数理解析研究所講究録
第 1809 巻 2012 年 128-139

Palindromic completion: a new operation*

Szilárd Zsolt Fazekas[1], Robert Mercaş[2], and
辻 佳代子 (Kayoko Shikishima-Tsuji)[3]

[1] Department of Mathematics, Kyoto Sangyo University,
Motoyama, Kamigamo, Kita-Ku Kyoto 603-8555, Japan,
szilard.fazekas@gmail.com
[2] Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik,
PSF 4120,D-39016 Magdeburg, Germany,
robertmercas@gmail.com
[3] 天理大学 (Tenri University), 1050 Somanouchi Tenri 632-8510, Japan,
tsuji@sta.tenri-u.ac.jp

Abstract. Palindromes are words that read from either the left or the right end are the same. In this paper we introduce a new word operation, that of palindromic completion, in which symbols are added to either side of the word such that the new obtained words are palindromes. This notion represents somehow a particular type of hairpin completion, where the length of the hairpin is at most one. We give precise descriptions of regular languages that are closed under this operation and show that in this setting the regularity of the closure under the operation is decidable.
Keywords: Palindromes, palindromic completion, palindromic iterated completion, regular languages, decidability.

1 Introduction and preliminaries

Palindromes are sequences which read the same starting from either end. Besides their importance in combinatorial studies of strings, mirrored complementary sequences occur frequently in DNA and are often found at functionally interesting locations such as replication origins or operator sites. Already in the 1950's it was recognized that palindromic regions of DNA can exist in a cruciform structure with intrastrand base pairing of the self-complementary sequence, i.e., if a palindromic sequence occurs in a double strand, then pulling apart the two strands at the middle of the palindrome one can perform a "transfer-twist" in which each strand twists about itself, reducing the energy needed to separate the strands. A similar phenomenon is when a single strand of DNA curls back on itself to become self-complementary, after which a polymerase chain reaction extends the "shorter" end to generate a complete double strand, the result being a partial double helix with a bend in it. The structure is called a hairpin

* The work of Szilárd Zsolt Fazekas was supported by the *Japanese Society for the Promotion of Science* under number *P10827* and the work of Robert Mercaş was supported by the *Alexander von Humboldt Foundation*

or stem-loop, and it is an important building block of many RNA secondary structures.

The latter was the main motivation for the study of the mathematical hairpin concept considered to be a word in which some suffix is matching, Watson-Crick complementary to, a middle factor of the word. We exhibit a particularisation of the hairpin concept: whenever designating the complement of some symbol to be the symbol itself, palindromes can be seen as hairpins for which the loop has size at most one (we consider that whenever a matching is possible, the binding happens). Note that in the biological phenomenon serving as inspiration for the model, the length of the hairpin in the case of stable bindings is also limited (approximatively 4-8 base-pairs exist normally in such situations).

We say that whenever a word has a palindrome as prefix or suffix, then it is extended with the reverse of the rest of the word. A concept somehow similar, that of palindromic closure was introduced in [1] where the author only considers the shortest left or right palindrome constructed in such a fashion. To illustrate this with an example, after palindromic completion from the word $abbb$ we get both $abbba$ and $abbbba$, depending on which of the suffixes bb or bbb we choose for the binding loop. Furthermore, we consider iterated palindromic completion, the successive application of palindromic completion, taken to the limit. Under these conditions we prove that one can obtain precise characterizations of both words and regular languages whose iterated palindromic completion is regular. Moreover, we show that in this setting the problem of whether the iterated completion of some language is regular or not is decidable.

We assume the reader to be familiar with fundamental concepts as alphabet, word, language and regular expression (for more details see [2]) and end this Section with definitions regarding combinatorics on words and formal languages.

The length of a finite word w is the number of not necessarily distinct symbols it consists of and is written $|w|$. The number of occurrences of a certain letter a in w is $|w|_a$. The i-th symbol we denote by $w[i]$ and use the notation $w[i \dots j]$ to refer to the part of a word starting at the i-th and ending at the j-th position.

Words together with the operation of concatenation form a free monoid, which is usually denoted by Σ^* for an alphabet Σ. Repeated concatenation of a word w with itself is denoted by w^i for natural numbers i.

A word u is a *prefix* of w if there exists an $i \leq |w|$ such that $u = w[1 \dots i]$. We denote this by $u \leq_p w$. If $i < |w|$, then the prefix is called *proper*. Suffixes are the corresponding concept reading from the back of the word to the front. A word w has a positive integer k as a *period* if for all i, j such that $i \equiv j (\mathrm{mod} k)$ we have $w[i] = w[j]$, whenever both $w[i]$ and $w[j]$ are defined.

