RAIRO-Inf. Theor. Appl. **38** (2004) 189-228
DOI: 10.1051/ita:2004010

# CIRCULAR SPLICING AND REGULARITY *

Paola Bonizzoni[1], Clelia De Felice[2], Giancarlo Mauri[1] and Rosalba Zizza[2]

**Abstract.** *Circular splicing* has been very recently introduced to model a specific recombinant behaviour of circular DNA, continuing the investigation initiated with linear splicing. In this paper we restrict our study to the relationship between *regular circular languages* and languages generated by *finite circular splicing systems* and provide some results towards a characterization of the intersection between these two classes. We consider the class of languages $X^*$, called here *star languages*, which are closed under conjugacy relation and with $X$ being a regular language. Using automata theory and combinatorial techniques on words, we show that for a subclass of star languages the corresponding circular languages are (Paun) *circular splicing languages*. For example, star languages belong to this subclass when $X^*$ is a free monoid or $X$ is a finite set. We also prove that each (Paun) circular splicing language $L$ over a one-letter alphabet has the form $L = X^+ \cup Y$, with $X, Y$ finite sets satisfying particular hypotheses. *Cyclic* and *weak cyclic languages*, which will be introduced in this paper, show that this result does not hold when we increase the size of alphabets, even if we restrict ourselves to regular languages.

**Mathematics Subject Classification.** 68Q05, 68Q45, 68Q70.

*Keywords and phrases.* Molecular computing, splicing systems, formal languages, automata theory, variable-length codes.

* *Partially supported by MIUR Project "Linguaggi Formali e Automi: teoria ed applicazioni", by the contribution of EU Commission under The Fifth Framework Programme (project MolCoNet IST-2001-32008) and by* 60% *Project "Linguaggi Formali e Modelli di Calcolo" (University of Salerno).*

[1] Dipartimento Informatica Sistemistica e Comunicazione, Università degli Studi di Milano, Bicocca, Via Bicocca degli Arcimboldi 8, 20126 Milano, Italy;
e-mail: `bonizzoni@disco.unimib.it; mauri@disco.unimib.it`

[2] Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84081 Baronissi (SA), Italy; e-mail: `defelice@dia.unisa.it; zizza@dia.unisa.it`

## 1. Introduction

Requests for new applications from the world of computing and technology are stimulating interest towards computational models inspired by biological mechanisms. Molecular computing is one of the research directions in this framework and one of its highlights is in experiments which have produced efficient algorithms for solving typical NP-complete problems, underlining the computational power of biological phenomena (see Adleman's experiment and Lipton's experiment among them [25]).

In the wake of these discoveries, molecular computing has been developing in several directions which are both technological and theoretical. In particular, one research direction which is strictly related to formal languages theory, is based on the notion of *splicing systems*. This aspect will be considered here. The notion of splicing systems was first introduced in [17], where Head modelled a recombinant behaviour of DNA molecules (under the action of restriction and ligase enzymes) as a particular operation between words in a formal language thus suggesting the possibility of using molecules to perform computations. In short, a restriction enzyme is able to recognize a pattern in a DNA molecule and cut the molecule inside the pattern in a specific position, thus providing two segments of DNA. Then a ligase enzyme binds together pairs of sequences from two molecules into a new one. In order to treat this cut and paste phenomenon in formal language theory, an initial set $I$ of strings (the initial set of DNA molecules) and a set $R$ of splicing rules or special words (simulators of enzyme behaviour) are given, such that the set of all possible molecules generated by the biochemical process of cut and paste is the language of all possible words that may be generated by applying the rules to the strings. We deliberately ignore other molecular considerations and basic details (for a complete monograph see [25]).

DNA occurs in both linear and circular form and, correspondingly, linear and *circular splicing systems* have been defined. There are three definitions of (linear) splicing systems given by Head, Paun and Pixton. The computational power of these systems, *i.e.* the class of languages which are generated by them, has been described. This computational power also depends on which level in the Chomsky hierarchy the initial set $I$ and the set of the rules $R$ belong to. Under some hypotheses on $I, R$, splicing systems can reach the same power of Turing machines [19,24]. At the opposite end of the hierarchy, when we restrict ourselves to splicing systems with a finite set $R$ of rules and a finite set $I$ of strings (*finite linear splicing systems*), we get a proper subclass of regular languages, as shown in [13,16,22,26]. This class of languages has not been yet completely characterized even if partial results are known [7,8]. However, the size of this family of regular languages increases when we substitute Head's systems with Paun's systems and Paun's systems with Pixton's systems [11].

On the contrary, few results are known about circular splicing systems. In Nature circular splicing occurs in a recombinant mechanism (transposition) between bacteria and plasmids. Moreover, the replication is possible only if DNA has a circular form in the host cell. Furthermore, it could be interesting to use circular

DNA in Adleman's experiment, since plasmids replicate themselves without errors or exponential weight increase, which are weak points in Adleman's approach [1]. In fact, in [18, 20] a new method of computing, applicable to a wide variety of algorithmic problems, has been introduced by using DNA plasmids: a circular DNA plasmid is specifically constructed step by step, using a scheme of enzymatic treatments.

In this paper we deal with circular splicing systems even if we provide interesting links with the linear case, deepening the observations made in [28]. Furthermore, in our theoretical model there is no hypothesis on the number of copies of each molecule in the initial set.

Classical results in formal language theory and combinatorial tools have been of great use when obtaining our results. Indeed, circular splicing systems deal with *circular languages* the elements of which are equivalence classes under the conjugacy relation (*circular words*). It is worthy of note that this notion plays an important role in combinatorics on words as well as in some language factorization problems (*e.g.* the problem of finding factorizations of free monoids, see [2, 23]).

As in the linear case, three definitions of the circular splicing operation have been given (by Head, Paun and Pixton) along with a counterpart of the notion of regular language, namely the notion of *regular circular language*. Here we give some results towards a characterization of those regular circular languages which are generated by finite circular splicing systems.

In order to be more precise, given a *star language* (*i.e.* a language $X^*$ closed under conjugacy relation and with $X$ a regular language) and considering a supplementary hypothesis, we prove that the corresponding circular language is generated by a finite Paun circular splicing system (Th. 6.1). This supplementary hypothesis deals with special labels (*cycles*) of closed paths in the transition diagram of a finite state automaton recognizing $X^*$, along with simple results concerning these labels which have been proved in [9] and recalled here. Furthermore, the abovementioned result concerning star languages has a relationship with the theory of variable-length codes [2]. Indeed, we prove that for all star languages $X^*$ when $X$ is a finite set or a *rational code*, the corresponding circular languages are regular circular languages generated by finite Paun circular splicing systems (Prop. 6.3 and Cor. 6.1).

In contrast with the linear case, a finite initial set and a finite set of rules do not guarantee that for the language $L$ generated by the corresponding circular splicing system, we have that $L$ is a regular circular language. However, this happens for languages over a one-letter alphabet and in Proposition 7.1 we also prove that in this case $L = X^+ \cup Y$, with $X, Y$ finite sets satisfying simple hypotheses which make subgroups of cyclic groups intervene (notice that in the special case of a one-letter alphabet, we can identify each language with the corresponding circular language). The situation is much more complicated for alphabets of larger size – even if we restrict ourselves to regular languages. Indeed, there exist regular languages which are not the Kleene closure of a regular language and such that the corresponding circular languages are generated by finite Paun circular splicing systems. Examples of these languages are *cyclic* and *weak cyclic* languages which are introduced in Section 8 (Prop. 8.3).

Let us briefly sketch the contents of this paper. In Section 2 we gathered some known results and other easy new results on circular languages (Sect. 2.1) and splicing (Sect. 2.2). Differences between linear and circular splicing have been underlined in Section 2.3 whereas related notions have been collected in Section 2.4. As far as we know, very few results for circular splicing systems are known, and these are surveyed in Section 2.5, along with several additional hypotheses which might or might not be added to circular splicing systems. In Section 3 we explicitly observe that the computational power of the three circular splicing systems is different. Some questions are also asked. In Section 4, we introduce the general problem we are attacking. Section 5 is devoted to definitions and results concerning cycles. The last three sections in this paper are dedicated to our main results which are given with respect to the Paun definition (with no additional hypotheses). However, digressions on Pixton's systems are also given. In Section 6 we prove our result for star languages. The case of languages over a one-letter alphabet is investigated in Section 7. In Section 8 we introduce cyclic and weak cyclic languages. A preliminary shorter version of some of the results gathered in Sections 2–4 and 6 in this paper was presented at DNA6 [5]. In addition, Theorem 6.1 (reported in Sect. 6 with a correct proof) is a generalization of Theorem 1 presented (with a wrong proof) in the same paper [5].

## 2. Languages and splicing: definitions and properties

In this section we give the necessary definitions and results concerning languages and splicing systems which will be used in the next part of this paper. Circular languages will be considered in Section 2.1 whereas circular splicing systems will be examined in Sections 2.2–2.4. Additional hypotheses which are introduced in literature are reported in Section 2.5 but will not be taken into account in this paper.

Let us also fix some notations. We denote by $A^*$ the free monoid over a finite alphabet $A$ and by $|w|$ the length of $w \in A^*$. For every $a \in A$, $w \in A^*$, we denote by $|w|_a$ the number of occurrences of $a$ in $w$, whereas for a subset $X$ of $A^*$, we denote by $|X|$ the cardinality of $X$. We also set $A^+ = A^* \setminus 1$, where 1 is the empty word. A word $x \in A^*$ is a *factor* of $w \in A^*$ if $u_1, u_2 \in A^*$ exist such that $w = u_1 x u_2$ and $x$ is a *proper factor* of $w$ if $u_1 u_2 \neq 1$. Furthermore, $x$ is a *prefix* (resp. *suffix*) of $w$ if $u_1 = 1$ (resp. $u_2 = 1$) and $x$ is a *proper prefix* (resp. *proper suffix*) of $w$ if $u_2 \neq 1 = u_1$ (resp. $u_1 \neq 1 = u_2$). In addition, for each $w \in A^*$, we set $alph(w) = \{a \in A \mid a \text{ is a factor of } w\}$.

We also use the following notations and we suppose the reader is familiar with elementary finite state automata theory (see [2,21]). Let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a finite state automaton, where $Q$ is a finite set of states, $q_0 \in Q$ is the initial state and $F \subseteq Q$ is the set of final states. The transition function $\delta$ is defined in the classical way [2,21]. $\mathcal{A}$ is *deterministic* if, for each $q \in Q$, $a \in A$, there exists at most one state $q' \in Q$ such that $\delta(q,a) = q'$. In addition, $\mathcal{A}$ is *trim* if each state is accessible and coaccessible, *i.e.*, if for each state $q \in Q$ there exist $x, y \in A^*$ such that $\delta(q_0, x) = q$ and $\delta(q, y) \in F$. As usual, in the transition diagram of a trim deterministic automaton $\mathcal{A}$, each final state will be indicated by a double circle and the initial state will be indicated by an arrow without a label going into it. A *successful path* is a path in the transition diagram of $\mathcal{A}$ going from the initial state to a final state.

