International Journal of Foundations of Computer Science
© World Scientific Publishing Company

# INNER PALINDROMIC CLOSURE*

JÜRGEN DASSOW
*Otto-von-Guericke-Universität Magdeburg, Fakultät für Informatik, PSF 4120,*
*D-39016 Magdeburg, Germany, dassow@iws.cs.uni-magdeburg.de*

FLORIN MANEA and ROBERT MERCAŞ and MIKE MÜLLER
*Christian-Albrechts-Universität zu Kiel, Institut für Informatik,*
*D-24098 Kiel, Germany, {flm,rgm,mimu}@informatik.uni-kiel.de*

Received (Day Month Year)
Accepted (Day Month Year)
Communicated by (xxxxxxxxxx)

We introduce the inner palindromic closure as a new operation ♠, which consists in expanding a factor $u$ to the left or right by a non-empty word $v$ such that $vu$ or $uv$, respectively, is a palindrome of minimal length. We investigate several language theoretic properties of the iterated inner palindromic closure $♠^*(w) = \bigcup_{i\geq 0} ♠^i(w)$ of a word $w$.

*Keywords*: Palindromic closure; Regular languages.

## 1. Introduction

The investigation of repetitions of factors in a word is a very old topic in formal language theory. For instance, already in 1906, Thue proved that there exists an infinite word over an alphabet with three letters which has no factor of the form $ww$. Since the eighties a lot of papers on combinatorial properties concerning repetitions of factors were published (see [16] and the references therein).

The repetitive aspect received further interest in connection with its importance in natural languages and in DNA sequences and chromosomes [17]. Motivated by these applications, grammars with derivations consisting in "duplications" (a word $xuwvy$ is derived to $xwuwvy$ or $xuwvwy$), were introduced, see [5].

Combining the combinatorial, linguistic and biological aspect, the duplication language $D(w)$ associated to a word $w \in \Sigma^+$, which is the language containing all words that double some factor of $w$, i. e., $D(w) = \{xuuy \mid w = xuy, x, y \in \Sigma^*, u \in \Sigma^+\}$ and its iterated version $D^*(w) = \bigcup_{i\geq 0} D^i(w)$, were introduced. In the papers [1, 4, 7, 18], the regularity of $D^*(w)$ was discussed; for instance, it was

*F. Manea is supported by the *DFG* grant 596676. R. Mercaş and M. Müller are supported by the *DFG* grant 582014.

shown that, for any word $w$ over a binary alphabet, $D^*(w)$ is regular and that $D^*(abc)$ is not regular. Further results on iterated duplication languages can be found in [12], while for cases of bounded duplication, i. e., the duplicated words have length bounded by a constant, we refer to [13].

It was noted that words $w$ containing hairpins (i. e., $w = xuyh(u^R)z$) and words $w$ with $w = xuy$ and $u = h(u^R)$, where $u^R$ is the mirror image of $u$ and $h$ is a letter-to-letter isomorphism, are of interest in DNA structures (see [10, 11], where the Watson-Crick complementarity gives the isomorphism). Therefore, operations leading to words with hairpins as factors were studied, see [2, 3].

Following [8], here we consider a word generating operation that leads to words which have palindromes (words that read the same from both left and right) as factors (which is a restriction to the identity as the isomorphism $h$ above). An easy step would be to obtain $xuu^Ry$ from a word $xuy$ in analogy to the duplication. But then all newly obtained palindromes are of even length. Thus it seems to be more interesting to consider the palindrome closure defined by de Luca [6]. Here a word is extended to a palindrome of minimal length. We allow this operation to be applied to factors and call it inner palindromic closure. We also study the case of iterated operation applications and a restricted variant of the closure operation.

The paper is organised as follows. After presenting the main basic concepts, we define the new operation, inner palindromic closure, and its versions in Section 2, where we also list some simple properties. In Sections 3 and 4, we discuss the regularity of sets obtained by the inner palindromic closures. Finally, we present some language classes associated with the new operation.

**Basic definitions.** For more details on the concepts we define here see [16].

A set $M \subseteq \mathbb{N}^m$ of vectors is called linear, if it can be represented as

$$M = \{B + \sum_{i=1}^{n} \alpha_i A_i \mid \alpha_i \in \mathbb{N}, 1 \le i \le n\}$$

for some vectors $B$ and $A_i$, $1 \le i \le n$. It is called semi-linear if it can be represented as a finite union of linear sets.

An alphabet $\Sigma$ is a non-empty finite set with the cardinality denoted by $\|\Sigma\|$, and the elements called letters. A sequence of letters constitute a word $w \in \Sigma^*$, and we denote the *empty word* by $\varepsilon$. The set of all finite words over $\Sigma$ is denoted by $\Sigma^*$, and any subset of it is called a language. Moreover, for a language $L$, by $\text{alph}(L)$ we denote the set of all symbols occurring in words of $L$.

If $w = u_1v_1u_2v_2 \dots u_nv_n$ and $u = u_iu_{i+1} \dots u_j$ for $1 \le i \le j \le n$, we say that $u$ is a *scattered factor* of $w$, denoted as $u \preccurlyeq w$. Consider now $v_k = \varepsilon$ for all $1 \le k \le n$. We say that $u$ is a *factor* of $w$, and, if $i = 1$ we call $u$ a *prefix*, denoted $u \le_p w$. If $j = n$ we call $u$ a *suffix*, denoted $u \le_s w$. If $i > 1$ or $j < |w|$, then $u$ is called *proper*.

The length of a finite word $w$ is the number of all symbols it consists of, denoted by $|w|$. The number of occurrences of some letter $a$ in $w$ is designated by $|w|_a$. The *Parikh vector* of a word $w \in \Sigma^*$ with $\Sigma = \{a_1, a_2, \dots, a_{\|\Sigma\|}\}$, denoted by $\Psi(w)$, is defined as $\Psi(w) = \langle |w|_{a_1}, |w|_{a_2}, \dots, |w|_{a_{\|\Sigma\|}} \rangle$. A language $L$ is called linear or semi-linear, if its set of Parikh vectors is linear or semi-linear, respectively.

For $i \geq 0$, the $i$-fold catenation of a word $w$ with itself is denoted by $w^i$ and is called the $i$th power of $w$. When $i = 2$, we call the word $w^2 = ww$ a square.

For a word $w \in \Sigma^*$, we denote its mirror image (or reversal) by $w^R$ and say that $w$ is a palindrome if $w = w^R$. For a language $L$, let $L^R = \{w^R \mid w \in L\}$.

We say that a language $L \subseteq \Sigma^*$ is *dense*, if, for any word $w \in \Sigma^*$, $\Sigma^* w \Sigma^* \cap L$ is non-empty, i. e., each word occurs as a factor in $L$.

We recall Higman's Theorem.