The central concept to the work presented here is *palindromicity*. First off, for a word w by w^R we denote its reversal, that is $w[|w| \dots 1]$. If $w = w^R$, the word is called a palindrome; for words of even length we have $w = uu^R$, while for odd length we have $w = uau^R$ with u a word and a some letter at their centre. The set of all palindromes of a language L is denoted by $\mathcal{P}al(L) = \mathcal{P}al \cap L$. If $\mathcal{P}al(L) = L$, the language L is called a palindromic.

Horváth, Karhumäki and Kleijn [3] have characterized the regular languages consisting only of palindromes:

Theorem 1. *A regular language $L \subseteq \Sigma^*$ is palindromic, if and only if it is a union of finitely many languages of the form $L_p = \{p\}$ or $L_{r,s,q} = qr(sr)^*q^R$ where p, r and s are palindromes, and q is an arbitrary word.*

We note here that the location of $\mathcal{P}al$ – the language of all palindromes – in the Chomsky Hierarchy is well-known; it is linear context-free. Another fact worth noting is that the primitive root of every palindrome is again a palindrome.

Trivially, every palindrome $p = aqa$, with q a (possibly empty) palindrome, has palindromic prefixes λ, a and aqa, therefore whenever we say a palindrome has a non-trivial palindromic prefix (suffix), we mean that it has a proper prefix (suffix) of length at least two which is a palindrome.

Definition 1. *For a word uv, where $v \notin \Sigma \cup \{\lambda\}$ (respectively, $u \notin \Sigma \cup \{\lambda\}$) is a palindrome, uvu^R (respectively, $v^R uv$) is in the right(left)-*palindromic completion *of uv. We say that w' is in the palindromic completion of w if it is either in the right or left palindromic completion of w. We denote this relation by $w \vdash w'$. The reflexive, transitive closure of $\vdash$ is the* iterated palindromic completion, *in notation $\vdash^*$, where for two words w and w' we say $w \vdash^* w'$ if $w = w'$ or there exist words $v_1, \ldots, v_n$ with $v_1 = w$, $v_n = w'$ and $v_i \vdash v_{i+1}$ for $1 \leq i \leq n-1$.*

Definition 2. *For a language L, we denote $L = L_0$ and for any $n > 0$ we let L_n be the palindromic completion of L_{n-1}, i.e., $L_n = \{w \mid \exists u \in L_{n-1} : u \vdash w\} \cup L$. For a language L, we denote by L_∞ the iterated palindromic completion of L, in other words $L_\infty = \bigcup_{n \to \infty} L_n$.*

For a singleton language $L = \{w\}$, let $\{w\}_n$ denote L_n, i.e., the nth palindromic completion of the word w.

2 Palindromic Completion

A trivial observation that we make is that the palindromic completion of a word is always a finite set, given that it has finitely many palindromic pre- or suffixes.

To see that the class of regular languages is not closed under palindromic completion, consider the language $L = aa^+ba$. After one palindromic completion step we get $L_1 = ab \cdot L \cup \{a^n ba^n \mid a \geq 2\}$, which, intersected with the regular language a^*ba^*, results in the non-regular context-free language $\{a^n ba^n \mid n \geq 2\}$.

As we will see, even when the starting language is a very simple one, its iterated palindromic completion can become highly complex. We shall characterize the regular languages, which have regular closure under this operation, but first let us treat the closure of words. The following lemma is quite useful later on:

Lemma 1. *For a palindrome w starting with a palindrome v of length greater than one, there always exist palindromes $x \neq \lambda$ and y, such that $v, w \in x(yx)^*$.*

Proof. Assume we have $w = vu$, such that v is a palindrome, the other case being similar. If $2|v| < |w|$, since v is a prefix of the palindrome w, and v is a palindrome itself, taking $v = x$ the conclusion follows.

It is well known (see [4]), that if a word has a border longer than half of the word, then it has a border shorter than half. Any border of a palindrome is a palindrome itself and this leads back to the previous case. □

The next immediate result says that, since each palindromic prefix of a word is also a suffix of it, the left and right palindromic completions are symmetric.

Lemma 2. *For palindromes, the right palindromic completion equals the left palindromic completion.*

Thus, whenever considering several steps of palindromic completion for some language L, it is enough to consider either the right, or the left, palindromic completion of L_1.