Let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a trim deterministic finite state automaton, let $a_1, \ldots, a_n, q, q'$ be such that $a_1, \ldots, a_n \in A$, $q, q' \in Q$, $\delta(q, a_1 \cdots a_n) = q'$. We say that $q$ and $\delta(q, a_1 \cdots a_i)$, $1 \leq i \leq n$, are the *states crossed* by the transition $\delta(q, a_1 \cdots a_n) = q'$ and, for each $i \in \{1, \ldots, n-1\}$, $\delta(q, a_1 \cdots a_i)$ is an *internal state crossed* by the same transition. Given a regular language $L \subseteq A^*$, it is well known that there exists a *minimal* finite state automaton $\mathcal{A}$ recognizing it, *i.e.*, such that $L = L(\mathcal{A})$. This canonical automaton is determined uniquely up to a renaming of the states and (when we make it trim) it has the minimal number of states. In this paper, for a regular language $L$, $\mathcal{A} = (Q, A, \delta, q_0, F)$ will be the minimal automaton recognizing it. Regular languages will be also represented at times by means of regular expressions.

### 2.1. Circular languages

As we have already said in Section 1, circular words have been intensively examined in formal language theory. For a given word $w \in A^*$, a circular word ${}^\sim w$ is the equivalence class of $w$ with respect to the *conjugacy* relation $\sim$ defined by $xy \sim yx$, for $x, y \in A^*$ [23]. For every $a \in A$, the notations $|{}^\sim w|$, $|{}^\sim w|_a$ and $alph({}^\sim w)$ will be defined respectively as $|w|$, $|w|_a$ and $alph(w)$, for any representative $w$ of ${}^\sim w$. Furthermore, when the context does not make it ambiguous, we will use the notation $w$ for a circular word ${}^\sim w$. ${}^\sim A^*$ is the set of all circular words over $A$, *i.e.* the quotient of $A^*$ with respect to $\sim$. Given $L \subseteq A^*$, ${}^\sim L = \{{}^\sim w \mid w \in L\}$ is the *circularization* of $L$, *i.e.* the set of all circular words corresponding to elements of $L$, while every language $L$ such that ${}^\sim L = C$, for a given circular language $C \subseteq {}^\sim A^*$, is called a *linearization* of $C$. The *full linearization* $Lin({}^\sim L)$ of ${}^\sim L$ is the set of all the strings in $A^*$ corresponding to the elements of ${}^\sim L$, *i.e.*, $Lin({}^\sim L) = \{w' \in A^* \mid \exists w \in L : \ w' \sim w\}$. In the next part of this paper, for all $X_1, \ldots, X_n \subseteq A^*$, we set ${}^\sim(X_1 \cdots X_n) = {}^\sim X_1 \cdots X_n$.

**Example 2.1.** For $w = abbaa$, we have $Lin({}^\sim abbaa) = \{abbaa, bbaaa, baaab, aaabb, aabba\}$. Every non empty subset of $Lin({}^\sim abbaa)$ is a linearization of ${}^\sim abbaa$. The circularization of $\{abbaa, bbaaa\}$ is ${}^\sim abbaa$.

Circular splicing deals with circular strings and circular languages and as a result, with formal languages which are full linearizations of circular languages. The following proposition states an obvious property of full linearizations. It is reported here for the sake of completeness.

**Proposition 2.1.** *A language $L$ is the full linearization $L = Lin(C)$ of a circular language $C$ if and only if $L$ is closed under the conjugacy relation, i.e., for all $l, l' \in A^*$ if $l \in L$ and $l' \in\ {}^{\sim}l$, we have $l' \in L$.*

*Proof.* Let $L = Lin(C) = \{w' \in A^* \mid \exists\ {}^{\sim}w \in C : \ w' \sim w\}$ be the full linearization of a circular language $C$. Let $l \in L$. Then, ${}^{\sim}l \in C$ and, by using the definition of full linearization, for every $l' \in\ {}^{\sim}l$, we have $l' \in L$. Then, $L$ is closed under the conjugacy relation. Conversely, $L \subseteq Lin({}^{\sim}L)$ and, if $L$ is closed under the conjugacy relation, then it holds $Lin({}^{\sim}L) = L$. □

In the next part of this paper, we will consider languages closed under the conjugacy relation and the action of circular splicing over their circularization. We denote by $Fin$ (resp. $Reg$) the class of finite (resp. regular) languages over $A$. Given a family of languages $FA$ in the Chomsky hierarchy, $FA^{\sim}$ consists of all those circular languages $C$ which have some linearization in $FA$. In this paper we deal only with circular languages having a regular linearization, *i.e.*, with $Reg^{\sim} = \{C \subseteq {}^{\sim}A^* \mid \exists L \in Reg : \ {}^{\sim}L = C\}$. It is classically known that given a regular language $L \subseteq A^*$, $Lin({}^{\sim}L)$ is regular (see Ex. 4.2.11 in [21] where $Lin({}^{\sim}L)$ is called $CYCLE(L)$). As a result, given a circular language $C$, we have $C \in Reg^{\sim}$ if and only if its full linearization $Lin(C)$ is regular. If $C \in Reg^{\sim}$ then $C$ is a *regular circular language*. Analogously, we can define context-free (resp. context-sensitive, recursive, recursively enumerable) circular languages.

## 2.2. Circular splicing

The splicing operation in the circular case deals with biological phenomena which are different from the ones modelled by the linear case, but as well as for linear splicing, three definitions of circular splicing are known and are recalled below.

`Head's definition` [28]. A *Head circular splicing system* is a quadruple $SC_H = (A, I, T, P)$, where $I \subseteq {}^{\sim}A^*$ is the initial circular language, $T \subseteq A^* \times A^* \times A^*$ and $P$ is a binary relation on $T$ such that, for each $(p, x, q)$,$(u, y, v) \in T$, if $(p, x, q)P(u, y, v)$ then $x = y$. Thus, given two circular words ${}^{\sim}hpxq$, ${}^{\sim}kuxv \in {}^{\sim}A^*$ with $(p, x, q), (u, x, v) \in T$ and $(p, x, q)P(u, x, v)$, the splicing operation is defined to produce ${}^{\sim}hpxvkuxq$.

`Paun's definition` [24]. A *Paun circular splicing system* is a triplet $SC_{PA} = (A, I, R)$, where $I \subseteq {}^{\sim}A^*$ is the initial circular language and $R \subseteq A^*|A^*\ \$A^*|A^*$, with $|, \$ \notin A$, is the set of rules. Then, given a rule $r = u_1|u_2\$u_3|u_4$ and two circular words ${}^{\sim}hu_1u_2$,${}^{\sim}ku_3u_4$, the splicing by the rule is defined to cut and linearize the two strings obtaining $u_2hu_1$ and $u_4ku_3$, and to paste and circularize them obtaining ${}^{\sim}u_2hu_1u_4ku_3$.

`Pixton's definition` [26]. A *Pixton circular splicing system* is a triplet $SC_{PI} = (A, I, R)$ where $A$ is a finite alphabet, $I \subseteq {}^{\sim}A^*$ is the initial circular language, $R \subseteq A^* \times A^* \times A^*$ is the set of rules and if $r = (\alpha, \alpha'; \beta) \in R$ then there exists $\beta'$ such that $\overline{r} = (\alpha', \alpha; \beta') \in R$. Thus, given two circular words ${}^{\sim}\alpha\epsilon$,${}^{\sim}\alpha'\epsilon'$, the splicing by the two rules $r, \overline{r}$ is defined to cut and linearize the two strings, obtaining $\epsilon\alpha, \epsilon'\alpha'$, and then to paste, substitute and circularize them, producing $^{\sim}\epsilon\beta\epsilon'\beta'$. Any pair of rules $(r, \overline{r})$ of the given form may be used.

**Remark 2.1.** We notice that in a Pixton circular splicing system, for $r, \overline{r} \in R$, we can have $r = \overline{r}$. Furthermore, for a rule $r = (\alpha, \alpha'; \beta) \in R$, we could have different rules $\overline{r}_i = (\alpha', \alpha; \beta'_i) \in R$ such that the pair $(r, \overline{r}_i)$ may be used. Nevertheless, we will adopt the notation $(r, \overline{r})$ since the context will not make it ambiguous. Symmetrically, for given rules $r_i = (\alpha, \alpha'; \beta_i) \in R$, $1 \leq i \leq v$, the same rule $\overline{r}_i = \overline{r} = (\alpha', \alpha; \beta') \in R$ can be associated with all the $r_i$'s. An example of this situation is given in Example 3.2.

**Remark 2.2.** As usual, we restrict ourselves to circular splicing systems with a finite or regular initial circular language. Furthermore, it goes without saying that the sets $T$ and $R$ are finite in the three definitions of circular splicing given above.

**Remark 2.3.** We must note that in the original definition of circular splicing language given by Paun in [19], rules in $R$ can be used in two different ways: one way has been described above, the other, called *self-splicing*, will be defined in Section 2.5. In this paper, in order to make the three definitions uniform, we have preferred to omit self-splicing from Paun's definition.

We now give the definition of circular splicing languages. For a given splicing system $SC_X$, with $X \in \{H, PA, PI\}$, we denote $(w', w'')\vdash_r w$ the fact that $w$ is produced from (or spliced by) $w', w''$ by using a rule $r$. Furthermore, given a language $C \subseteq {}^{\sim}A^*$, we denote $\sigma'(C) = \{z \in {}^{\sim}A^* \mid \exists w', w'' \in C, \exists r \in R.\ (w', w'')\vdash_r z\}$. Thus, we define $\sigma^0(C) = C$, $\sigma^{i+1}(C) = \sigma^i(C) \cup \sigma'(\sigma^i(C))$, $i \geq 0$, and then $\sigma^*(C) = \bigcup_{i \geq 0} \sigma^i(C)$. We explicitly note that for Pixton's systems, the splicing operation is the combined action of a pair of rules, $r$ and $\bar{r}$, and we will use the notation $\vdash_{r,\bar{r}}$ to make this behaviour evident.

**Definition 2.1.** Given a splicing system $SC_X$, with $X \in \{H, PA, PI\}$, the circular language $C(SC_X) = \sigma^*(I)$ is the language generated by the system $SC_X$, $I$ being the initial circular language in $SC_X$.
A circular language $C$ is $C_X$ generated (or $C$ is a circular splicing language) if a splicing system $SC_X$ exists such that $C = C(SC_X)$.