**Theorem 1 (Higman [9])** *If $L$ is a language such that any two words in $L$ are incomparable with respect to the scattered factors partial order, then $L$ is finite.*

## 2. Definitions and preliminary results

We now look at a word operation following [6, 8], which consider extensions to the left and right of words such that the newly obtained words are palindromes.

**Definition 2.** *For a word $u$, the left (right) palindromic closure of $u$ is a palindrome $v$ with $|v| > |u|$ such that any other palindrome having $u$ as proper suffix (proper prefix) has length greater than $|v|$.*

Here the newly obtained words have the length minimal among all palindromes that have the original word as prefix or suffix.

As for duplication and reversal, we can now define a further operation.

**Definition 3.** *For a word $w$, the left (right) inner palindromic closure of $w$ is the set of all words $xvuy$ ($xuvy$) for which there exists a factorisation $w = xuy$ with possibly empty $x, y$ and non-empty $u, v$, such that $vu$ ($uv$) is the left (right) palindromic closure of $u$. We denote these operations by $\spadesuit_\ell(w)$ and $\spadesuit_r(w)$, respectively, and define the inner palindromic closure $\spadesuit(w)$ as the union of $\spadesuit_\ell(w)$ and $\spadesuit_r(w)$.*

The operation is extended to languages and an iterated version is introduced.

**Definition 4.** *For a language $L$, let $\spadesuit(L) = \bigcup_{w \in L} \spadesuit(w)$. We set $\spadesuit^0(L) = L$, $\spadesuit^n(L) = \spadesuit(\spadesuit^{n-1}(L))$ for $n \geq 1$, $\spadesuit^*(L) = \bigcup_{n \geq 0} \spadesuit^n(L)$. Any set $\spadesuit^n(L)$ is called a finite inner palindromic closure of $L$, and we say that $\spadesuit^*(L)$ is the iterated inner palindromic closure of $L$.*

The first observation is immediate, as the operation does not erase letters:

**Lemma 5.** *For every word $w$, if $u \in \spadesuit^*(w)$, then $w \preccurlyeq u$.*

The next results are in tone with the ones from Proposition 3.1.1 in [12]:

**Proposition 6.** *For any semi-linear (linear) language, its iterated inner palindromic closure is semi-linear (respectively, linear).*

**Proof.** Let $\text{alph}(L) = \Sigma$. For a subset $U$ of $\Sigma$, let $L_U$ be the set of words in $L$ which contain only letters from $U$ and each letter of $U$ occurs at least once.

Since $L$ is semi-linear, so it is every $L_U$, too. Hence there are vectors $B_j$, $1 \leq j \leq m$, and $A_{j,i}$, $1 \leq j \leq m$, $1 \leq i \leq n_j$, for $m$ and $n_j$, $1 \leq j \leq m$ such that

$$\Psi(L_U) = \bigcup_{j=1}^{m} \{B_j + \sum_{i=1}^{n_j} \alpha_{j,i} A_{j,i} \mid \alpha_{j,i} \in \mathbb{N}\}.$$

Since any inner palindromic closure of a $w \in L_U$ adds a word $v$ over $U$ which has a linear combination of the vectors $\Psi(a)$, for all $a \in U$, as its Parikh vector $\Psi(v)$, and any letter can be introduced by inner palindromic closures in an arbitrary number,

$$\Psi(\spadesuit^*(L_U)) = \bigcup_{j=1}^{m} \{B_j + \sum_{i=1}^{n_j} \alpha_{j,i} A_{j,i} + \sum_{a \in U} \beta_a \Psi(a) \mid \alpha_{j,i} \in \mathbb{N}, \beta_a \in \mathbb{N}\}.$$

Thus $\spadesuit^*(L_U)$ is semi-linear for any $U$. Because $\spadesuit^*(L) = \bigcup_{U \subseteq \Sigma} \spadesuit^*(L_U)$ and the union of semi-linear sets is also semi-linear, $\spadesuit^*(L)$ is semi-linear. □

As singletons are linear sets, we make the following assertion:

**Corollary 7.** *For any word, its iterated inner palindromic closure is linear.*

The following result is a somewhat simpler version of Proposition 12.

**Proposition 8.** *For any word $w$, the language $\spadesuit^*(w)$ is dense over alph$(w)$.*

**Proof.** Let $v = a_1 a_2 \dots a_n$ be a word over $\Sigma = \text{alph}(w)$.

Let $w = b_1 b_2 \dots b_r a_1 w'$, where $b_i \neq a_1$ for $1 \leq i \leq r$. Then we obtain by inner palindromic closures the following sequence of words starting from $w$:

$$b_1 b_2 \dots b_{r-1} a_1 b_r a_1 w', \quad b_1 b_2 \dots a_1 b_{r-1} a_1 b_r a_1 w', \quad \dots, \quad a_1 b_1 a_1 \dots a_1 b_{r-1} a_1 b_r a_1 w',$$

i. e., we obtain a word $a_1 w''$, where all letters occurring in $w$ also occur in $w''$. Now we can "move" $a_2$ from its first position in $w''$ to the left (as above with $a_1$) and get a word $a_1 a_2 w'''$. This process is continued with $a_3, a_4, \dots, a_n$ and a word $a_1 a_2 \dots a_n \overline{w}$ is produced which belongs to $\spadesuit^*(L)$ and has $v$ as a subword. □

We mention that Proposition 8 does not hold for languages. This can be seen from $L = \{ab, ac\}$. Obviously, any word in $\spadesuit^*(L)$ contains only $a$ and $b$ or only $a$ and $c$. Therefore $abc \in \Sigma^*$ is not a factor of any word in $\spadesuit^*(L)$.

**Lemma 9.** *Let $\Sigma = \{a_1, a_2, \dots, a_k\}$ and define the recursive sequences*

$$w'_i = w_{i-1} w'_{i-1} \text{ and } w_i = w'_i a_i \text{ for } 1 \leq i \leq k,$$

*with $w'_0 = w_0 = \varepsilon$. Then alph$(w_i)^* w_i \subseteq \spadesuit^*(w_i)$.*

**Proof.** Note that, for $0 \leq j < i \leq k$, $w'_i$ is a palindrome and $w_j <_p w_i$. We want to generate $b_1 b_2 \dots b_n w_i$ with $b_\ell \in \text{alph}(w_i)$ for $1 \leq \ell \leq n$. Let $b_1 = a_j$. Since $w_j \leq_p w_i$, $w_i = w'_j a_j v$ for some $v$. Since $w'_j$ is a palindrome, we obtain $a_j w'_j a_j v = b_1 w_i$ by an inner palindromic closure step. The conclusion follows after performing the procedure in succession for $b_2, \dots, b_n$. □

We now define a variant of the inner palindromic closure, where we restrict the length of the words which are involved in the palindromic closure. First we introduce a parametrised version of the palindromic closure operation from [6].