3 Iterated Palindromic Completion

Next natural question considers the iterated palindromic completion? Without loss of generality, we assume that all languages investigated in the case of iterated completion have only words longer than two. The case of palindromic completion on unary alphabets is not difficult to prove. Even more, we have that even for arbitrary unary languages the iterated palindromic completion is regular:

Proposition 1. *The class of unary regular languages is closed under palindromic completion. Furthermore, the iterated palindromic completion of any unary language is regular.*

Proof. We know that all unary regular languages can be expressed as a finite union of languages of the form $\{a^k(a^n)^* \mid k, n \text{ are some non-negative integers}\}$. Hence, since after a one step palindromic completion of each word a^m we get the language $\{a^\ell \mid \ell < 2m\}$, the first part of our result follows. Moreover, for some arbitrary unary language, after the iterated completion we get the language $\{a^j a^* \mid j \text{ is the minimum integer among all } m\text{'s } \}$. □

Next let us investigate what happens in the singleton languages case.

Proposition 2. *The class of iterated palindromic completion of singletons is incomparable with the class of regular languages.*

Proof. To show that regular languages are obtained take the word a^2ba. It is not difficult to check that the language obtained is $\{a^2ba\} \cup \{ab(a^2b)^n a \mid n \geq 1\} \cup \{a^2(ba^2)^n \mid n \geq 1\}$. Since all languages are regular, so is their union.

To see now that we not always get regular, nay, non-context-free, consider the word $u = a^3ba$. After one step completion we get the words $aba^3ba, aba^4ba, a^3ba^3$. The iterated palindromic completion of the first two is $\{aba(a^3ba)^n \mid n \geq 1\} \cup$

$\{aba(a^2ba)^n \mid n \geq 1\}$. However, the one step completion of the word a^3ba^3 gives $\{a^3ba^3ba^3, a^3ba^4ba^3\}$. From $\{a^3b(a^4b)^n a^3 \mid n \geq 1\}$ one can always get $\{a^3b(a^4b)^m a^3 \mid 1 < n+1 \leq m \leq 2n-1\}$, and $\{a^3b(a^4b)^n a^3b(a^4b)^n a^3 \mid n \geq 1\}$. From the latter, the completion includes $L = \{a^3b(a^4b)^n a^3b(a^4b)^m a^3b(a^4b)^n a^3 \mid 1 \leq n \leq m \leq 2n+1\}$. In particular, $\{u\}_\infty \cap a^3(b(a^4b)^+a^3)^3$ gives us exactly L, which is easily shown not to be context-free by applying the pumping lemma. □

Proposition 3. *For all words of the form $w = up(qp)^n u^R$, where p and q are palindromes and u is a suffix of pq, there exist palindromes p', q' such that $w = p'(q'p')^m$ with $n \leq m \leq n+2$.*

Proof. Depending on the lengths of u and q we distinguish the following cases:
1. $|u| \leq \frac{|q|}{2}$ - in this case $q = u^R xu$, for some (possibly empty) palindrome x. Thus, w can be written as $up(u^R xup)^n u^R = upu^R(x.upu^R)^n$, therefore by assigning $p' = upu^R$ and $q' = x$ we can conclude the proof.
2. $\frac{|q|}{2} < |u| \leq |q|$ - in this case the prefix u and the suffix u^R overlap in q, i.e., $q = xyxyx$ for some palindromes x and y, where $u = xyx$. Thus, $w = xyxp(xyxyxp)^n xyx = x(yxpxy.x)^{n+1}$ and we find that $p' = x$ and $q' = yxpxy$ satisfy our requirements.
3. $|q| < |u|$ - in this case $u = xq$ for some suffix x of p. Thus, $w = xqp(qp)^n qx^R = xq(pq)^{n+1}x^R$ with x a suffix of p, which brings us back to cases 1 or 2 (if the latter, the exponent increases by one yet again). □

The following result tells us that whenever a palindrome w has u as a palindromic prefix it has $|w| - |u|$ as period:

Lemma 3. *[1] A palindrome w has period $p < |w|$ if and only if it has a palindromic prefix of length $|w| - p$.*