**Example 2.2.** In [28] it is shown that $\{^{\sim}(aa)^n | n \geq 0\}$ is $C_H$ generated by $SC_H = (A, I, T, P)$ with $A = \{a\}$, $I = \{^{\sim}1, ^{\sim}aa\}$, $T = \{(1, a, 1)\}$, and $(1, a, 1)P(1, a, 1)$. We can observe that $\{^{\sim}(aa)^n | n \geq 0\}$ is also $C_{PA}$ generated, by choosing $SC_{PA} = (A, I, R)$, with $A = \{a\}, I = \{^{\sim}aa\} \cup 1, R = \{aa|1\$1|aa\}$, as shown in Section 7. Finally, in Example 3.1 we prove that $C = {}^{\sim}(aa)^*b$ is $C_{PI}$ generated by choosing $SC_{PI} = (A, I, R)$, with $A = \{a, b\}, I = \{^{\sim}b, ^{\sim}a^2b, ^{\sim}a^4b\}, R = \{(a^2, a^2b; a^2), (a^2b, a^2; 1)\}$.

We end this section with Lemma 2.1 which shows that in order to characterize the class of the $C_{PA}$ generated languages $C$, we can limit ourselves to the $C_{PA}$ generated languages $C$ with $1 \notin C$, *i.e.*, to the circular splicing systems

$SC_{PA} = (A, I, R)$ with $1 \notin I$. The proof is rather simple and reported below only for the sake of completeness.

**Lemma 2.1.** *Let $SC_{PA} = (A, I, R)$ be a circular splicing system. Then, the following conditions hold.*

1) *For each $w \in {}^{\sim}A^*$, $r \in R$, if $r$ can be applied to the pair $(w, 1)$ (resp. $(1, w)$), then $(w, 1) \vdash_r w$ (resp. $(1, w) \vdash_r w$).*
2) *For each $w_1, w_2 \in {}^{\sim}A^*$, $r \in R$, if $(w_1, w_2) \vdash_r 1$ then $w_1 = w_2 = 1$.*
3) *Let $C = C(SC_{PA})$. Then $1 \in C$ if and only if $1 \in I$.*
4) *If $C = C(SC_{PA})$ and $1 \in C$, then $C \setminus \{1\}$ is a $C_{PA}$ generated language.*
5) *If $C = C(SC_{PA})$ and $1 \notin C$, then $C \cup \{1\}$ is a $C_{PA}$ generated language.*

*Proof.* 1) Let $w \in {}^{\sim}A^*$, $r = u_1|u_2\$u_3|u_4 \in R$ be such that $r$ can be applied to the pair $(w, 1)$ and let $w_1 \in {}^{\sim}A^*$ be such that $(w, 1) \vdash_r w_1$. Then, looking at the definition of the circular splicing operation, there exist $h, k \in A^*$ such that ${}^{\sim}hu_1u_2 = w$, ${}^{\sim}ku_3u_4 = 1$ and ${}^{\sim}u_2hu_1u_4ku_3 = w_1$. As a direct result we have $k = u_3 = u_4 = 1$ and so $w = w_1$. A similar argument allows us to conclude that if $r$ can be applied to the pair $(1, w)$ and $(1, w) \vdash_r w_1$ then $w = w_1$.

2) As a preliminary observation, we notice that, by looking at the definition of the circular splicing operation, for each circular splicing system $SC_{PA} = (A, I, R)$, for each $w_1, w_2 \in {}^{\sim}A^*$, $r \in R$, if $(w_1, w_2) \vdash_r w$ then $alph(w) = alph(w_1) \cup alph(w_2)$. Consequently, if $(w_1, w_2) \vdash_r 1$ then $alph(w_1) \cup alph(w_2) = \emptyset$ and $w_1 = w_2 = 1$.

3) If $1 \in I$ then $1 \in C$ since $I = \sigma^0(I) \subseteq \bigcup_{i \geq 0} \sigma^i(I) = \sigma^*(I) = C$. Conversely, let $1 \in C = \bigcup_{i \geq 0} \sigma^i(I)$ and, by contradiction, suppose that $1 \notin I = \sigma^0(I)$. Then, there exists $i > 0$ such that $1 \in \sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$. Thus, for the minimal integer $i$ such that the above condition is satisfied, we have $1 \in \sigma'(\sigma^{i-1}(I))$, *i.e.*, there exist $w_1, w_2 \in \sigma^{i-1}(I)$, $r \in R$, such that $(w_1, w_2) \vdash_r 1$. By using 2), we get $w_1 = w_2 = 1$ and this is a contradiction.

4) Let $C = C(SC_{PA})$ with $SC_{PA} = (A, I, R)$ and $1 \in C$. By using 3), we get $1 \in I$. We claim that $C \setminus 1 = C(SC'_{PA})$, where $SC'_{PA} = (A, I \setminus 1, R)$.

Indeed, we first prove, by using induction over $i$, that for each $i \in \boldsymbol{N}$ we have $1 \notin \sigma^i(I \setminus 1) \subseteq \sigma^i(I)$. This relation is obviously satisfied for $i = 0$ since $1 \notin \sigma^0(I \setminus 1) = I \setminus 1 \subseteq I = \sigma^0(I)$ and, for $i > 0$, in view of 2), by using the relation $\sigma^{i+1}(I \setminus 1) = \sigma^i(I \setminus 1) \cup \sigma'(\sigma^i(I \setminus 1))$ and the induction hypothesis. As a consequence, $C(SC'_{PA}) = \bigcup_{i \geq 0} \sigma^i(I \setminus 1) \subseteq C \setminus 1$ since $1 \notin \bigcup_{i \geq 0} \sigma^i(I \setminus 1) \subseteq \bigcup_{i \geq 0} \sigma^i(I) = C$.

Conversely, let us prove that $C \setminus 1 = \bigcup_{i \geq 0} \sigma^i(I) \setminus 1 \subseteq C(SC'_{PA})$, *i.e.*, that for each $i \in \boldsymbol{N}$, we have $\sigma^i(I) \setminus 1 \subseteq C(SC'_{PA})$, by using induction over $i$. If $i = 0$ then $\sigma^0(I) \setminus 1 = I \setminus 1 = \sigma^0(I \setminus 1) \subseteq C(SC'_{PA})$. Otherwise, for $i > 0$, we have $\sigma^i(I) \setminus 1 = [\sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))] \setminus 1$. By induction hypothesis, $\sigma^{i-1}(I) \setminus 1 \subseteq C(SC'_{PA})$. Let $w \in \sigma'(\sigma^{i-1}(I)) \setminus 1$. Then, there exist $w_1, w_2 \in \sigma^{i-1}(I)$, $r \in R$, such that $(w_1, w_2) \vdash_r w$. In view of 1), we have $w_1 \neq 1$ or $w_2 \neq 1$. So, by using once again 1), either $w_1 = 1$, $w_2 = w \in \sigma^{i-1}(I) \setminus 1$ or $w_2 = 1$, $w_1 = w \in \sigma^{i-1}(I) \setminus 1$ or $w_1, w_2 \in \sigma^{i-1}(I) \setminus 1$ and, by using the induction hypothesis, $w \in C(SC'_{PA})$.

5) Let $C = C(SC_{PA})$ with $SC_{PA} = (A, I, R)$ and $1 \notin C$. We claim that $C \cup \{1\} = C(SC'_{PA})$, where $SC'_{PA} = (A, I \cup \{1\}, R)$.

Indeed, it is easy to prove, by using induction over $i$, that for each $i \in \boldsymbol{N}$ we have $\sigma^i(I) \subseteq C(SC'_{PA}) = \bigcup_{i\geq 0} \sigma^i(I \cup \{1\})$. This relation is obvious for $i = 0$ since $\sigma^0(I) = I \subseteq I \cup \{1\} = \sigma^0(I \cup \{1\})$ and, for $i > 0$, by using the relation $\sigma^{i+1}(I) = \sigma^i(I) \cup \sigma'(\sigma^i(I))$ and the induction hypothesis. Consequently, since $1 \in I \cup \{1\} = \sigma^0(I \cup \{1\})$, we have $C \cup \{1\} = \bigcup_{i\geq 0} \sigma^i(I) \cup \{1\} \subseteq C(SC'_{PA})$.

Conversely, let us prove that $C(SC'_{PA}) = \bigcup_{i\geq 0} \sigma^i(I \cup \{1\}) \subseteq C \cup \{1\}$, *i.e.*, that for each $i \in \boldsymbol{N}$ we have $\sigma^i(I \cup \{1\}) \subseteq C \cup \{1\}$. Indeed, it is obvious that $\sigma^0(I \cup \{1\}) = I \cup \{1\} \subseteq C \cup \{1\}$. Furthermore, in view of 1), for $i > 0$, we have $\sigma^i(I \cup \{1\}) = \sigma^{i-1}(I \cup \{1\}) \cup \sigma'(\sigma^{i-1}(I))$. Thus, $\sigma^i(I \cup \{1\}) \subseteq C \cup \{1\}$, by using the induction hypothesis. □

In the next part of this paper, any circular splicing system $SC_X = (A, I, R)$, with $X \in \{PA, PI\}$, is called a *finite* circular splicing system if both $I, R$ are finite sets. Analogously, a Head circular splicing system $SC_H = (A, I, T, P)$ is finite if both $I, T$ are finite sets.

### 2.3. Linear and circular splicing

Attempting to extend the results from the linear case to the case of circular splicing is a natural research direction which arose. Example 2.3 shows that this attempt fails and actually underlines the difference between linear and circular splicing, deepening the observations made in [28]. In our examples we also use Proposition 2.2 which allows us to state that some circular languages are not generated by finite Paun's splicing systems. The analogous result for finite Head's splicing systems can be proved by using a similar argument or can be obtained as a byproduct of Proposition 3.1. Since we only deal with finite circular splicing systems, a circular language $C$ will be called $C_X$ generated, with $X \in \{H, PA, PI\}$, if a *finite* circular splicing system $SC_X$ exists such that $C = C(SC_X)$.