**Definition 10.** *For a word $u$ and $m, n \in \mathbb{N}$, we define the sets*

$$L_{m,n}(w) = \{u \mid u = u^R, u = xw \text{ for } x \neq \varepsilon, |x| \geq n,\ m \geq |w| - |x| \geq 0\},$$
$$R_{m,n}(w) = \{u \mid u = u^R, u = wx \text{ for } x \neq \varepsilon, |x| \geq n,\ m \geq |w| - |x| \geq 0\}.$$

*The left (right) $(m, n)$-palindromic closure of $w$ is the shortest word of $L_{m,n}(w)$ (resp., $R_{m,n}(w)$), or undefined if $L_{m,n}(w)$ (resp., $R_{m,n}(w)$) is empty.*

Therefore, an element of $L_{m,n}(w)$ is a palindrome $u$ obtained by extending the word $w$ by a prefix $x$ of length at least $n$ such that the centre of the newly obtained palindrome $u$ is inside the prefix of length $\lceil \frac{m}{2} \rceil$ of $w$. That is, $u = xvv^Rx^R$ where $n \leq |x|$, $2|v| \leq m$, and $w = vv^Rx^R$, or $u = xvav^Rx^R$, where $n \leq |x|$, $2|v| + 1 \leq m$, and $w = vav^Rx^R$. The left $(m, n)$-palindromic closure is the shortest such word $u$ (the shortest $v$ is chosen). The right $(m, n)$-palindromic closure is defined similarly.

We define now the parametrised version of the inner palindromic closure.

**Definition 11.** *For non-negative integers $n, m$ with $n > 0$, we define the $\spadesuit_{(m,n)}$ one step inner palindromic closure of a word $w$ as*

$$\spadesuit_{(m,n)}(w) = \{xy'z \mid w = xyz, y' \text{ is the left/right } (m, n)\text{-palindromic closure of } y\}.$$

This notion can easily be extended to languages, while its iterated version $\spadesuit^*_{(m,n)}$ is defined just as in the case of the inner palindromic closure.

A statement similar to Proposition 8 also holds for the bounded operation.

**Proposition 12.** *For any word $w$ with $|w| \geq n$ and positive integer $m$, the language $\spadesuit^*_{(m,n)}(w)$ is dense with respect to the alphabet alph$(w)$.*

**Proof.** Note that if $u \leq_p w$ with $|u| \geq n$ and $u[|u|] = a$, then there is a word $w'$ starting with $a$ in $\spadesuit_{(m,n)}(w)$. If the letter $a$ appears only in the prefix of length $n-1$ of $w$, then we do as follows. Let $w_0 = w$ and let $w_{i+1}$ be the left $(m, n)$-palindromic closure of $w_i$ for $i \geq 0$. As $w_i <_p w_{i+1}$, there exists $i_a$ such that $w_{i_a}$ has a prefix of length at least $n$ that ends with $a$. Continuing this process, we derive a word $w'$ that for each letter $s \in \text{alph}(w)$ has a prefix of length at least $n$ ending with $s$.

We now show that every word $a_1 \cdots a_n \in \text{alph}(w)^*$ appears as a prefix of some word in $\spadesuit_{(m,n)}(w)$: First, we generate $w'$ and let $v_0 = w'$. By the above, $v_0 = x_1a_1y_1$ for some $|x_1| \geq n - 1$. Then, with a left $(m, n)$-palindromic closure on $x_1a_1$ (which produces a word from $\spadesuit_{(m,n)}(v_0)$), we obtain from $v_0$ a palindrome $a_1v_1$, where $v_0 <_p v_1$. Thus, $v_1$ also has prefixes of length greater than $n$ that end with every letter in alph$(w)$. Next, $a_1a_2v_2$ is obtained from $a_1v_1 = a_1x_2a_2y_2$ by a left $(m, n)$-palindromic closure on $x_2a_2$. The process is repeated until we get $a_1 \cdots a_nv_n$. □

The next result is related to Proposition 12 and will be useful in the sequel.

**Lemma 13.** *Let $\Sigma$ be an alphabet with $||\Sigma|| \geq 2$ and $a \notin \Sigma$, and integers $m, n > 0$. Let $w = a^m y_1 a \cdots y_{p-1} a y_p$ be a word such that $alph(w) = \Sigma \cup \{a\}$, $m, p > 0$, $y_i \in \Sigma^*$ for $1 \leq i \leq p$, $|y_1| > 0$, and for some $j$ with $1 \leq j \leq p$, $|y_j| \geq n$. Then, for each $v \in \Sigma^*$ with $|v| \geq n$, there exists $w' \in \spadesuit^*_{(m,n)}(w)$ such that $v \leq_p w'$ and $|w'|_a = |w|_a$.*

**Proof.** As a first step, for a word $z = z_1 a z_2 a \cdots a z_k$, where $a \notin \bigcup_{1 \leq i \leq k} \text{alph}(z_i)$ and $|z_1| \geq n$, we define $z'_1 = z_1$ and $z'_i = (z'_{i-1})^R z_i$ for $1 < i \leq k$. Let $z' = z'_1 a \cdots a z'_k$. It is immediate that $z' \in \spadesuit^*_{(m,n)}(z)$, as it is obtained by applying iteratively right $(m,n)$-palindromic closure to the factors $z'_i a$ to get $z'_i a (z'_i)^R$, for $i > 0$. Moreover, $\text{alph}(z'_k) = \bigcup_{1 \leq i \leq k} \text{alph}(z_i)$ and $|z'_i| \geq n$ for all $1 \leq i \leq k$.

As a second step, for a word $v = v_\ell a v_{\ell-1} a \cdots a v_1$, where $a \notin \bigcup_{1 \leq i \leq \ell} \text{alph}(v_i)$ and $|v_1| \geq n$, we define $v'_1 = v_1$ and $v'_i = v_i (v'_{i-1})^R$ for $1 < i \leq \ell$. Let $v' = v'_\ell a \cdots a v'_1$. It is immediate that $v' \in \spadesuit^*_{(m,n)}(v)$, as it can be obtained by applying iteratively left $(m,n)$-palindromic closure to the factors $a v'_i$ to obtain $(v'_i)^R a {v'_i}^R$, for $i > 0$. Moreover, $\text{alph}(y'_\ell) = \bigcup_{1 \leq i \leq k} \text{alph}(y_i)$ and $|y'_i| \geq n$ for all $1 \leq i \leq \ell$.