Proposition 4. *Let $u_i p_i (q_i p_i)^{k_i} u_i^R$ be a sequence of palindromes with*

$$u_i p_i (q_i p_i)^{k_i} u_i^R \vdash u_{i+1} p_{i+1} (q_{i+1} p_{i+1})^{k_{i+1}} u_{i+1}^R,$$

where $1 \leq i \leq n$, and $u_1 = u_n$, $p_1 = p_n$ and $q_1 = q_n$. Then, for all i with $1 \leq i \leq n$ there exist palindromes p, q and positive integers t_i, such that

$$u_i p_i (q_i p_i)_i^k u_i^R = p(qp)^{t_i}.$$

Proof. Since $w \vdash w'$ implies $w \leq_p w'$, we get $u_1^R \leq_p (q_1 p_1)^{k_n - k_1}$. Then, there exist words u, v with $uv = q_1 p_1$ and some $t \geq 0$, such that we can write $u_1^R = (q_1 p_1)^t u$, hence $u_1 = u^R (p_1 q_1)^t$. But, $p_1 q_1 = p_1^R q_1^R = (q_1 p_1)^R = (uv)^R = v^R u^R$, therefore $u_1 p_1 (q_1 p_1)^{k_1} u_1^R = u^R (v^R u^R)^t (v^R u^R)^{k_1} p_1 (q_1 p_1)^t u = u^R (p_1 q_1)^{2t+k_1} p_1 u$ and $u_1 p_1 (q_1 p_1)^{k_n} u_1^R = u^R (p_1 q_1)^{2t+k_n} p_1 u$. Taking this further gives us that for every i with $1 \leq i \leq n$ there exists a $t_i > 0$ and a suffix x_i of $p_i q_i$ such that $x_i p_i (q_i p_i)^{k_i} x_i^R \vdash^* x_i p_i (q_i p_i)^{k_i + t_i} x_i^R$. Now we can apply Proposition 3, which gives us that these are all words of the form $p(qp)^+$ and Lemma 1 makes sure that one can find a unique pair p, q to express all of the words. □

Theorem 2. *The iterated palindromic completion of a word w is regular if and only if for all words $w' \in \{w\}_1$ with $w \neq w'$ there exist unique palindromes p and q with $|p| \geq 2$, such that:*

- $w' \in p(qp)^+$
- *w' has no palindromic prefixes except for the words in $p(qp)^*$.*

Proof. Due to Lemma 2, for the iterated completion we need only consider the finite union of all one sided iterated palindromic completion of words $w' \in \{w\}_1$.
(IF) For this direction the result is easily obtained, since, at each completion step, from some word of form $p(qp)^n$ with $n \geq 1$ we get all words $p(qp)^n, \ldots, p(qp)^{2n}$, for $n \geq 1$. Thus, the final result is a finite union of regular languages.
(ONLY IF) Now assume that $\{w\}_\infty$, the iterated palindromic completion of some word w, is regular. Following Theorem 1, $\{w\}_\infty$ can be written as the union of some finite language $\{p \mid p \text{ palindrome}\}$ and some finite union of languages $\{qr(sr)^*q^R \mid r, s \in \Sigma^* \text{ palindromes}\}$.

We neglect the case of the finite language $\{p \mid p \text{ palindrome}\}$, since this would contain just elements of $\{w\} \cup \{w\}_1$ that cannot be extended further on. Thus, we consider from $\{w\}_\infty$ only the finite union of languages of the form $\{qr(sr)^*q^R \mid q, r, s \in \Sigma^* \text{ and } r, s \text{ palindromes}\}$.

Following Dirichlet's principle for the finiteness of variables q with the help of the pigeon hole principle, we get that for some big enough integer k_1 and some i_1, we have that $qr(sr)^{k_1}q^R \vdash^* qr(sr)^{k_1+i_1}q^R$. We can apply Proposition 4 and get some palindromes u, v, such that $qr(sr)^*q^R \subset u(vu)^*$. Moreover, from the same Proposition we have that all the intermediate palindromic completion steps are in the language $qr(sr)^*q^R$, hence, in $u(vu)^+$. Now we know there exist at most finitely many pairs of palindromes u, v, such that $w' \in u(vu)^+$. Suppose that exist n pairs of palindromes (u_i, v_i) such that $w' \in u_i(v_iu_i)^+$ with $u_i \neq u_j$ and $|u_i| \geq 2$, for $1 \leq i \neq j \leq n$. With the help of the Fine and Wilf's periodicity Theorem and Lemma 3 we easily get that $u_iv_iu_i \in x(yx)^+$ for some palindromes x and y and $1 \leq i \leq n$ and the proof is concluded. □

Next natural question for one to consider, is what happens in the case of regular languages. We already know that the one step palindromic completion is not closed to regularity.

Proposition 5. *Iterated palindromic completion of a regular language can be non-context-free.*

Proof. Indeed, for this we just consider the language $L = \{aa^nba \mid n \geq 1\}$.