**Proposition 2.2.** *Let $L = Lin(C)$ be an infinite language which is the full linearization of a circular language $C$. Suppose that $L$ satisfies the following condition*

$$\forall l, m \in L : \quad lm \notin L.$$

*Then $C$ is not generated by a finite Paun circular splicing system.*

*Proof.* By contradiction, suppose that a finite Paun circular splicing $SC_{PA} = (A, I, R)$ exists such that $C = C(SC_{PA})$. Let $w$ be a word of $L$ with length greater than the maximal length of the circular words in $I$. Since $\sim\! w \in C \setminus I$, there exist $\sim\! hu_1u_2$, $\sim\! ku_3u_4 \in \sim\! L$ such that $w \sim u_2hu_1u_4ku_3$. Now, $L$ being closed under the conjugacy relation (Prop. 2.1) and $w$ being in $L$, we have $u_2hu_1u_4ku_3 \in L$. Thus, $u_2hu_1 \in \sim\! hu_1u_2$ and $\sim\! hu_1u_2 \in \sim\! L$ and hence $l = u_2hu_1 \in L$. By the same argument, $m = u_4ku_3 \in L$ and $lm = u_2hu_1u_4ku_3 \in L$ which is a contradiction. □

**Example 2.3.** In [15] (resp. [4]) it is shown that $(aa)^*$ is not generated by a finite Head (resp. Pixton) linear splicing system, while, as observed in Example 2.2, its circularization $\{\sim(aa)^n | n \geq 0\}$ is generated by a finite Paun (or Head) circular splicing system. On the contrary, let us consider the regular language $(aa)^*b$. Using results in [7, 9], we know that $(aa)^*b$ is generated by a finite Paun linear system. Take its circularization $C = \sim(aa)^*b$. Notice that there is a linearization $\{a^k b a^k \mid k \in \boldsymbol{N}\}$ of $C$ which is not regular. However, the full linearization $Lin(C) = (aa)^*\{b, aba\}(aa)^*$, given in [26], is regular. Thus, $C \in Reg^\sim$ (Sect. 2.1). Furthermore, for all $l, m \in Lin(C)$, we have $|lm|_b = 2$ and so $lm \notin Lin(C)$. Using Proposition 2.2, we have that $C$ cannot be generated by a finite Paun (or Head) circular splicing system.

### 2.4. Descriptional complexity

In Section 4, we introduce the main problem we deal with in this paper but also other interesting questions about circular splicing could be asked and will be outlined in the next part of this section. One of these interesting problems could be the investigation of the *splicing sub-system* of a given circular splicing system. For example, $\sim(A^2)^*$, for $A = \{a, b\}$, is $C_{PA}$ generated by $I = \sim A^2$ and $R = \{w_1|1\$1|\, w_2 \mid \; w_1, w_2 \in A^2\}$ (Prop. 6.3). Thus, with the same $A$ and $I$, by considering $R_1 = \{aa|1\$1|aa\} \subseteq R$, we obtain $\sim(aa)^*$, with $R_2 = \{bb|1\$1|bb\} \subseteq R$ we generate $\sim(bb)^*$ and with $R_3 = \{ab|1\$1|ab\} \subseteq R$ we obtain $\sim(ab)^*$. It would be interesting to prove or to disprove that all circular languages can be obtained as described before.

Let us take $L_1 = \{w \in A^* \mid \exists h, k \in \boldsymbol{N}\ |w|_a = 2k, |w|_b = 2h\}$. It is easy to see that $L_1$ is a language closed under conjugacy. Moreover, $L_1$ is a regular language since $L_1$ is the shuffle of $(aa)^*$ and $(bb)^*$ (see [21] for the definition of the shuffle operation between languages). Thus, $\sim L_1$ is a regular circular language. In Section 6, we will see that the splicing system $SC_{PA} = (A, I, R)$ generates $\sim L_1$, when we choose $A = \{a, b\}$, $I = \{\sim aa, \sim bb, \sim abab\} \cup 1$ and $R = \{aa|1\$1|1, bb|1\$1|1, ab|1\$1|1\}$. On the other hand, one can see that $\sim L_1$ cannot be generated by choosing a proper subset of $R$. This observation is related to the notion of a *minimal splicing system* which is introduced in [25] and which is a typical research topic concerning *descriptional complexity*. It goes without saying that this notion is the counterpart of the minimal automaton for regular languages. Here, the minimality of the system could be referred to the cardinality of $R$ or to the length of the rules in $R$. In the first case, a complete answer to this question will be given for languages over a one-letter alphabet in Section 7.

### 2.5. Other models of circular splicing systems

As we have already said in Section 1, additional hypotheses can be added to the definitions of circular splicing given by Paun and Pixton. We report them below along with a result proved in [26] which uses these hypotheses.

**Hypothesis 1.** *$R$ is a symmetric scheme, i.e. for each rule $r = u_1|u_2\$u_3|u_4$ (resp. $r = (\alpha, \alpha'; \beta)$) in the splicing system $SC_{PA}$ (resp. $SC_{PI}$) there is the rule $\overline{r} = u_3|u_4\$u_1|u_2$ (resp. $\overline{r} = (\alpha', \alpha; \beta')$).*

**Remark 2.4.** Any set $R$ of rules in a splicing system $SC_{PI}$ is implicitly supposed to be symmetric.

**Hypothesis 2.** $R$ is a reflexive scheme, *i.e.* for each rule $u_1|u_2\$u_3|u_4$ (resp. $(\alpha, \alpha'; \beta)$) in the splicing system $SC_{PA}$ (resp. $SC_{PI}$) there are the rules $u_1|u_2\$u_1|u_2$ and $u_3|u_4\$u_3|u_4$ (resp. $(\alpha, \alpha; \alpha)$ and $(\alpha', \alpha'; \alpha')$).

**Hypothesis 3.** Self-splicing. Self-splicing is defined in a splicing system $SC_{PA}$ (resp. $SC_{PI}$) producing $^\sim u_4hu_1$ and $^\sim u_2ku_3$ from $^\sim hu_1u_2ku_3u_4$ and the rule $u_1|u_2\$u_3|u_4$ (resp. $^\sim\beta\epsilon', ^\sim\beta'\epsilon$ starting from $^\sim\alpha\epsilon\alpha'\epsilon'$ and the rules $(\alpha, \alpha'; \beta)$, $(\alpha', \alpha; \beta')$). We denote $w \vdash_r (w', w'')$ (resp. $w \vdash_{r,\bar{r}} (w', w'')$) the fact that $(w', w'')$ is produced from $w$ by using self-splicing with a rule $r$ (resp. a pair of rules $r$ and $\bar{r}$).

Below we report a known result which uses Pixton's systems with Hypotheses 1–3 and which generalizes a similar theorem proved for linear splicing [26]. In order to be more precise, we first give the definition of circular languages generated by splicing systems which use both splicing and self-splicing operations.

Given a splicing system $SC_X$, with $X \in \{PA, PI\}$, and given a language $C \subseteq {}^\sim A^*$, we denote $\tau'(C) = \{z \in {}^\sim A^* \mid \exists w', w'' \in C, \exists r \in R.\ (w', w'') \vdash_r z\} \cup \{z', z'' \in {}^\sim A^* \mid \exists w \in C, \exists r \in R.\ w \vdash_r (z', z'')\}$ (resp. $\tau'(C) = \{z \in {}^\sim A^* \mid \exists w', w'' \in C, \exists r \in R.\ (w', w'') \vdash_{r,\bar{r}} z\} \cup \{z', z'' \in {}^\sim A^* \mid \exists w \in C, \exists r \in R.\ w \vdash_{r,\bar{r}} (z', z'')\}$). Thus, we define $\tau^0(C) = C$, $\tau^{i+1}(C) = \tau^i(C) \cup \tau'(\tau^i(C))$, $i \geq 0$, and then $\tau^*(C) = \bigcup_{i \geq 0} \tau^i(C)$.

**Theorem 2.1** [26]**.** *Let $SC_{PI} = (A, I, R)$ be a circular splicing system with $I$ a regular circular language and $R$ reflexive and symmetric. Then $\tau^*(I)$ is regular.*

## 3. Computational power of circular splicing systems

In Section 4 we will introduce the problem we will focus on in the next part of this paper. So, before going on, it is necessary to specify which of the three abovementioned definitions will be referred to in our investigation. This observation naturally leads to the question posed in Problem 3.1 which has a negative answer, as Example 3.1 shows. Proposition 3.1 is an attempt to give a more precise answer showing that the computational power of circular splicing systems increases when we substitute Head's systems with Paun's systems. We also give examples that lead us to believe that an analogous result holds for Paun's systems and Pixton's systems (Exs. 3.1 and 3.2). Nevertheless, in this paper we mainly deal with finite Paun's systems and with the corresponding class of generated languages, denoted $C(Fin, Fin)$.

Below we formalize the problem of comparing the computational power of the three definitions of circular splicing given in Section 2.2.

**Problem 3.1.** *Given $C = C(SC_X)$, with $X \in \{H, PA, PI\}$, does $SC_Y$ exist, with $Y \neq X$, $Y \in \{H, PA, PI\}$, such that $C = C(SC_Y)$?*

Proposition 3.1 gives a partial answer to Problem 3.1. This proposition shows that Paun's systems have a greater computational power than Head's systems.

**Proposition 3.1.** *If $C \subseteq {}^{\sim}A^*$ is $C_H$ generated, then $C$ is $C_{PA}$ generated.*

*Proof.* Given $SC_H = (A, I, T, P)$ and $C = C(SC_H)$, let us consider the system $SC_{PA} = (A, I, R)$, where $R = \{r \mid r = px|q\$ux|v,\ (p,x,q),(u,x,v) \in T,\ (p,x,q)P(u,x,v)\}$, and $C' = C(SC_{PA})$. Set $C = \bigcup_{i\geq 0} \sigma_H^i(I)$ and $C' = \bigcup_{i\geq 0} \sigma_P^i(I)$, where we denote by $\sigma_H$ (resp. $\sigma_P$) the function $\sigma$ defined in Section 2.2 when we refer to $SC_H = (A, I, T, P)$ (resp. $SC_{PA} = (A, I, R)$). We prove that $C = C'$, *i.e.*, that for each $i \in \boldsymbol{N}$ we have $\sigma_H^i(I) = \sigma_P^i(I)$, by using induction over $i$.

Indeed, for $i = 0$, we obviously have $\sigma_H^0(I) = I = \sigma_P^0(I)$. On the other hand, for $i > 0$, we have $w \in \sigma_H^i(I) = \sigma_H^{i-1}(I) \cup \sigma'_H(\sigma_H^{i-1}(I))$ if and only if either $w \in \sigma_H^{i-1}(I)$ or there exists $w' = {}^{\sim}hpxq, w'' = {}^{\sim}kuxv \in \sigma_H^{i-1}(I)$, with $(p,x,q),(u,x,v) \in T$, $(p,x,q)P(u,x,v)$, $w = {}^{\sim}hpxvkuxq$. In the first case, by using the induction hypothesis, we have $w \in \sigma_P^{i-1}(I) \subseteq \sigma_P^i(I)$. The second case is equivalent to the existence of $w', w'' \in \sigma_P^{i-1}(I) = \sigma_H^{i-1}(I)$ (induction hypothesis) and to a rule $r = px|q\$ux|v$ in $R$ (by construction of $R$) such that $w', w''$ generate $w$ through $r$, *i.e.*, $w \in \sigma_P^i(I)$. Conversely, if $w \in \sigma_P^i(I)$, then either $w \in \sigma_P^{i-1}(I) = \sigma_H^{i-1}(I) \subseteq \sigma_H^i(I)$ (induction hypothesis) or $w \in \sigma'_P(\sigma_P^{i-1}(I))$. As we have already observed, in the latter case, we have $w \in \sigma'_H(\sigma_H^{i-1}(I)) \subseteq \sigma_H^i(I)$. □

**Remark 3.1.** One can ask whether the result contained in Proposition 3.1 can be extended to finite Pixton's systems, *i.e.*, if each $C \subseteq {}^{\sim}A^*$ which is $C_{PA}$ generated is also $C_{PI}$ generated. We do not know whether this question has a positive answer. However, a trivial extension of the argument contained in the proof of Proposition 3.1 does not work.