Now consider the word $w$ from our hypothesis. We apply the first step described above to the factor $y_j a y_{j+1} \cdots a y_p$ to obtain $y'_j a \cdots a y'_p$, where $\text{alph}(y'_p) = \bigcup_{j \leq i \leq p} \text{alph}(y'_i)$ and $|y'_p| \geq n$. Afterwards, we apply the second step procedure to the factor $y_1 a y_2 a \cdots a y_{j-1} a y'_j a \cdots a y'_p$ to obtain $y''_1 a y''_2 a \cdots a y''_{j-1} a y''_j a \cdots a y''_p$, where $\text{alph}(y''_1) = \bigcup_{1 \leq i \leq p} \text{alph}(y_i) = \Sigma$ and $|y''_1| \geq n$. Accordingly, $w'' = a^m y''_1 a y''_2 a \cdots a y''_p \in \spadesuit^*_{(m,n)}(w)$.

Now, for $v \in \Sigma^*$ we obtain the word $w''_v = a^m v^R y_v a \cdots y''_p a$ from $w''$, for some $y_v \in \Sigma^*$, just as in the proof of Proposition 12. If $|v| \geq n$, we can obtain from $w''_v$ the word $v w''_v$ by applying to $a^m v^R$ a left $(m,n)$-palindromic closure to get $v a^m v^R$. This concludes our proof. □

## 3. On the regularity of the inner palindromic closure

We start with some facts regarding words over a binary alphabet.

**Lemma 14.** *For $w = a^n b^m$ with $n, m > 0$, $\spadesuit(w)$ contains all words of length $n + m + 1$ with either $a$ or $b$ inserted at any position $i$ of $w$, where $0 \leq i < n + m$.*

**Proof.** To insert $a$ between positions $j$ and $j+1$ with $j < n$ we take the palindromic prefix $a^{n-j}$ and perform a $\spadesuit_\ell$ step on it to get $a^{n-j+1}$ which fulfils the conditions. When $n \leq j \leq m$, we perform a $\spadesuit_r$ step on the word $ab^{j-n}$, which produces the palindrome $ab^{j-n}a$. The case of the insertion of a letter $b$ is symmetrical. □

Consequently, for binary words the converse of Lemma 5 holds as well:

**Corollary 15.** *For any binary words $w$ and $u$, $w \preccurlyeq u$ if and only if $u \in \spadesuit^*(w)$.*

**Proof.** By Lemma 5, we have that $w$ is a scattered factor of all words in $\spadesuit^*(w)$. Using Lemma 14, all words $u$ having $w$ as scattered factor are in fact in $\spadesuit^*(w)$

since in each of them we can insert $a$'s and $b$'s at arbitrary positions. The idea is to assign to each element of $w$ a corresponding element in $u$, and next, considering only two consecutive groups of $a$'s and $b$'s, use Lemma 14 to expand these groups or introduce new elements, as necessary, □

Bovet and Varricchio [1] showed that iterated duplication closures of binary languages are regular. The inner palindromic closure operation behaves similarly:

**Theorem 16.** *If $L \subseteq \{a,b\}^*$, then the language $\spadesuit^*(L)$ is regular.*

**Proof.** According to Theorem 1, there exists a finite set $L_0$ with $L_0 \subseteq L$ such that for every word $w \in L$ there is a word $w_0 \in L_0$ with $w_0 \preccurlyeq w$. By Corollary 15, $\spadesuit^*(L)$ is the union of the sets $SW(w_0) = \{w' \in \text{alph}(w_0)^* \mid w_0 \preccurlyeq w'\}$, for all $w_0 \in L_0$. As all the sets $SW(w_0)$ are regular, it follows that $\spadesuit^*(L)$ is regular. □

Let us now shift our attention to alphabets of at least three letters. Obviously, the finite inner palindromic closure of a finite language is finite and thus regular. However, for regular languages the closure is not necessarily regular.

**Theorem 17.** *If $L$ is regular and $k$ a positive integer, $\spadesuit^k(L)$ can be non-regular.*

**Proof.** For a fixed $k$, we consider the language $L = c_1a_1^+c_2a_2^+\ldots c_ka_k^+b$. We intersect $\spadesuit^k(L)$ with the language given by the regular expression:

$$c_1a_1^+c_2a_2^+\ldots c_ka_k^+b(a_k^+c_k\ldots a_2^+c_2a_1^+c_1)(a_k^+c_k\ldots a_3^+c_3a_2^+c_2)\ldots(a_k^+c_ka_{k-1}^+c_{k-1})a_k^+c_k$$

It is not hard to see that in any word of the intersection the number of $a_i$'s in every maximal unary group adjacent to $c_i$ is the same. Since this is a non-regular language and regular languages are closed under intersection, we conclude. □

It remains an *open problem* whether or not the iterated inner palindromic closure of a regular language $L$, where $\|\text{alph}(L)\| \geq 3$, is also regular.

## 4. Parametrised inner palindromic closure

We now discuss the regularity of $\spadesuit^*_{(m,n)}(w)$. Before we state our results, we establish two facts on the avoidance of patterns. Due to space constraints we only prove completely the second result.

**Theorem 18.** *There exist infinite binary words avoiding both palindromes of length 6 and longer, and squares of words with length 3 and longer.*

**Proof.** Rampersad et al. [15] exhibited an infinite square-free word $w$ avoiding the factors $ac, ad, ae, bd, be, ca, ce, da, db, eb$, and $ec$. The morphism $\gamma$, defined by

$$\gamma(a) = abaabbab, \quad \gamma(b) = aaabbbab, \quad \gamma(c) = aabbabab,$$
$$\gamma(d) = aabbbaba, \quad \gamma(e) = baaabbab,$$

maps this word $w$ to a word with the desired properties:

As any palindrome of length $n > 2$ contains one of length $n-2$, a word avoiding palindromes of lengths 6 and 7 also avoids longer ones. Also, palindromes of length 6 or 7 would occur in the images of words of length 2. No such palindromes appear in $\gamma(\{ab, ba, bc, cb, cd, dc, de, ea, ed\})$, therefore neither in $\gamma(w)$. To show that $\gamma(w)$ contains only the squares $aa, bb, abab$ and $baba$ we apply methods similar to [15]. □

**Theorem 19.** *There exist infinite ternary words avoiding both palindromes of length* 3 *and longer, and squares of words with length* 2 *and longer.*

**Proof.** We claim that the morphism $\psi$, that is defined by

$$\psi(a) = abbccaabccab, \qquad \psi(b) = bccaabbcaabc, \qquad \psi(c) = caabbccabbca,$$

maps all infinite square-free ternary words $h$ to words with the desired properties.

It it straightforward to see that $\psi(h)$ does not contain palindromes of length 3 or 4, as all those would be contained inside $\psi(u)$ for some length 2 word $u$ and we can check that there are none. Hence, no longer palindromes exist either.