Taking a closer look at the iterated palindromic completion of L, we get that the language obtained is $L_\infty \subset L \cup L'$, where $L' \subset \{(\prod_{i\geq 1} a^{n_i}b)a^{n_1} \mid n_1 \leq n_i \leq 2n_1 - 2 \text{ for all } i\}$. It is easy to show, that the language $L_\infty \cap a^+ba^+ba^+$ is non-context-free, employing the pumping lemma for context-free languages. Since the class of context-free languages is closed under intersection with regular languages, it follows that L_∞ is outside the class, as well. □

Proposition 6. *Let $p, q, u \in \Sigma^*$ with p, q palindromes. If all palindromic prefixes of $upqpu^R$ are trivial ones, then for any $i \geq 0$ so are those of $up(qp)^i u^R$.*

Proof. Suppose p' is the shortest non-trivial palindromic prefix of any word $up(qp)^k u^R$, $k \geq 0$. Since p' is not a prefix of $upqpu^R$, the length of up is less than the length of p', hence, we have $p' = up(qp)^i x$, for some $i \leq k$ and word x which is a prefix of q, qp or u^R. If x is a prefix of q, then $x^R px$ is a suffix of p', hence, a non-trivial palindromic prefix of p', and, therefore, p' is not the shortest. If x is a prefix of qp, but not of q, then $x = qx'$ and $x'^R(qp)^i qx'$ is a palindromic suffix, hence, prefix of p', contradicting our assumption. Similarly, if x is a prefix of u, then $x^R p(qp)^i x$ is a shorter non-trivial palindromic prefix than p' itself. □

Theorem 3. *For a regular language L, its iterated palindromic completion L_∞ is regular if and only if L can be written as the union of disjoint regular languages L', L'', and L''', where*

- *$L' = L'_1 = \{w \in L \mid \{w\}_\infty \subseteq L\}$;*
- *$L'' = \{w \in L \mid \{w\}_1 = \{w\}_\infty \not\subseteq L\}$ and all words of L'' are prefixes[4] of words in the finite union of languages of the form $up(qp)^* u^R$, where $upqpu$ has only trivial palindromic prefixes and p, q are palindromes;*
- *$L''' = \{w \in L \mid \{w\}_\infty \neq \{w\}_1\}$ and all words of L''' are prefixes[4] of words in $\bigcup_{i=1}^n p_i(q_i p_i)^+$, where $n \geq 0$ is an integer depending on L and p_i, q_i are palindromes, with $p_i q_i$ primitive.*

Proof. (IF) This direction is straightforward given the fact that L is the union of three regular languages.
(ONLY IF) Clearly, any language $L \subset \Sigma^*$ can be written as a union of three disjoint languages where one of them (L') contains the words which have neither non-trivial palindromic prefixes nor suffixes or their iterated palindromic completion is included in L, another (L'') has all the words which have either non-trivial prefixes or suffixes, and the third one (L''') contains the words which can be extended in both directions by palindromic completion. If L_∞ and two of the aforementioned languages are regular, then the third one is, as well.

Here, we assume that L_∞ is regular, hence $L_\infty \setminus L$ is regular, too. Moreover, $L_\infty \setminus L$ is a palindromic language, since all of its words are the result of palindromic completion. From Theorem 1 it follows that there exists a finite set of words x_i, p_i, q_i, where $i \in \{1, \ldots, n\}$ and p_i, q_i are palindromes, such that the words in $L_\infty \setminus L$ are elements of $x_i p_i (q_i p_i)^* x_i^R$, for some $1 \leq i \leq n$.

First we will identify L'''. For each j, using once more the pigeon hole principle, it must be the case that there exist big enough integers k_1 and k_2 with

$$x_j p_j (q_j p_j)^{k_1} x_j^R \vdash^* x_j p_j (q_j p_j)^{k_2} x_j^R,$$

or for some $i \neq j$ and k_j, we have

$$x_j p_j (q_j p_j)^{k_j} x_j^R \vdash^* x_i p_i (q_i p_i)^{k_1} x_i^R \vdash^* x_i p_i (q_i p_i)^{k_2} x_i^R.$$

[4] the prefixes are at least $|up| + \lceil \frac{|q|}{2} \rceil + 1$ and $|p_i| + \lceil \frac{|q_i|}{2} \rceil + 1$ long, respectively, because the shorter ones do not extend beyond one step completion when pq (and $p_i q_i$, respectively) is primitive