Indeed, given $SC_{PA} = (A, I, R)$, we consider the system $SC_{PI} = (A, I, R')$, where $R' = \{(u_1u_2, u_3u_4; u_1u_4), (u_3u_4, u_1u_2; u_3u_2) \mid u_1|u_2\$u_3|u_4 \in R\}$. Examples exist such that $C(SC_{PA}) \neq C(SC_{PI})$.

For instance, let us consider the finite Paun circular splicing system $SC_{PA} = (A, I, R)$, where $A = \{a, b\}$, $I = \{\ {}^{\sim}aa,\ {}^{\sim}bb\ \}$ and $R = \{aa|1\$1|bb, a|a\$b|b\}$. We prove that, for each $w \in C(SC_{PA}) = \bigcup_{i\geq 0} \sigma^i(I)$, there exist $m, n \in \boldsymbol{N}$ such that $|w|_a = 2n$, $|w|_b = 2m$, by using induction on the minimal $i$ such that $w \in \sigma^i(I)$. Indeed, if $i = 0$, *i.e.*, $w \in I$, then the conclusion holds. Thus, let us suppose $i > 0$. Since $\sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$, by using induction hypothesis, we can suppose that there exist $w_1, w_2 \in \sigma^{i-1}(I)$, $r \in R$ such that $(w_1, w_2) \vdash_r w$. Thus, $w_1 = {}^{\sim}haa$, $w_2 = {}^{\sim}kbb$ and, by induction hypothesis, $n_1, n_2, m_1, m_2 \in \boldsymbol{N}$ exist such that $|w_1|_a = 2n_1$, $|w_1|_b = 2m_1$, $|w_2|_a = 2n_2$, $|w_2|_b = 2m_2$. As a result, we have $|w|_a = 2n$, $|w|_b = 2m$, with $n = n_1 + n_2$, $m = m_1 + m_2$. As a consequence ${}^{\sim}aaabbb \notin C(SC_{PA})$.

Now, let us consider the canonical transformation of the splicing system $SC_{PA}$ given above into the Pixton system $SC_{PI} = (A, I, R')$, where $R' = \{t_1, \bar{t_1}, t_2, \bar{t_2}\}$ with $t_1 = (aa, bb; aabb)$, $\bar{t_1} = (bb, aa; 1)$, $t_2 = (aa, bb; ab)$, $\bar{t_2} = (bb, aa, ba)$.

We can easily check that $(\sim aa, \sim bb) \vdash_{t_1,\bar{t_2}} \sim aabbba$. As a consequence $C(SC_{PA}) \neq C(SC_{PI})$.

On the other hand, let $SC_{PA} = (A, I, R)$ be a Paun system satisfying the following property: for each $u_1|u_2\$u_3|u_4, u'_1|u'_2\$u'_3|u'_4 \in R$, if $u_1u_2 = u'_1u'_2$ and $u_3u_4 = u'_3u'_4$ then, we have $u_1u_4 = u'_1u'_4$, $u_3u_2 = u'_3u'_2$. Thus, it is easy to see that, for the Pixton system $SC_{PI} = (A, I, R')$, with $R' = \{(u_1u_2, u_3u_4; u_1u_4), (u_3u_4, u_1u_2; u_3u_2) \mid u_1|u_2\$u_3|u_4 \in R\}$, we have $C(SC_{PA}) = C(SC_{PI})$.

The following examples give results concerning the computational power of circular splicing systems. Namely, we have already observed that Paun's circular splicing systems cannot generate all regular circular languages (Ex. 2.3). Pixton's systems seem to have a greater computational power and this is emphasized in Examples 3.1 and 3.2. Nevertheless, regular circular languages exist that cannot be generated by any finite circular splicing system [10].

**Example 3.1.** Let us consider again $C = \sim(aa)^*b$. We know that $C$ cannot be $C_H$ or $C_{PA}$ generated (Ex. 2.3). On the contrary, we prove that $C$ is $C_{PI}$ generated by choosing $SC_{PI} = (A, I, R)$, with $A = \{a, b\}, I = \{\sim b, \sim a^2b, \sim a^4b\}, R = \{(a^2, a^2b; a^2), (a^2b, a^2; 1)\}$. Firstly let us show that $C(SC_{PI}) = \bigcup_{i \geq 0} \sigma^i(I) \subseteq C$, by induction on the minimal $i$ such that $w \in \sigma^i(I)$. If $i = 0$, *i.e.*, $w \in \sigma^0(I) = I$, then $w \in C$. Now, let $w \in \sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$, with $i > 0$ and let us prove that $w \in C$. By using induction hypothesis, we can suppose that there exist $w_1, w_2 \in \sigma^{i-1}(I)$ and $r, \bar{r} \in R$ such that $(w_1, w_2) \vdash_{r,\bar{r}} w$. Since $w_1, w_2 \in \sigma^{i-1}(I)$, then, by induction hypothesis, $w_1, w_2 \in C$. Thus, $k', h' \in \boldsymbol{N}$ exist such that $|w_1|_a = 2k', |w_2|_a = 2h'$ and $|w_1|_b = 1 = |w_2|_b$, *i.e.* $w_1 = \sim(aa)^{k'}b$, $w_2 = \sim(aa)^{h'}b$, $k', h' \geq 1$. Given that $R = \{(a^2, a^2b; a^2), (a^2b, a^2; 1)\}$, looking at Pixton's definition of splicing, with the same notations as in this definition, we see that $w_1 = \sim aa\epsilon$ with $\epsilon = (aa)^{k'-1}b$, $w_2 = \sim aab\epsilon'$ with $\epsilon' = (aa)^{h'-1}$ and $w = \sim \epsilon aa\epsilon'$. Thus, $w = \sim(aa)^{k'}b(aa)^{h'-1} \in C$.

*Vice versa*, let us show that $C \subseteq C(SC_{PI})$, *i.e.*, by using induction over $n$, we prove that for every $n \geq 0$, we have $w = \sim(aa)^nb \in C(SC_{PI})$. If $n = 0, 1, 2$, then $w \in I$. Let us suppose $n > 2$ and $\sim(aa)^ib \in C(SC_{PI})$, for $0 \leq i < n$. We prove that there exist $w_1, w_2 \in C$ and $r, \bar{r} \in R$ such that $(w_1, w_2) \vdash_{r,\bar{r}} \sim(aa)^nb$. Take $w_1 = \sim(aa)^{n-1}b$ and $w_2 = \sim(aa)^2b$. By induction hypothesis, $w_1, w_2 \in C(SC_{PI})$. Furthermore, by setting $(aa)^{n-2}b = \epsilon$, $aa = \epsilon'$ and by using the rules $r, \bar{r} \in R$, we get $(w_1, w_2) \vdash_{r,\bar{r}} \sim(aa)^{n-2}baaaa = \sim(aa)^nb$.

In the example below we use the notion of factors of a circular word: $x \in A^*$ is a *factor* of a circular word $w$ when a representative $w'$ of $w$ exists such that $x$ is a factor of $w'$.

**Example 3.2.** Consider $L_2 = \{w \in A^* \mid \exists h, k \in \boldsymbol{N}\ |w|_a = 2k+1, |w|_b = 2h+1\}$. It is easy to see that $L_2$ is a language closed under conjugacy. Moreover, $L_2$ is a regular language since $L_2$ is the shuffle of $(aa)^*a$ and $(bb)^*b$ (see [21] for the definition of the shuffle operation between languages). Thus, $\sim L_2$ is a regular circular language. On the other hand, for all $l, m \in L_2$, since $|lm|_a$ and $|lm|_b$ are both even, then $lm \notin L_2$. So, thanks to Proposition 2.2, we have that $^\sim L_2$ is not $C_H$ or $C_{PA}$ generated.

Let us prove that $^\sim L_2$ is $C_{PI}$ generated by choosing $SC_{PI} = (A, I, R)$, with $A = \{a, b\}$, $I = \{^\sim ab, ^\sim aaab, ^\sim bbba\}$, $R = \{r_0, r_1, r_2, r_3\}$, $r_1 = (ab, 1; aa), r_2 = (ab, 1; bb), r_3 = (ab, 1; abab)$, $\overline{r}_1 = \overline{r}_2 = \overline{r}_3 = r_0 = (1, ab; 1)$ (see Rem. 2.1).

Firstly, let us show that $C(SC_{PI}) = \bigcup_{i\geq 0} \sigma^i(I) \subseteq^\sim L_2$, by induction on the minimal $i$ such that $w \in \sigma^i(I)$. Clearly, $\sigma^0(I) = I \subseteq^\sim L_2$. Let $w \in \sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$ with $i > 0$. By using induction hypothesis, we can suppose that $w_1, w_2 \in \sigma^{i-1}(I)$ exist and $r, \overline{r} \in R$ also exist such that $(w_1, w_2)\vdash_{r,\overline{r}} w$. By induction hypothesis, $w_1, w_2 \in^\sim L_2$. We can distinguish three cases.

*Case* 1. Let $\overline{r} = r_0 = (1, ab; 1)$ and $r = r_1 = (ab, 1; aa)$. Thus $w_1 =^\sim \epsilon$, $w_2 =^\sim ab\epsilon'$ and so $w =^\sim \epsilon\epsilon' aa$. We have $w \in^\sim L_2$. Indeed, $|w|_a = |\epsilon\epsilon' aa|_a = |w_1|_a + |\epsilon'|_a + 2 = |w_1|_a + |w_2|_a + 1$. By induction hypothesis $|w_1|_a$ and $|w_2|_a$ are both odd, so $|w|_a$ is odd. Moreover, $|w|_b = |\epsilon\epsilon' aa|_b = |\epsilon\epsilon'|_b = |w_1|_b + |\epsilon'|_b = |w_1|_b + |w_2|_b - 1$. By induction hypothesis $|w_1|_b$ and $|w_2|_b$ are both odd, so $|w|_b$ is odd.