It remains to show, there are no longer squares occurring in $\psi(h)$. For squares $ww$ with $|w| \leq 22$, we have to check only finitely many cases, as those squares are completely contained inside the image of some factor of length 5 of $h$. Computer calculations show that no such squares occur. Assume for the sake of a contradiction that $\psi(h)$ contains a square $ww$ with $w \geq 23$ and consider its earliest occurrence. We have that each occurrence of $w$ contains at least one full image of a symbol under $\psi$. Thus $w = s\psi(u)p$ for some proper factor $u$ of $h$ and $s$ (respectively, $p$) being a proper suffix (respectively, prefix) of $\psi(x)$ for some $x \in \Sigma$. Since the images of every letter occurs as either prefix or suffix of some $\psi(xy)$, where $x, y \in \Sigma$, we conclude that $|s| + |p| = 12$ and $ps = \psi(z)$ for some $z \in \Sigma$. If $s = \varepsilon$ it follows that $ww = \psi(uzuz)$, a contradiction with the fact that $h$ is square-free. Otherwise, given that the last letter of $s$ uniquely determines its preimage $\psi(z)$, we reach again a contradiction since $\psi(zuzu)$ is an early occurrence of a square in $\psi(h)$. □

Next we exhibit a method to construct words whose iterated inner $(m, n)$-palindromic closure is not regular, for integers $m, n > 0$. To this end, we associate to an integer $k \geq 2$ a pair of numbers $(p_k, q_k)$ if there exists a $k$-letter infinite word avoiding both palindromes of length greater or equal to $q_k$ and squares of words of length greater or equal to $p_k$. If more pairs exist, we take $(p_k, q_k)$ to be any of them.

**Theorem 20.** *Let $m > 0$ and $k \geq 2$ be two integers and define $n = \max\{\frac{q_k}{2}, p_k\}$. Let $\Sigma$ be a $k$-letter alphabet with $a \notin \Sigma$ and $w = a^m y_1 a y_2 \cdots a y_{r-1} a y_r$ be a word such that alph$(w) = \Sigma \cup \{a\}$, $r > 0$, $y_i \in \Sigma^*$ for all $1 \leq i \leq r$, and there exists $j$ with $1 \leq j \leq r$ and $|y_j| \geq n$. Then $♠^*_{(m,n)}(w)$ is not regular.*

**Proof.** Let $\alpha$ be an infinite word over $\Sigma$ that avoids palindromes of length at least $q_k$ and squares of words of length $p_k$. Due to Proposition 12, for each prefix $u$ of $\alpha$ longer than $n$, there exists $w_u$ with $|w_u|_a = r - 1$ such that $ua^m w_u \in ♠^*_{(m,n)}(w)$.

We analyse how the words $ua^mv$, with $u \leq_p \alpha$ and $|v|_a = r-1$, are obtained by iterated $(m,n)$-palindromic closure steps from $w$. As $u$ contains no $a$'s, no squares of words of length $p_k$, as well as no palindromes with length greater than or equal to $q_k$, and the application of an $(m,n)$-palindromic closure step introduces a palindrome in the derived word, we get that the only possible cases of application of the operation in the derivation of $ua^mv$ are the following:

**(1)** $v = xyz$ and $y$ is the $(m,n)$-palindromic closure of $y'$ (thus, $|y'| < |y|$ and $|y|_a = |y'|_a$); in this case we have that $ua^mv$ is in $\spadesuit_{(m,n)}(ua^mxy'z)$.
**(2)** $u = u'x$, $v = yz$, and $xa^my$ is the $(m,n)$-palindromic closure of $a^my$ (thus, $x = y^R$ and neither $x$ nor $y$ contain $a$'s); now we have that $ua^mv$ is in $\spadesuit_{(m,n)}(u'a^myz)$.
**(3)** $u = xyz$ and $y$ is the $(m,n)$-palindromic closure of $y'$ (thus, $|y'| < |y|$ and $y'$ contains no $a$'s); in this case we have that $ua^mv$ is in $\spadesuit_{(m,n)}(xy'za^mv)$.

Since we only apply $(m,n)$-palindromic closure operations, and the word we want to derive has the form $ua^mv$ with $|a^mv|_a = |w|_a$, it is impossible to apply any palindromic closure step that adds to the derived word more $a$ symbols or splits the group $a^m$ that occurs at the beginning of $w$. Intuitively, the palindromic closure operations that we apply are localised, due to the restricted form of the operation: they either occur inside $u$, or inside $v$, or are centred around $a^m$.

Moreover, by choosing $n \geq \frac{q_k}{2}$ if at any step we apply a palindromic closure operation of the type **(3)** above, then the final word $u$ contains a palindrome of length greater than $q_k$. To see this, we assume, for the sake of a contradiction, that such an operation was applied. Then, we look at the last operation of this kind that was applied. Obviously, none of the operations of type **(1)** or **(2)** that were applied after that operation of type **(3)** could have modified the palindrome of length at least $q_k$ introduced by it in the derived word before $a^m$. Thus, that palindrome would also appear in $u$, a contradiction.

Therefore, all the intermediate words obtained during the derivation of $ua^mv$ from $w$ have the form $u'a^mv'$, where $u' \leq_p \alpha$ (maybe empty) and $v'$ has exactly $|w|_a - m$ symbols $a$. We now look at the operations that can be applied to such a word. In particular, we note that we cannot have more than $|v'| - n$ consecutive derivation steps preserving the length of the word occurring after the first sequence of $a$'s. Hence, we can apply at most $|v'| - n$ consecutive operations of type **(2)**.

Indeed, after $\ell$ such derivation steps one would obtain from $u'a^mv'$ a word $u'v_1 \cdots v_\ell a^mv'$ where $v_i^R \leq_p v'$ and $|v_i| \geq n$ for every $1 \leq i \leq \ell$. Assume towards a contradiction, that $\ell > |v'| - n$. Then, exists $j$ so that $1 \leq j < \ell$ and $|v_j| \geq |v_{j+1}|$. Therefore, $u'v_1 \cdots v_\ell$ contains a square of length at least $2n \geq 2p_k$. This square remains for the rest of the derivation, as neither a type **(1)** nor a type **(2)** operation can introduce letters inside it. Another contradiction with the form of $u$ is reached.

We use this last remark to show by induction on the number of derivation steps, for every prefix $u$ of $\alpha$ and $ua^mv \in \spadesuit^*_{(m,n)}(w)$, that $|u| \leq |v|^3$ must hold.

If the word is derived in only one step, this holds trivially, since the fact that the prefix $u$ can be added to $w$ implies that $|u| \leq |y_1|$.

Let us now assume that it holds for words obtained in at most $k$ derivation steps, and show it for words obtained in $k+1$ derivation steps. If the last applied step to obtain $ua^m v$ is of type **(1)**, then we obtained $ua^m v$ from $ua^m v'$ for some $v'$ shorter than $v$. By induction, we have that $|u| \leq |v'|^3$, and, consequently, $|u| \leq |v|^3$. According to our previous remark, at most the last $|v| - n$ consecutive steps applied were of type **(2)**. These increased the length of $u$ by at most $\sum_{n \leq i \leq |v|} i \leq \frac{|v|(|v|+1)}{2}$. Therefore, we get $|u| - \frac{|v|(|v|+1)}{2} \leq (|v|-1)^3$; hence $|u| \leq |v|^3$.