In the first case we can apply Proposition 4 and get that there exist palindromes $p \neq \lambda$ and q such that $x_j p_j (q_j p_j)^{k_i} x_j^R \in p(qp)^+$, for $i \in \{1,2\}$, and all intermediary words $x_j p_j (q_j p_j)^{k_j} x_j^R$ are also in $p(qp)^+$. In the second case we can apply Proposition 4 to the second relation. Then by Lemma 1 and by Proposition 4 we get that all three words are in $p(qp)^+$, for suitable p, q. From here, the condition follows by the fact that the language of all prefixes of $\bigcup_{k=1}^{n} p_k(q_k p_k)^+$ is a regular language, hence, its intersection with L is also regular. That $p_i q_i$ are primitive, we can assume without loss of generality, because otherwise we can find their primitive root with the help of the Fine and Wilf Theorem and Lemma 3, which still satisfy the condition.

We know that $L_\infty \setminus L'''_\infty = L'_\infty \cup L''_\infty$ is regular, therefore $L_{\text{diff}} = (L'_\infty \cup L''_\infty) \setminus L \subset L''_\infty$ is a palindromic regular language. Again, from Theorem 1 we know that L_{diff} can be written as the finite union of languages of the form $up(qp)^* u^R$. Clearly then, all words in L'' are prefixes of some $up(qp)^n u^R$. Since by definition $L''_1 = L''_\infty$, we know that the words $up(qp)^n u^R$ have no non-trivial palindromic prefixes, hence, by Proposition 6 we have that $upqpu$ does not either. Assign L'' to be the finite union of the languages $up(qp)^+ \cap L$. This way, L'' is regular and since from L'' we can obtain L_{diff} by palindromic completion, it meets the requirements. All that is left is to designate L' to be $(L \setminus L''') \setminus L''$, which is regular and for all words we have that they either have only trivial palindromic prefixes or suffixes, or their palindromic completion is already in L. □

As a consequence of Theorems 1 and 3, the following result is obtained:

Corollary 1. *If, for some regular language L, we have that L_∞ is regular, then for any integer $n \geq 1$ we have that L_n is regular.*

4 Membership and decidability questions

We conclude this paper with some complexity results, which build on the previously obtained characterizations. In what follows, a deterministic finite automaton (DFA) is a quintuple $\langle Q, \Sigma, q_0, \sigma, F \rangle$, where Q is the set of states, q_0 the initial state, Σ the input alphabet, σ the transition function and F the set of final states. For details on finite automata and closure properties, see [2].

We start this part with of our investigation with the observation that while in the classical hairpin completion case the extension of a word is both to the right and the left of the word, here, due to the palindromicity property the two extensions are identical making this case somehow simpler. The membership problem for the one step palindromic completion of a word is trivial as one has to check for the shortest word if it is both a prefix and a suffix of the longer one and these two occurrences overlap. Obviously, the time needed for this is $\mathcal{O}(n)$ and the result is optimal. A more interesting problem is the membership problem for the iterated palindromic completion. We show that in this setting the problem is decidable, and, moreover, solvable in quadratic time.

Lemma 4. *If u, v are palindromes with u prefix of v and $|u| > |v|/2$, then $u \vdash v$.*

Proof. Since u is a prefix of v we can denote $v = xyx^Rz = z^Rx^Ryx$. Thus, z is a suffix of x^Ryx and now with the help of Proposition 3 the result is available.

Proposition 7. *For two palindromes u, v, we have $u \vdash^* v$ if and only if u is a prefix of v and for every prefix w of v with length greater than u, w has as prefix a non-trivial palindrome of length greater than $|w|/2$.*

Proof. In other words for palindromes u and v, we say that v can be obtained from u if and only if u is a prefix of v and for any palindromic prefixes of v they all have as prefix some palindrome of length greater than half theirs.
(ONLY IF) This case is quite obvious, since starting with the palindrome u we always have after some completions steps u as both prefix and suffix. Moreover, after each step the palindrome on which we do the completion is both a prefix and a suffix of the new word.
(IF) In order for v to be part of the iterated palindromic completion of a word it must be the case that second of the properties holds. Taking into account that v starts with u and the second property holds, with the help of Lemma 4 we get that v is in the language given by the iterated palindromic completion of u. □

Theorem 4. *One can decide in linear time if for two words u and v, where v is palindrome of length $n > |u|$, we have $u \vdash^* v$.*