*Case* 2. Let $\overline{r} = r_0 = (1, ab; 1)$ and $r = r_2 = (ab, 1; bb)$. This case is symmetric to case 1, so we can use the same argument as in case 1 by substituting $a$ with $b$ and *vice versa*.

*Case* 3. Let $\overline{r} = r_0 = (1, ab; 1)$ and $r = r_3 = (ab, 1; abab)$. Thus, $w_1 =^\sim \epsilon$, $w_2 =^\sim ab\epsilon'$ and so $w =^\sim \epsilon\epsilon' abab$. We have $w \in^\sim L_2$. Indeed, $|w|_a = |\epsilon\epsilon' abab|_a = |\epsilon\epsilon'|_a + 2 = |\epsilon|_a + |\epsilon'|_a + 2 = |w_1|_a + |w_2|_a + 1$. By induction hypothesis $|w_1|_a$ and $|w_2|_a$ are both odd, so $|w|_a$ is odd. Moreover, $|w|_b = |\epsilon\epsilon' abab|_b = |\epsilon\epsilon'|_b + 2 = |w_1|_b + |w_2|_b + 1$. Once again, by induction hypothesis $|w_1|_b$ and $|w_2|_b$ are both odd, so $|w|_b$ is odd.

*Vice versa*, let us show that $^\sim L_2 \subseteq C(SC_{PI})$ by using induction over $|w|$, $w \in^\sim L_2$. If $w \in^\sim L_2$ and $|w| < 5$, then $w \in I$. Let $w \in^\sim L_2$ with $|w| = n \geq 5$. Once again, we can distinguish three cases.

*Case* 1. Suppose that $aa$ is a factor of $w$, *i.e.*, $w_1 aa w_2 \in w$. Take $x = w_2 w_1$ and $y = ab$. We have $|x| < |w|$, $^\sim y \in I \subseteq C(SC_{PI})$ and $^\sim x \in^\sim L_2$ ($|x|_a = |w|_a - 2$ and $|x|_b = |w|_b$ are both odd numbers). Thus, by induction hypothesis, $^\sim x \in C(SC_{PI})$. Furthermore, $(^\sim x, ^\sim y)\vdash_{r_0,r_1} w$. Indeed, when we set $\epsilon = w_2 w_1$, $\beta = 1$, $\epsilon' = 1$, $\beta' = aa$, we get $(x, y)\vdash_{r_0,r_1} {}^\sim\epsilon\beta\epsilon'\beta' = {}^\sim w_2 w_1 aa = w$.

*Case* 2. Suppose that $bb$ is a factor of $w$, *i.e.*, $w_1 bb w_2 \in w$. This case is symmetric to case 1, so by using the same argument as in case 1, we can prove that $w =^\sim w_2 w_1 bb \in C(SC_{PI})$ by using $r_0, r_2$ (applied to $^\sim x =^\sim w_2 w_1$ and $^\sim y =^\sim ab$).

*Case* 3. Suppose that neither $aa$ nor $bb$ are factors of $w$. Thus, $w =^\sim (ab)^t$, where $t$ is an odd number and $t > 1$. Consider $^\sim x =^\sim (ab)^{t-2}$ and $^\sim ab \in^\sim L_2$. Since $t$ is odd, $t - 2$ is odd too, *i.e.* $^\sim x =^\sim (ab)^{t-2} \in^\sim L_2$. In addition, we have $^\sim ab \in I \subseteq C(SC_{PI})$, $|x| < |w|$ and so, by induction hypothesis, $^\sim x \in C(SC_{PI})$.

Finally, when we set $\alpha = 1, \epsilon = x, \beta = 1$ and $\alpha' = ab$, $\epsilon' = 1, \beta' = abab$, we have $({}^\sim x, {}^\sim ab) \vdash_{r_0, r_3} {}^\sim \epsilon\beta\epsilon'\beta' = {}^\sim xabab = {}^\sim (ab)^t$.

## 4. Main problem

It is already known that in contrast with the linear case, $C(Fin, Fin)$ is not intermediate between two classes of languages in the Chomsky hierarchy. For example, in [28] the authors show that ${}^\sim(ca)^n(cb)^n$ is $C_H$ generated. A slight modification of their arguments shows that ${}^\sim a^n b^n$ is $C_H$ generated with a finite initial circular language, but ${}^\sim a^n b^n$ is a context-free circular language which is not a regular circular language (since it has no regular linearization). On the other hand, we have already seen examples of regular circular languages that cannot be generated by using finite Paun circular splicing systems (Exs. 2.3 and 3.2). In addition, regular circular languages exist which cannot be generated by any finite circular splicing system [10]. So far, we have not yet discovered whether $C(Fin, Fin)$ contains any context-sensitive or recursively enumerable circular language which is not context-free. Nevertheless, in this paper we restrict our investigation to the following problem.

**Problem 4.1.** *Characterize* $Reg^\sim \cap C(Fin, Fin)$.

In the next sections, we approach Problem 4.1 by dealing with different classes of languages. In Problem 4.2, which generalizes Problem 4.1, we also take into account the additional hypotheses given in Section 2.5.

**Problem 4.2.** *Can we characterize* $FA^\sim \cap C(Fin, Fin)$ *(resp.* $FA^\sim \cap C(Reg, Fin)$*) for each class* $FA$ *of languages in the Chomsky hierarchy, for every definition of* $C(Fin, Fin)$ *(resp.* $C(Reg, Fin)$*) and for every possible combination of the three hypotheses given in Section 2.5?*

We conclude this section with comments and questions related to Problem 4.1, *i.e.* the characterization of the regular circular languages generated by finite circular splicing systems. In all of them, we will refer to Paun's definition of circular splicing but the same conclusions maintain for Pixton's systems [10].

First of all, note that the class of regular languages closed under conjugacy relation is closed under union. However, while ${}^\sim(A^2)^*$, ${}^\sim(A^3)^* \in C(Fin, Fin)$, as shown in Proposition 6.3, we prove in Proposition 4.1 that ${}^\sim(A^2)^* \cup {}^\sim(A^3)^* \notin C(Fin, Fin)$. This means that $Reg^\sim \cap C(Fin, Fin)$ is not closed with respect to union. One could ask for additional hypotheses to be added so that the union of regular circular splicing languages will still be a circular splicing language. Proposition 6.3 gives a partial answer to this question. This research line could bring us to a complete characterization of the class of regular circular languages that are generated by finite circular splicing. In the next proposition, as usual, for positive integers $s, t$ we say that $s$ divides $t$ if $t = sv$ for a positive integer $v$.

**Proposition 4.1.** $\sim(A^2)^* \cup \sim(A^3)^* \notin C(Fin, Fin)$.

*Proof.* By contradiction, suppose that $\sim(A^2)^* \cup \sim(A^3)^* = C(SC_{PA})$, for a finite circular splicing system $SC_{PA} = (A, I, R)$. Circular splicing systems with an initial finite set and an empty set of rules generates only finite sets. So, $R \neq \emptyset$ and let $r = u_1|u_2\$u_3|u_4 \in R$. Now, there exists $h \in A^*$ such that 3 does not divide $s = |hu_1u_2|$ (take $h = 1$ if $|u_1u_2| \neq 0 \pmod 3$, $h \in A$ otherwise). Analogously, there exists $k \in A^*$ such that 2 does not divide $t = |ku_3u_4|$ (take $k = 1$ if $|u_3u_4| \neq 0 \pmod 2$, $k \in A$ otherwise). Thus, consider the two words $w = (hu_1u_2)^2$ and $z = (ku_3u_4)^3$. We have $\sim w \in \sim(A^2)^*$, $\sim z \in \sim(A^3)^*$ and so $\sim w, \sim z \in C(SC_{PA})$. Furthermore, rule $r$ can be applied to $\sim w$, $\sim z$. By definition, this rule $r$ generates a circular word $\sim y$ which should be in $C(SC_{PA})$ and $|y| = 2s+3t$. Now, $\sim y \notin \sim(A^2)^*$ (since 2 does not divide $2s+3t$) and $\sim y \notin \sim(A^3)^*$ (since 3 does not divide $2s+3t$), and this is a contradiction. □

## 5. Cycles

In this section we introduce some definitions which will be used in the statements of the main results in this paper. Let $L$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be the minimal finite state automaton recognizing $L$. In the next part of this paper, we say that a word $c \in A^+$ is a *label of a closed path* if there exists $q \in Q$ such that $\delta(q, c) = q$. Intuitively, if a regular infinite language $L$ is generated by a finite circular splicing system, we must exhibit a finite set of rules which are able to produce words with a non-bounded number of occurrences of such labels as factors. This is the reason for which we give a definition for referring to some special labels $c$ (*cycles*) of closed paths in $\mathcal{A}$. As a matter of fact we mainly deal with a restricted subset of cycles, namely *simple cycles*, but the more general Definition 5.1 is given below for the sake of completeness. As another motivation, we notice that the same notion of a cycle has been considered in connection with the counterpart of Problem 4.1 for linear splicing systems in [9] where Proposition 5.1 has also been proved. Finally, in [5], the definition of cycles has been outlined along with the notion of *fingerprint of a cycle*. These two notions allowed us to prove the main result in [5]. This main result is generalized in Theorem 6.1 and its proof does not make use of these notions.

**Definition 5.1** *Cycle*. Let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be a trim deterministic finite state automaton. Let $c, q$ be such that $c \in A^+$, $q \in Q$ and $\delta(q, c) = q$. Let $a_1, \ldots, a_n \in A$ be such that $c = a_1 \cdots a_n$, $n \geq 1$. The word $c \in A^+$ is a *cycle* in $\mathcal{A}$ (with respect to the transition $\delta(q, c) = q$) if the internal states crossed by the transition $\delta(q, c) = q$ are different from $q$ (*i.e.*, for all $c', c'' \in A^+$ such that $c = c'c''$, we have $\delta(q, c') \neq q$) and either the states $\delta(q, a_1 \cdots a_i)$, $i \in \{1, \ldots, n\}$, are different from one another (*simple cycle*) or the condition which follows is satisfied.

- There exist a positive integer $k$, $u_1, \ldots, u_{k+1}, c_1, \ldots, c_k \in A^+$, positive integers $p_1, \ldots, p_k$ and $q'_0, q'_1, \ldots, q'_k$, $q'_{k+1} \in Q$, with $q'_0 = q'_{k+1} = q$, such that

- $c = u_1 c_1^{p_1} u_2 \cdots u_k c_k^{p_k} u_{k+1}$;
- $\delta(q'_{i-1}, u_i) = q'_i$, $1 \leq i \leq k+1$ and the internal states crossed by the transition $\delta(q'_0, u_1 \cdots u_{k+1})$ are different from one another (and with respect to $q$);
- for each $i \in \{1, \ldots k\}$, $\delta(q'_i, c_i) = q'_i$ and either $c_i$ is a cycle (with respect to $\delta(q'_i, c_i) = q'_i$) or $p_i = 1$ and $c_i$ is the concatenation of powers of distinct cycles, *i.e.*, $c_i = (c'_1)^{r_1}(c'_2)^{r_2} \cdots (c'_s)^{r_s}$, with $c'_j$ being a cycle (with respect to $\delta(q'_i, c'_j) = q'_i$) and $c'_j \neq c'_{j'}$, $j, j' \in \{1, \ldots, s\}$.

