also context-free). Conversely, consider a palindromic context-free language L. By Lemma 5, it can be generated by a grammar $G = (V, \Sigma, S, P)$ having $P \subseteq \{X \to aYa \mid X, Y \in V, a \in \Sigma\} \cup \{X \to a \mid X \in V, a \in \Sigma\} \cup \{X \to \lambda \mid X \in \Sigma\}$. For every $a \in \Sigma \cup \{\lambda\}$, define the grammar $G_a = (V, \Sigma, S, P_a)$ with $P_a = P \setminus \{X \to b \mid b \in \Sigma \cup \{\lambda\}, b \neq a\}$). Obviously, $L(G) = \cup_{a \in \Sigma} L(G_a)$. Moreover. for every $a, b \in \Sigma \cup \{\lambda\}$, $L(G_a) \cap L(G_b) \neq \emptyset$ if and only if $a = b$. Therefore, L is a disjoint union of the languages $L(G_a), a \in \Sigma \cup \{\lambda\}$. By the construction of $G_a, a \in \Sigma \cup \{\lambda\}$, it is clear that $G_{a,\ell} = (V, \Sigma, S, P_{a,\ell}$ with $P_{a,\ell} = \{X \to Yb \mid X \to bYb \in P_a, X, Y \in V, a \in \Sigma\} \cup \{X \to b \mid X \to b \in P_a, X \in V, a \in \Sigma \cup \{\lambda\}\}$ is a regular language. Similarly, $G_{a,r} = (V, \Sigma, S, P_{a,r}$ with $P_{a,r} = \{X \to bY \mid X \to bYb \in P_a, X, Y \in V, a \in \Sigma\} \cup \{X \to b \mid X \to b \in P_a, X \in V, a \in \Sigma \cup \{\lambda\}\}$ is regular. Moreover, $L_a = L(G_{a,\ell}) = L(G_{a,r})$, and $L = \bigcup_{a \in \Sigma \cup \{\lambda\}} \{pap^R : p \in L_a\}$. □

Finally, for the sake of completeness, let us make an easy observation. Every palindromic context-sensitive (phrase-structured) language has the form

$$L = \bigcup_{a \in \Sigma \cup \{\lambda\}} \{pap^R : p \in L(a)\},$$

where the $L(a)$ $(a \in \Sigma \cup \{\lambda\})$ are context-sensitive (phrase-structured) languages (uniquely determined by L).

References

[1] Bar-Hillel, Y.; Perles, M.; Shamir, E.: On formal properties of simple phrase structure grammars. Zeitschrift für Phonetik, Sprachwuissenschaft, und Kommunikationsforschung, **14** (1961), 143-177.

[2] Cheptea, D; Martín-Vide, C.; Mitrana, V.: A new operation on words suggested by DNA biochemistry: Hairpin completion. *In Proc. Conf. Transgressive Computing*, 2006, 216-228.

[3] Fazekas, S.Z.; Manea, F.; Mercas, R.; Shikishima-Tsuji, K.: The pseudopalindromic completion of regular languages. Inform. Comput. **239** (2014), 222-236.

[4] Fine, N. J.; Wilf, H. S.: Uniqueness theorems for periodic functions. Proc. Am. Math. Soc. **16** (1965), 109-114.

[5] Ginsburg, S.; Spanier, E. H.: Bounded ALGOL-like languages. *Trans. Am. Math. Soc.*, **113** (1964), 333-368.

[6] Ginsburg, S.; Rice, H. G.: Two families of languages related to ALGOL. *J. Assoc. Computing Machinery*, **9** (1962), 350–371.

[7] Horváth, S.; Karhumäki, J.; Kleijn, J.: Results concerning palindromicity. (Mathematical aspects of informatics, Mägdesprung, 1986). *J. Inform. Process. Cybernet.* **23** (1987), no. 8-9, 441–451.

[8] Ilie, L.: On a conjecture about slender context-free languages. *Theoret. Comput. Sci.,* **132** (1994), 427–434.

[9] Latteux, M; Thierrin, G.: Semidiscrete context-free languages. *Internat. J. Comput. Math.* **14** (1983), 3–18.

[10] de Luca, A; Luca, A.D.: Pseudopalindrome closure operators in free monoids. *Theoret. Comput. Sci.,* **362** (2006), 282-300.

[11] Lyndon, R. C.; Schützenberger, M. P.: The equation $a^m = b^n c^p$ in a free group. *Michigan Math. J.*, **9** (1962), 289-298.

[12] Raz, D.: Length considerations in context-free languages. *Theoret. Comput. Sci.*, **183** (1997), 21–32.

[13] Shyr, H. J.; Thierrin, G.: Disjunctive languages and codes. *In: Karpiński (ed.): Proc. Conf. FCT'77*, **56** (1977), Springer-Verlag, 171–176.

Received 11th October 2013