Proof. Following the result of the previous Proposition we only need to check if any palindrome w from the palindromic completion of u is a prefixes of v, which is done in linear time, and then check if all palindromic prefixes of v have as prefix a palindrome of length at least half theirs. Identifying all palindromic prefixes of v of length greater than that of w is easily done in $\mathcal{O}(n)$ using s slight modification of the algorithm from [5]. Next, looking at the lengths of all elements in this set, we need to check that the difference between no two consecutive ones is double the smallest of them; again linear time is enough to do this. If YES then v is part of the iterated palindromic completion of v. □

As previously mentioned, one can identify in time $\mathcal{O}(n)$ all palindromic prefixes of some word v of length n. From those, one can efficiently compute the palindromic completion distance between two given words u and v. We start with the longest element of $\{u\}_1$, and in each step choose v's longest palindromic prefix which is shorter than twice the length of the current one. The greedy technique ensures optimality with the help of Proposition 7, while Lemma 4 proves the correctness of each step, therefore:

Theorem 5. *Given a word u and a palindrome v of length $n > |u|$, one can compute in linear time the minimum number of palindromic completion iterations needed in order to get from u to v, when possible.*

Let us now look at the regular closure property related to this operation.

Theorem 6. *For some word w of length n, it is decidable in $\mathcal{O}(n^2)$ whether $L_\infty(w)$ is regular.*

Proof. In linear time one can find all periods of a word. If we denote $n = p_i q_i + r_i$, where p_i are all periods of w and $r_i < p_i$, it is left to check if taking r_i the smallest palindrome from the sequence, for all other palindromes r_j with $r_i \leq_p r_j$ it is the case that $r_j = r_i(r_i v)^*$, for some unique palindrome v. Since deciding whether a word is palindrome is done in $\mathcal{O}(n)$, the result is concluded. □

Theorem 7. *Given a regular language L, it is decidable whether $L = L_\infty$.*

Proof. We suppose, without loss of generality, as the algorithm given here is intractable even for DFAs, that L is presented to us as a DFA with n states. The DFA is given by its set of states Q, initial state q_0, transition function $\sigma : Q \times \Sigma^* \to Q$ and set of final states F.
If $L \neq L_\infty$, then there exist some non-empty word u and palindrome p of length at least two, such that $up \in L$, but $upu^R \notin L$. Let us suppose that u is the shortest such word. We will show, that should u exist, we can find it after finitely many steps. Let L_{ul} denote denote the language $\{w \mid \sigma(q_0, w) = \sigma(q_0, u)\}$. Define F_u as the set of final states which we can reach by first reading u, that is,

$$F_u = \{q \in F \mid \exists w \in \mathcal{P}al : \sigma(q_0, uw) = q\}$$

and the language accepted starting from one of these states

$$L_{ur} = \{w \mid \exists p \in F_u, q \in F : \sigma(p, w) = q\}.$$

Then, u is the shortest word in $L_{ul} \setminus L_{ur}^R = L_{ul} \cap \overline{L_{ur}^R}$, where L^R is the reversal of L and $\overline{L}$ is the complement, i.e., $\overline{L} = \Sigma^* \setminus L$. The number of words to check in order to find u, if it exists, is unfortunately quite high:
- the automaton accepting L_{ul} has at most n states,
- for L_{ur} we get a NFA of at most n states, so at most 2^n states for the DFA,
- reversal and determinisation of the L_{ur} automaton takes it up to 2^{2^n} states,
- $L_{ul} \cap \overline{L_{ur}^R}$ results in an automaton having at most $n2^{2^n}$ states, and the shortest word accepted by an automaton is at most as long as the number of states.

Thus, for all words u with $|u| \leq n2^{2^n}$, we have to check $((u \cdot \mathcal{P}al) \cap L)u^R \setminus L = \emptyset$. If for at least one the set is not empty, we answer NO, otherwise YES. □

Theorem 8. *Given a regular language L, it is decidable whether L_∞ is regular. If the answer is YES, it is possible to construct an automaton accepting L_∞.*

Proof. The outline of the decision procedure is as follows: first we identify the words p_i, q_i forming L''', if there are any. Then we construct a DFA which accepts $L' \cup L'' = L \setminus L'''$. In the resulting automaton we check for the words u_k, p_k and q_k - if any - which form L'' and construct the automaton for $L' = (L \setminus L''') \setminus L''$, and from that, the DFA accepting the language L_{ps} of all proper prefixes and suffixes of words in L'. Last, we check whether $L' = L'_\infty$, that is $L' = L'_1$, and we can check this by Theorem 7. If yes, then L_∞ is regular, otherwise it is not.