A cycle $c$ with respect to $\delta(q, c) = q$ is simply named cycle when the context makes the meaning evident.

The characterization of the labels of closed paths which are not cycles given in the proposition below has been proved in [9]. In this proposition, concatenation means the concatenation of words in $A^*$.

**Proposition 5.1** [9]**.** *Let $L$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be the minimal finite state automaton recognizing $L$. Let $c, q$ be such that $c \in A^+$, $q \in Q$ and $\delta(q, c) = q$. If $c$ is not a cycle, then one of the following cases occurs.*

(1) *There exist $c_1, \ldots, c_k \in A^+$ and positive integers $p_1, \ldots, p_k$ such that $c = c_1^{p_1} c_2^{p_2} \cdots c_k^{p_k}$, $c_i$ is a cycle (with respect to $\delta(q, c_i) = q$), $1 \leq i \leq k$, and either $k > 1$ or $k = 1$ and $p_k > 1$.*
(2) *There exist $h, c_1, g, \lambda \in A^+$, $q' \in Q$, such that $c = hc_1\lambda c_1 g$, $\delta(q, h) = q'$, $\delta(q', c_1) = q' = \delta(q', \lambda)$, $\delta(q', g) = q$, $c_1$ is a cycle and $\lambda$ is not a power of $c_1$ (i.e., $\lambda \neq c_1^t$, $t \geq 1$).*

## 6. Regular languages in $C(Fin, Fin)$: cycle closed star languages

In this section, we will introduce a class of regular languages whose circularization we prove to be in $C(Fin, Fin)$. As we have already said, we will consider languages $L$ closed under conjugacy relation, and the action of circular splicing over their circularization $^\sim L = C$. Furthermore, since we will be referring to Problem 4.1, $L$ will be regular. Here, we also consider the particular case in which $L = X^*$ is the Kleene closure of a regular language (Def. 6.1). In this case, we prove that the circularization of $L = X^*$ is in $C(Fin, Fin)$ under the additional hypothesis that either $X$ is finite or that $X^*$ is a *cycle closed language*, *i.e.*, $X^*$ contains each simple cycle in the minimal finite state automaton $\mathcal{A} = (Q, A, \delta, q_0, F)$ recognizing $L$ (Th. 6.1, Prop. 6.3). Several examples are given comparing the class of languages satisfying these additional hypotheses with both circularizations of star languages and regular circular languages generated by (Paun or Pixton) splicing systems (Exs. 6.2–6.4) and some questions are posed (Probs. 6.1 and 6.2).

**Definition 6.1.** A *star language* is a language $L \subseteq A^*$ that satisfies the following conditions:

(1) $L = X^*$, with $X$ a regular language;
(2) $L$ is closed under conjugacy relation.

Under an additional hypothesis, a star language has its circularization in $C(Fin, Fin)$, as shown in Theorem 6.1, the proof of which can be obtained thanks to the following proposition.

**Proposition 6.1.** *Let $X^*$ be a star language and let $SC_{PA} = (A, I, R)$ be a splicing system. If $I \subseteq^{\sim} X^*$ then $C(SC_{PA}) \subseteq^{\sim} X^*$.*

*Proof.* Let us prove that $C(SC_{PA}) = \bigcup_{i \geq 0} \sigma^i(I) \subseteq^{\sim} X^*$ holds, by using induction on the minimal $i$ such that $^{\sim}y \in \sigma^i(I)$. If $i = 0$, *i.e.*, $^{\sim}y \in \sigma^0(I) = I$ then $^{\sim}y \in^{\sim} X^*$, since $I \subseteq^{\sim} X^*$. Otherwise, $^{\sim}y \in \sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$ with $i > 0$. By using induction hypothesis, we can suppose that $y' \in^{\sim} y$ exists such that $y' = u_2hu_1u_4ku_3$ with $^{\sim}u_2hu_1, ^{\sim}u_4ku_3 \in \sigma^{i-1}(I)$. By induction hypothesis, $^{\sim}u_2hu_1, ^{\sim}u_4ku_3 \in^{\sim} X^*$ and, $X^*$ being closed under conjugacy, $u_2hu_1, u_4ku_3 \in X^*$. Because $X^*$ is a submonoid, $u_2hu_1u_4ku_3 = y' \in X^*$, and so $^{\sim}y \in^{\sim} X^*$. □

**Definition 6.2.** Let $L$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be the minimal finite state automaton recognizing $L$. $L$ is *cycle closed* if for each simple cycle $c$ in $\mathcal{A}$, we have $c \in L$.

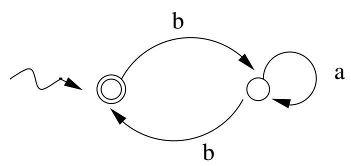


FIGURE 1. Automaton for $(ba^*b)^*$.

The notion of cycle closed language $L$ is referred to the minimal finite state automaton recognizing $L$. Nevertheless, it is easy to see that all the arguments which follow maintain when $L$ is a language containing all the simple cycles in a trim deterministic finite state automaton recognizing $L$. If we give this more general definition, decidability questions can obviously be asked. Namely, can we decide whether a star language is cycle closed? One of the difficulties in answering this question is that in order to prove a star language $L$ does not have this property, we must check all finite automata recognizing $L$. So, a further development of this paper will be to prove (or disprove) that a star language which is cycle closed with respect to a finite automaton recognizing it, is always cycle closed with respect to a standard automaton, *e.g.*, the minimal automaton recognizing it.

**Example 6.1.** Let us consider the regular language $L = (ba^*b)^*$. $L = (ba^*b)^*$ is not a cycle closed language. Indeed, for the cycle $c_1 = a$ in the minimal automaton which recognizes it, depicted in Figure 1, we have $c_1 = a \notin L$. Furthermore, let $\mathcal{A}$ be a trim deterministic automaton recognizing $L$. Let $k$ be a positive integer greater than the number of the states in $\mathcal{A}$ and consider $ba^kb \in L$. It is easy to see that there exists $j$ such that $j \leq k$ and $a^j$ is a cycle in $\mathcal{A}$. Thus, since $L \cap a^* = \emptyset$, $L$ is not a cycle closed language with respect to $\mathcal{A}$. Observe that $L$ is not a star language.

We now prove that the circularization of a *cycle closed star language* (*i.e.*, a language which satisfies both Defs. 6.1 and 6.2) is generated by a finite circular splicing system. In order to do that, we need two preliminary results. In Lemma 6.1 we will define a set of words $I_1$ and we will show that $I_1$ is a finite set. The circularization of $I_1$ will be the initial set of the splicing system associated with a cycle closed star language $L$ (Th. 6.1). Informally, $I_1$ will be the set of the labels $w$ of the successful paths in $\mathcal{A}$ such that, for each state $q$ crossed by the transitions associated with $w$, there exists at most one closed path which starts and ends in $q$ and this closed path has a simple cycle as a label. In Lemma 6.2, we prove that each label of a closed path has a simple cycle as a factor.

**Lemma 6.1.** *Let $L$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be the minimal finite state automaton recognizing $L$. Let*

$$I_1 = \{y \in L \mid \forall u, v \in A^*, c \in A^+, q \in Q \text{ if } y = ucv$$

$$\text{with } \delta(q_0, u) = q = \delta(q, c), \text{ then } c \text{ is a simple cycle in } \mathcal{A}\}\cdot$$

*Thus, $I_1$ is a finite set. Furthermore, suppose that $L$ is a cycle closed language and let $\mathcal{C}(L)$ be the set of the simple cycles in $\mathcal{A}$. Thus, $\mathcal{C} = \mathcal{C}(L) \subseteq I_1$ and $\mathcal{C}$ is also finite.*

*Proof.* Obviously, if we prove that $I_1$ is a finite set, the proof is ended since $\mathcal{C} \subseteq I_1$ for a cycle closed regular language $L$. On the other hand, each $y \in I_1$ is the label of a successful path $\pi$ in the transition diagram of $\mathcal{A}$ such that, for every $q \in Q$, $\pi$ contains at most two occurrences of $q$ (otherwise, $u, z \in A^*$, $c_1, c_2 \in A^+$ and $q \in Q$ exist such that $y = uc_1c_2z$, $\delta(q, c_1) = q$, $\delta(q, c_2) = q$. Thus, using Definition 5.1, $c = c_1c_2$ is not a simple cycle and $y \notin I_1$). Consequently, $|y| \leq 2|Q|$ and $I_1$ is a finite set. □

**Lemma 6.2.** *Let $L$ be a regular language and let $\mathcal{A} = (Q, A, \delta, q_0, F)$ be the minimal finite state automaton recognizing $L$. Let $c \in A^+$ be the label of a closed path in $\mathcal{A}$, i.e., there exists $q \in Q$ such that $\delta(q, c) = q$. Then, $u, v \in A^*$, $d \in A^+$, $q' \in Q$ exist such that $c = udv$, $\delta(q, u) = q' = \delta(q', d)$, $\delta(q', v) = q$ and $d$ is a simple cycle with respect to the transition $\delta(q', d) = q'$.*

*Proof.* Suppose that $c \in A^+$ is the label of a closed path in $\mathcal{A}$, *i.e.*, there exists $q \in Q$ such that $\delta(q, c) = q$. We prove that the property contained in the statement is satisfied for $c \in A^+$, by using induction on $|c|$. If $c$ is a simple cycle, *e.g.* when $|c| = 1$, then the conclusion holds by choosing $u = v = 1$. Otherwise, looking at Definition 5.1 and Proposition 5.1, we have that $y, z \in A^*$, $c' \in A^+$, and $q_1 \in Q$ exist such that $yz \in A^+$, $c = yc'z$, $\delta(q, y) = q_1 = \delta(q_1, c')$, $\delta(q_1, z) = q$ and $c'$ is a cycle with respect to the transition $\delta(q_1, c') = q_1$. Since $|c'| < |c|$, by using the induction hypothesis we have that $u', v' \in A^*$, $d \in A^+$, $q' \in Q$ exist such that $c' = u'dv'$, $\delta(q_1, u') = q' = \delta(q', d)$, $\delta(q', v') = q_1$ and $d$ is a simple cycle with respect to the transition $\delta(q', d) = q'$. Consequently, $c = (yu')d(v'z)$ satisfies the property contained in the statement when we set $u = yu'$ and $v = v'z$. □

**Theorem 6.1.** *Let $X^*$ be a cycle closed star language. Then* $\sim X^* \in Reg^\sim \cap C(Fin, Fin)$.

*Proof.* Let $X^*$ be a cycle closed star language and let $\mathcal{A}$ be the minimal automaton recognizing $X^*$. Let $\mathcal{C} = \mathcal{C}(X^*)$ be the set of the simple cycles in $\mathcal{A}$ and let $I_1$ be the set defined in Lemma 6.1, *i.e.*,

$$I_1 = \{y \in X^* \mid \forall u, v \in A^*, c \in A^+, q \in Q \text{ if } y = ucv$$

$$\text{with } \delta(q_0, u) = q = \delta(q, c), \text{ then } c \text{ is a simple cycle in } \mathcal{A}\}\cdot$$

Let us consider the circular splicing system $SC_{PA} = (A, I, R)$ where $I = {\sim}I_1$ and $R = \{1|1\$1|c;\ c \text{ simple cycle in } \mathcal{A}\}$. Then, $I$ and $R$ are finite sets (Lem. 6.1) and $I = {\sim}I_1 \subseteq {\sim}X^*$. Let us denote $C = C(SC_{PA})$ and let us prove that $C = {\sim}X^*$.