We now show that the language

$$L = \{ua^m v \in \spadesuit^*_{(m,n)}(w) \mid |u| \geq n, |v|_a = r-1\}$$

is not regular. Since this language is obtained from $\spadesuit^*_{(m,n)}(w)$ by intersection with a regular language, if $L$ is not regular, then $\spadesuit^*_{(m,n)}(w)$ is not regular either.

We consider a word $u_0 a^m v_0 \in L$ such that $u_0 \leq_p \alpha$ with $|u_0| \geq n$; clearly, $L$ contains such a word. As we have shown above, $|u_0| \leq |v_0|^3$. We now take $u_1 \leq_p \alpha$ with $|u_1| > |v_0|^4$; it follows that $u_1 a^m v_0 \notin L$, thus $u_0$ and $u_1$ are in different equivalence classes with respect to the syntactic congruence defined by the language $L$. However, by the considerations made at the beginning of this proof, there exists $v_1$ such that $u_1 a^m v_1 \in L$. In the exact same manner we construct a word $u_2$, that is in a different equivalence class with respect to the syntactic congruence defined by the language $L$ than both $u_0$ and $u_1$, and so on. This means we have an infinite sequence $(u_i)_{i \geq 0}$ where every two elements are in different equivalence classes with respect to the syntactic congruence defined by the language $L$. Since the syntactic congruence of $L$ has an infinite number of equivalence classes, $L$ is not regular. □

The following theorem follows immediately from the previous results.

**Theorem 21.** *Let $w = a^p y_1 a \cdots y_{r-1} a y_r$, where $a \notin alph(y_i)$ for $1 \leq i \leq r$.*

***(i)*** *If $\|alph(w)\| \geq 3$ and $|y_j| \geq 3$ for some $1 \leq j \leq r$, then for every positive integer $m \leq p$ we have that $\spadesuit^*_{(m,3)}(w)$ is not regular.*

***(ii)*** *If $\|alph(w)\| \geq 4$ and $|y_j| \geq 2$ for some $1 \leq j \leq r$, then for every positive integer $m \leq p$ we have that $\spadesuit^*_{(m,2)}(w)$ is not regular.*

***(iii)*** *If $\|alph(w)\| \geq 5$, then for every positive integer $m \leq p$ we have that $\spadesuit^*_{(m,1)}(w)$ is not regular.*

***(iv)*** *For all positive integers $m$ and $n$ there exists $u$ with $\spadesuit^*_{(m,n)}(u)$ not regular.*

**Proof.** By Theorems 18 and 19 we can take $q_2 = 6$ and $p_2 = 3$, respectively, $q_3 = 3$ and $p_3 = 2$. Therefore, if we take $n = 3$, or $n = 2$, respectively, in the hypothesis of the theorem, then the results $(i)$ and $(ii)$ follow for any positive $m \leq p$.

The third statement follows from [14] , where an infinite word avoiding both squares and palindromes is constructed. Thus, we can take $p_k = q_k = 1$, while $n$ can be also taken to be 1. Finally, $(iv)$ is a consequence of $(iii)$. □

In general, the regularity of the languages $♠^*_{(m,n)}(w)$ for positive integers $m$ and $n$, and binary words $w$, $|w| \geq n$, is *left open*. We only show the following.

**Theorem 22.** *For any word $w \in \{a,b\}^*$ and integer $m \geq 0$, $♠^*_{(m,1)}(w)$ is regular.*

**Proof.** For this proof, we take $x, y \in \{a,b\}$ with $x \neq y$, and denote $u^{-1}v = w$ if $v = uw$ or $u^{-1}v = v$, otherwise; similarly, we denote $vu^{-1} = w$ if $v = wu$ or $vu^{-1} = v$, otherwise. Moreover, $ux^{-1}v = u'v$ if $u = u'x$ and $ux^{-1}v = u(x^{-1}v)$, otherwise. This quotient operation $(\cdot)^{-1}$ can be easily extended to languages.

Our goal is to give a recursive definition of $♠^*_{(m,1)}(w)$. That is, the iterated inner $(m,1)$-palindromic closure of $w$ is expressed as a finite union and concatenation of languages $♠^*_{(m,1)}(w')$, where $|w'| < |w|$, and maybe some other regular languages. We first identify a series of basic cases for which such a definition can be given easily: words that have no maximal unary group longer than $m$, words of the form $xy^qx$, and, finally, words of the form $xy^q$ or $y^qx$.

We call propagating, the process through which from $xy$ we can generate any word $\{a,b\}^*xy\{a,b\}^*$ by iteratively applying inner $(m,1)$-palindromic closure operations. Similar to Lemma 14, we generate any prefix from left to right by applying left closure to $xy$ (to obtain $yxy$) or to $x$ (to obtain $xx$), and any suffix from right to left by applying right closure to $xy$ (to obtain $xyx$) or to $y$ (to obtain $yy$). Note that $|w| \geq 1$, as otherwise $♠^*_{(m,1)}(w) = \{\varepsilon\}$.

Thus, if $uvxy^pxv^R \in ♠^*_{(m,1)}(uvxy^q)$ for $q \leq p$, then we can derive $uvxy^pxv^R$ by first deriving $uvxy^px$ in one step, and then appending any suffix (in particular $v^R$) by propagation, and expanding the last group of $y$'s. Similar arguments hold for deriving prefixes, that is, when $v^Rxy^pxvu \in ♠_{(m,1)}(y^qxvu)$ for $q \leq p$.

**Lemma 23.** *If $\alpha = x^{\ell_1}y^{h_1}x^{\ell_2}y^{h_2}x^{\ell_3} \in ♠^*_{(m,1)}(xy^qx)$ for $\ell_1, \ell_2, \ell_3, h_1, h_2 > 0$, then exist $0 < p, r < q$ such that $\alpha \in ♠^*_{(m,1)}(xy^pxy^rx)$ and one of the next holds:*

- *$p \leq m$ or $r \leq m$ and $p = q - r$;*
- *$m < p, r$, and $p = m + 2k$ and $r = q - m - k$, or, vice-versa, $r = m + 2k$ and $p = q - m - k$, for some $k > 0$.*

**Proof.** As $\alpha = x^{\ell_1}y^{h_1}x^{\ell_2}y^{h_2}x^{\ell_3} \in ♠^*_{(m,1)}(xy^qx)$, there exists a derivation of $\alpha$ from $xy^qx$. The first steps of this derivation includes $x^{e_1}y^fx^{e_3}$ obtained by simply pumping $x$'s and $y$'s. It also includes the one closure step where the group of $y$'s is split and the word $xy^pxy^qx$ is obtained. Further, $\alpha$ is derived be expanding the $x$'s and the $y$'s. It is very important to note that $h_1 + h_2 \geq p + r \geq q$.