The automata for the intermediary steps are computable using well-known algorithms (see [2]). What we have to show, is that the words u_k, p_k, q_k can be found, given an automaton. First, we check every cycle of length at most N_L (for the proof of the bound, see Appendix) in the automaton, where N_L is a constant computable from the representation of L. This can be easily done by a depth-first search. If the label of the cycle can be written as pq for some palindromes $p \neq \lambda$ and q, then we check all paths w of length at most N_L, which lead to the cycle from the initial state and all paths v of length at most N_L, going from the cycle to a final state. If there exist palindromes $x \neq \lambda$ and y such that xy is a cyclic shift of pq and $wpqv$ is a prefix or suffix of a word in $x(yx)^+$, then we identified a pair p_i, q_i for L'''. If there exist palindromes $x \neq \lambda$ and y and some word u, such that xy is a cyclic shift of pq and $wpqv = ux(yx)^i$ for some $i \geq 1$ then we identified a triple u_k, p_k, q_k for L''. After finding all pairs p, q for L''', we construct the automaton accepting $L \setminus L_{pq}$ for each pair, where L_{pq} is the set of prefixes of $p(qp)^+$ longer than $|p| + \lceil \frac{|q|}{2} \rceil + 1$. The language we get finally is $L' \cup L''$. Afterwards we subtract, for each triple u, p, q forming L'', the language of prefixes of $up(qp)^+u^R$ which are longer that $|up| + \lceil \frac{|q|}{2} \rceil + 1$. The resulting language is our candidate for L'. As mentioned above, if $L' = L'_1$, output YES, otherwise NO.

We end the proof by showing why it is enough to consider cycles of length not exceeding the number of states of a newly constructed automaton.

If L_∞ is regular, then so is L_1, by Corollary 1. If L_1 is regular, then Theorem 1 applies to $L_1 \setminus L$ and gives us that it can be written as the finite union of languages of the form $xr(sr)^*x^R$, with r, s palindromes.

For every state $p \in Q$, let us define the languages $\text{LEFT}_p = \{u \mid \sigma(q_0, u) = p\}$ and $\text{RIGHT}_p = \{u \mid \exists q \in F : \sigma(p, u) = q\}$. For every pair of states $p \in Q$, $q \in F$, such that there exists a palindrome $w \notin \Sigma \cup \{\lambda\}$, with $\sigma(p, w) = q$, let L_{pq} denote the language $\text{LEFT}_p \setminus \text{RIGHT}_q^R$, and for all other pairs $L_{pq} = \emptyset$. Now, the language

$$L_c = \bigcup_{p,q \in Q} L_{pq}$$

is regular, as it is the finite union of regular languages. Also, every word in L_c is the prefix of a word in one of the finitely many languages $xr(sr)^*x^R$ mentioned above. If L_c is infinite, then by pumping arguments and the Theorem of Fine and Wilf we easily get that the label of every cycle in the automaton accepting L_c is of the form w^k, where w is a cyclic shift of pq and $k \geq 1$. Hence, the same holds for cycles of length at most m, where m is the number of states of the automaton accepting L_c. On the other hand, suppose there is a pair r_1, s_1, such that all cycles which are cyclic shifts of $(r_1 s_1)^k$, for some $k \geq 1$, are longer than m. Then, again by pumping and the pigeon hole principle, we get that $r_1 s_1$ is the cyclic shift of some other pair r_2, s_2, where $|r_2 s_2| \leq m$. Hence, we conclude that by checking all cycles of length at most m of the automaton accepting L_c we can discover the pairs r, s from the characterization of palindromic languages in Theorem 1. The automaton accepting L_c can be constructed, given L, and from there m is computed by counting the states. □

References

1. Aldo de Luca. Sturmian words: Structure, combinatorics, and their arithmetics. *Theor. Comput. Sci.*, 183(1):45–82, 1997.
2. M. A. Harrison. *Introduction to Formal Language Theory.* Addison-Wesley, Reading, Massachusetts, 1978.
3. S. Horváth, J. Karhumäki, and J. Kleijn. Results concerning palindromicity. *J. Inf. Process. Cybern.*, 23:441–451, November 1987.
4. M. Lothaire. *Combinatorics on Words.* Cambridge University Press, 1997.
5. Glenn Manacher. A new linear-time "on-line" algorithm for finding the smallest initial palindrome of a string. *Journal of the ACM*, 22(3):346–351, 1975.