If $q \leq 2m+1$, then we derive in one step $xy^pxy^{q-p}x$, or $xy^{q-r}xy^rx$, from $xy^qx$, and then pump the $y$'s until we get $xy^{h_1}xy^{h_2}x$, and we are within the first condition of the Lemma. Thus we must have $q > 2m+1$. If $h_1 \leq m$, then we obtain $xy^pxy^{q-p}x$ from $xy^qx$ in one step, and then derive $xy^{h_1}xy^{h_2}x$ by pumping $y$'s. In this case, we just take $p = h_1$ and $r = q - p$. Similarly, if $h_2 \leq m$, then we obtain $xy^{q-r}xy^rx$ from $xy^qx$ in one step, and then derive $xy^{h_1}xy^{h_2}x$ by pumping $y$'s. We take $r = h_2$ and $p = q - r$. Therefore, we can assume that $h_1, h_2 > m$.

By the definition of the $(m,1)$-palindromic closure, in the case left we know that $p = m + 2k$ with $k > 0$ and $r = q - m - k > 0$, or $r = m + 2k$ with $k > 0$ and $p = q - m - k > 0$. Observe that now, we must have that either $m + 2k \leq p < 2q - m$ for some $k$, or $m + 2k \leq r < 2q - m$ for some $k$. □

In other words, Lemma 23 provides us with a normal-form-derivation for any word $x^{\ell_1}y^{h_1}x^{\ell_2}y^{h_2}x^{\ell_3}$: we find the numbers $p$ and $r$, and in the first step we derive $xy^pxy^rx$, and then we continue.

We can now proceed towards a recursive definition of $♠^*_{(m,1)}(w)$:

**A)** If $w$ has no maximal unary factor longer than $m$, then $♠^*_{(m,1)}(w)$ contains all words with $w$ as scattered factor (by propagation with reasons similar to Lemma 14).

**B)** If $w = xy^qx$ with $q > m$, then $♠^*_{(m,1)}(xy^qx)$ is the union of:
– $\{a,b\}^*(x^hy^px^\ell)\{a,b\}^*$ with $h, \ell \geq 1$ and $p \geq q$, and
– $♠^*_{(m,1)}(xy^px)x^{-1}♠^*_{(m,1)}(xy^rx)$ with $0 < p < m$ or $0 < r < m$ and $r = q - p$, and
– $♠^*_{(m,1)}(xy^px)x^{-1}♠^*_{(m,1)}(xy^rx)$ with $p = m + 2k$ and $r = q - m - k$, or, vice-versa, $r = m + 2k$ and $p = q - m - k$, and $p, r < q$.

We show the equality of $♠^*_{(m,1)}(xy^qx)$ with the respective union of the sets by double inclusion. However, the fact that the generated language includes the union of the three sets is a direct consequence of Lemma 23. To show the other inclusion, note that the first set covers the case when we generate an arbitrary prefix (by propagation to the left from $xy$) and an arbitrary suffix (by propagation to the right from $yx$), and then pump an arbitrary number of $x$'s and $y$'s in the corresponding groups; no other operations are performed in this case. Moreover, if a word is obtained in another way from $xy^qx$, then, by the remarks made at the beginning of the proof and in Lemma 23, we first generate a word $xy^pxy^rx$, where $p$ and $r$ fall in one of the two cases described in the statement of the recursive formula, and the conclusion follows once again.

It now easily follows that $♠^*_{(m,1)}(xy^qx)$ is regular for any $x$, $y$, and $q$: as $p, r < q$, $♠^*_{(m,1)}(xy^qx)$ is the union of the iterated inner $(m,1)$-palindromic closure of shorter words of the same form. Applying recursively the same formula further simplifies these sets, until we reach only closures of words that have no unary group longer than $m$, and we can apply the first case, described above.

**C)** If $w = xy^q$ with $q > m$, then $♠^*_{(m,1)}(xy^q)$ is the union of:
– $\{a,b\}^*(x^hy^p)$ with $h \geq 1$ and $p \geq q$, and
– $♠^*_{(m,1)}(xy^px)x^{-1}♠^*_{(m,1)}(xy^r)$ with $p \geq m$ and $r = q - m$, or $p = m + 2k$, $r = q - m - k$, and $k, r < q$.

The proof of the equality of the union of sets and $♠^*_{(m,1)}(xy^q)$ follows the proof of the case above. The only particular observation is that computing $♠^*_{(m,1)}(xy^q)$ implies computing the iterated inner $(m,1)$-palindromic closure of a (potentially) longer word $xy^px$. However, we know from the previous case that this is regular, and the regularity of $♠^*_{(m,1)}(xy^q)$ follows just like in the previous case.

**D)** If $w = y^qx$ with $q > m$, then $♠^*_{(m,1)}(y^qx)$ is the union of:

– $(y^p x^h)\{a,b\}^*$ with $h \geq 1$ and $p \geq q$, and
– $♠^*_{(m,1)}(y^r x)x^{-1}♠^*_{(m,1)}(xy^p x)$ with $p \geq m$ and $r = q-m$, or $p = m+2k$, $r = q-m-k$, and $k, r < q$.

Again, the proof is similar to the above, and symmetrical to case **C)**.

**E)** Finally, for $w = x^{h_1}y^{p_1}w'$ with $h_1, p_1 > 0$ and $w'$ not starting with $y$ we have:
– If $h_1 > 1$ then $♠^*_{(m,1)}(w) = (L_0 \cup \{x^j \mid j \geq h_1 - 1\})♠^*_{(m,1)}(xy^{p_1}w') \cup (L_0 \setminus \{x\}^*)♠^*_{(m,1)}(xy^{p_1-1}w')$, where $L_0 = ♠^*_{(m,1)}(x^{h_1}y)y^{-1}x^{-1}$.
– If $h_1 = 1$ and $w' = xw''$ then $♠^*_{(m,1)}(w) = (L_0 \cup \{xy^j \mid j \geq p_1 - 1\})♠^*_{(m,1)}(yw') \cup (L_0 \setminus x\{y\}^*)♠^*_{(m,1)}(yw'')$, where $L_0 = ♠^*_{(m,1)}(xy^{p_1}x)x^{-1}y^{-1}$.
– If $h_1 = 1$ and $w' = \varepsilon$ then we are in one of the basic cases, described above.

Let us show the first case. Using the first remark of this proof, we can simulate each derivation of a word from $w$ by deriving first a word $uy^{p_1-1}w'$, where $u \in ♠^*_{(m,1)}(x^{h_1}y)$. If $u = u'xy$, then we have derived $u'xy^{p_1}w'$ with $u' \in L_0 \cup \{x^j \mid j \geq h_1 - 1\}$, and we can continue our derivation by deriving the suffix $xy^{p_1}w'$. If $u = u'xy^\ell$ with $\ell > 1$, it is straightforward that we can derive from $x^{h_1}y$ the word $u'xy$ as well. So we first generate $u'xyy^{p_1-1}w'$, then we pump $\ell$ symbols $y$ in group of $y$'s that occurs after $u'x$, and get $u'$ followed by a word from $♠^*_{(m,1)}(xy^{p_1}w')$; again, we derive a word from $(L_0 \cup \{x^j \mid j \geq h_1 - 1\})♠^*_{(m,1)}(xy^{p_1}w')$. If $u = u'x$, then $u'$ is in $L_0$ and contains at least one $y$, so we can derive $u'xy^{p_1-1}w'$, and further continue by applying operations to $xy^{p_1-1}w'$.

For the second inclusion, we take $u \in L_0 \cup \{x^j \mid j \geq h_1 - 1\}$ and $v \in ♠^*_{(m,1)}(xy^{p_1}w')$. Now, we can clearly generate from $w$ the word $uxyy^{p_1-1}w' = uxy^{p_1}w'$; thus, we can also generate the word $uv$. Similarly we can also show that $(L_0 \setminus \{x\}^*)♠^*_{(m,1)}(xy^{p_1-1}w') \subseteq ♠^*_{(m,1)}(w)$.

The second case follows similarly.

As mentioned, these recursive definitions show that $♠^*_{(m,1)}(xy^q x)$, $♠^*_{(m,1)}(xy^q)$, and $♠^*_{(m,1)}(y^q x)$ are finite unions and concatenations of regular languages, thus regular. Following case **E)**, $♠^*_{(m,1)}(w)$ is also regular, for all binary words $w$. □

## 5. Final remarks

Apart from solving the open problems stated in this article, the study of classes of languages obtained through these operations seems interesting to us. The following initial results show several possible directions for such investigations.

For a class $\mathcal{L}$ of languages, we set $\mathcal{L}^R = \{L^R \mid L \in \mathcal{L}\}$, and for an integer $k \geq 1$, we define $\mathcal{L}_k = \{L \in \mathcal{L} \mid \|\text{alph}(L)\| = k\}$. Consider the classes

$$\mathcal{P}_{♠_\ell} = \{L' \mid L' = ♠^*_\ell(L) \text{ for some } L\}$$
$$\mathcal{P}_{♠_r} = \{L' \mid L' = ♠^*_r(L) \text{ for some } L\}$$
$$\mathcal{P}_{♠} = \{L' \mid L' = ♠^*(L) \text{ for some } L\}$$

Straightforward, for every language $L$, $♠_r(L) = (♠_\ell(L^R))^R$ and $♠^*_r(L) =$

$(♠_\ell^*(L^R))^R$ hold (for both operations the propagation rule works in only one direction). Thus, we immediately get $\mathcal{P}_{♠_r} = (\mathcal{P}_{♠_\ell})^R$ and $\mathcal{P}_{♠_\ell} = (\mathcal{P}_{♠_r})^R$.

The following result is not difficult:

**Lemma 24.** *The classes $\mathcal{P}_{♠_r} \setminus \mathcal{P}_{♠}$ and $\mathcal{P}_{♠_\ell} \setminus \mathcal{P}_{♠}$ are both not empty.*

When we consider only binary alphabets, we have the following statement.

**Proposition 25.** $(\mathcal{P}_{♠})_2 \subsetneq (\mathcal{P}_{♠_r})_2 = (\mathcal{P}_{♠_\ell})_2^R$ *and* $(\mathcal{P}_{♠})_2 \subsetneq (\mathcal{P}_{♠_\ell})_2 = (\mathcal{P}_{♠_r})_2^R$.

## References

[1] D. P. Bovet and S. Varricchio, On the regularity of languages on a binary alphabet generated by copying systems, *Information Processing Letters* **44** (1992) 119–123.
[2] D. Cheptea, C. Martín-Vide and V. Mitrana, A new operation on words suggested by DNA biochemistry: Hairpin completion, *Transgressive Computing* (2006) 216–228.
[3] J. Dassow and M. Holzer, Language families defined by a ciliate bio-operation: hierarchies and decision problems, *International Journal of Foundations of Computer Science* **16**(4) (2005) 645–662.
[4] J. Dassow, V. Mitrana and G. Păun, On the regularity of duplication closure, *Bulletin of the EATCS* **69** (1999) 133–136.
[5] J. Dassow, V. Mitrana and A. Salomaa, Context-free evolutionary grammars and the structural language of nucleic acids, *BioSystems* **43** (1997) 169–177.
[6] A. de Luca, Sturmian words: Structure, combinatorics, and their arithmetics, *Theoretical Computer Science* **183** (1997) 45–82.
[7] A. Ehrenfeucht and G. Rozenberg, On regularity of languages generated by copying systems, *Discrete Applied Mathematics* **8** (1984) 313–317.
[8] S. Z. Fazekas, R. Mercaş and K. Shikishima-Tsuji, Hairpin completion with bounded stem-loop, *DLT 2012*, *Lecture Notes in Computer Science* **7410** (2012), pp. 428–43.
[9] G. Higman, Ordering by divisibility in abstract algebras, *Proceedings of the London Mathematical Society* **3**(2) (1952) 326–336.
[10] L. Kari, S. Konstantinides, E. Losseva, P. Sosik and G. Thierrin, Hairpin structures in DNA words, *DNA*, *Lecture Notes in Computer Science* **3892** (2006), pp. 158–170.
[11] L. Kari and K. Mahalingam, Watson–Crick palindromes in DNA computing, *Natural Computing* **9**(2) (2010) 297–316.
[12] P. Leupold, Languages generated by iterated idempotencies, PhD thesis, Universitat Rovira y Virgili (2006).
[13] P. Leupold and V. Mitrana, Uniformly bounded duplication codes, *RAIRO – Theoretical Informatics and Applications* **41** (2007) 411–427.
[14] J.-J. Pansiot, A propos d'une conjecture de F. Dejean sur les répétitions dans les mots, *Discrete Applied Mathematics* **7** (1984) 297–311.
[15] N. Rampersad, J. Shallit and M.-W. Wang, Avoiding large squares in infinite binary words, *Theoretical Computer Science* **339**(1) (2005) 19–34.
[16] G. Rozenberg and A. Salomaa, *Handbook of Formal Languages* (Springer-Verlag New York, Inc., 1997).
[17] D. Searls, The computational linguistics of biological sequences, *Artificial Intelligence and Molecular Biology*, (AAAI Press, Cambridge, 1993), pp. 47–120.
[18] M.-W. Wang, On the irregularity of the duplication closure, *Bulletin of the EATCS* **70** (2000) 162–163.