Firstly, we will show that ${\sim}X^* \subseteq C$. Let ${\sim}y \in {\sim}X^*$. By induction on $|y|$, we prove that ${\sim}y \in C$, *i.e.*, ${\sim}y$ is generated by the finite splicing system given above. If ${\sim}y \in I$, then obviously we have ${\sim}y \in C$.

Thus, suppose ${\sim}y \notin I$. Since ${\sim}y \in {\sim}X^*$, each representative of ${\sim}y$ is the label of a successful path in $\mathcal{A}$. Furthermore, since ${\sim}y \notin I$, for each representative $y$ of ${\sim}y$ we have $y \notin I_1$. Consequently, $x, z \in A^*$, $c \in A^+$ exist such that $y = xcz$ with $\delta(q_0, x) = q$, $\delta(q, c) = q$, $\delta(q, z) \in F$ and $c$ is not a simple cycle. Since $c$ is the label of a closed path in $\mathcal{A}$, in view of Lemma 6.2, there exist $u, v \in A^*$, $d \in A^+$, $q' \in Q$ such that $c = udv$, $\delta(q, u) = q' = \delta(q', d)$, $\delta(q', v) = q$ and $d$ is a simple cycle with respect to the transition $\delta(q', d) = q'$.

Consequently, we have $y = xudvz$. Since $y$ is the label of a successful path and $\delta(q', d) = q'$, then $y' = xuvz$ is also the label of a successful path, and, by using induction hypothesis, ${\sim}y' \in C$. Consider the representative $vzxud$ of ${\sim}y$. Set $vzxud = u_2hu_1u_4ku_3$, with $h = vzxu$, $u_1 = u_2 = k = u_3 = 1$, $u_4 = d$. Clearly, $u_1|u_2\$u_3|u_4 = 1|1\$1|d \in R$. Moreover, ${\sim}u_2hu_1 = {\sim}vzxu = {\sim}y' \in C$ and ${\sim}u_4ku_3 = {\sim}d \in C$: by using the splicing definition we have ${\sim}vzxud = {\sim}y \in C$.

*Vice versa*, $C \subseteq {\sim}X^*$ follows by using Proposition 6.1, since $I \subseteq {\sim}X^*$. □

**Remark 6.1.** As already said, in [5], the authors introduced the more involved class of fingerprint closed languages (see Def. 3 in [5]). Furthermore, Theorem 1 in the same paper [5] is the same as Theorem 6.1 for fingerprint closed languages. While the proof of Theorem 1 in [5] is wrong, Theorem 6.1 is an improvement of the above mentioned Theorem 1, since fingerprint closed languages are also cycle closed languages.

**Example 6.2.** We point out that rational languages exist which are not star languages and which are the full linearization of regular circular splicing languages. Indeed, $L = {\sim}(ab)^*$ has $(ab)^* \cup (ba)^*$ as the full linearization, and this is not a star language. On the other hand, ${\sim}(ab)^*$ is a regular circular splicing language since it is generated by $SC_{PA} = (A, I, R)$, where $A = \{a, b\}, I = \{1, {\sim}ab\}, R = \{ab|1\$1|ab\}$. Indeed, for each $n > 1$, $w = {\sim}(ab)^n$ can be obtained by using the rule in $R$ from $\sim ab$ and $\sim(ab)^{n-1}$: an induction over $n$ shows that $L \subseteq C(SC_{PA})$. Once again, using induction over $|w|$, we can prove that $C(SC_{PA}) \subseteq L$: if $w$ is obtained by using the rule in $R$ from $w_1, w_2$, since $|w_1| < |w|$, $|w_2| < |w|$, by induction $w_1, w_2 \in \sim(ab)^*$. Thus, $w_1 =\sim hab, w_2 =\sim kab$ imply $h, k \in (ab)^*$. Finally, $w \in \sim(ab)^*$.

**Example 6.3.** Star languages exist which do not satisfy the hypothesis contained in Theorem 6.1. For instance, consider $L = A^* \setminus a^+ = (a^*ba^*)^*$ over a two-letter alphabet $A = \{a, b\}$. It is clear that $L$ is a star language. Furthermore, consider the minimal automaton $\mathcal{A}$ recognizing $L$ and depicted in Figure 2. We see that $a$ is a cycle in $\mathcal{A}$ and $a^* \cap L = \emptyset$. Thus, $L$ is not a cycle closed language. Now, let $\mathcal{A}$ be a trim deterministic automaton recognizing $X^*$. Let $k$ be a positive integer greater than the number of the states in $\mathcal{A}$ and consider $ba^k \in L$. It is easy to see that there exists $j$ such that $j \le k$ and $a^j$ is a cycle in $\mathcal{A}$. Thus, $L$ is not a cycle closed language with respect to $\mathcal{A}$. However, $\sim L$ is not $C_{PA}$ generated. Indeed, by contradiction, suppose that $C =\sim L = C(SC_{PA})$ with $SC_{PA} = (A, I, R)$. Since $\sim(a^*b) \subseteq C$, $n \in \boldsymbol{N}$ exists such that $w =\sim(a^n b) \in C \setminus I$. Thus, $\sim hu_1u_2 \in C$, $\sim ku_3u_4 \in C$ also exist such that $w =\sim u_2hu_1u_4ku_3$. Since we have only one occurrence of $b$ in $w$, we get $\sim hu_1u_2 \in\sim a^*$ or $\sim ku_3u_4 \in\sim a^*$, which is a contradiction.

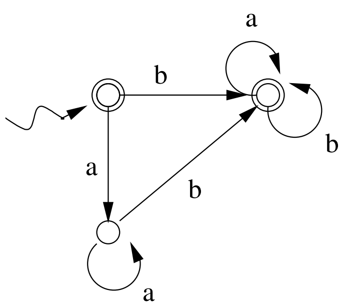


Figure 2. Automaton for $(a^*ba^*)^*$.

**Example 6.4.** On the contrary, $\sim(a^*ba^*)^*$ is $C_{PI}$ generated by $SC_{PI} = (A, I, R)$, with $A = \{a, b\}$, $I = \{\sim b, \sim ab, \sim aab\} \cup 1$ and $R = \{(b, a; b), (a, b; a), (b, b; b), (a, a; a), (ab, b; aa), (b, ab; b)\}$. Indeed, denote $L = (a^*ba^*)^*$ and $r_1 = (b, a; b), r_2 = (a, b; a), r_3 = (b, b; b), r_4 = (a, a; a), r_5 = (ab, b; aa), r_6 = (b, ab; b)$. We immediately observe that we must have $r_2 = \overline{r}_1$, $r_3 = \overline{r}_3$, $r_4 = \overline{r}_4$, $r_6 = \overline{r}_5$. Let us prove that $C(SC_{PI}) \subseteq\sim L$ by using induction on the length of the circular words. Indeed, looking at the form of $r_1, r_2, r_3, r_4$, we see that using these rules and starting from two circular words $\sim\epsilon\alpha$, $\sim\epsilon'\alpha'$ in $\sim L$ we have a circular word $\sim\epsilon\beta\epsilon'\beta' = \sim\epsilon\alpha\epsilon'\alpha'$ which is again in $\sim L$, since $L$ is a submonoid closed under conjugacy relation. Otherwise, if $w =\sim\epsilon\beta\epsilon'\beta'$ is obtained by using $r_5, r_6$ and $\sim\epsilon\alpha$, $\sim\epsilon'\alpha'$ in $\sim L$, we have that $\alpha = ab, \beta = aa$, $\alpha' = b = \beta'$, $\epsilon, \epsilon' \in A^*$. Thus, $\sim\epsilon\beta\epsilon'\beta' = \sim\epsilon aa\epsilon' b \in\sim L =\sim(A^* \setminus a^+)$.

Once again we prove that $^{\sim}L \subseteq C(SC_{PI})$ by using induction over $|w|$, $w \in^{\sim} L$, and we distinguish two cases: $|w|_b = 1$ and $|w|_b > 1$. Now, if $|w|_b = 1$, then $w =^{\sim} a^t b a^{t'}$, $t \geq 2$, $t' \geq 1$ (otherwise, $w \in I$). Thus, $w$ can be obtained from $w_1 =^{\sim} aba^{t-2} \in^{\sim} L$, $w_2 =^{\sim} ba^{t'} \in^{\sim} L$ by using $r_5, \overline{r}_5$. If $|w|_b = m > 1$, then clearly we can write $w = w_1 w_2$, where $|w_1|_b = m_1$, $|w_2|_b = m_2$, $m_1 + m_2 = m$ and $w_1, w_2 \in^{\sim} L$. Now, we can see that rules $r_1, \overline{r}_1$ or $r_3, \overline{r}_3$ or $r_4, \overline{r}_4$ allow us to produce $w$ from $w_1, w_2$.

As far as we know, the structure of regular languages which are closed under the conjugacy relation is unknown, even for the simple case of languages which are the Kleene closure of a regular language. Nevertheless, this structure has been completely described in [3,27] when $X^*$ is a free monoid. The necessary definitions for recalling this result can be found in [2]. We briefly report them below.

An algebraic description of some subclasses of the class of the regular languages $L \subseteq A^*$ has been given by means of the *syntactic monoid* $\mathcal{M}(L)$ of $L$. This is the quotient of $A^*$ with respect to the syntactic congruence $\equiv_L$, defined as follows: $w \equiv_L w'$, with $w, w' \in A^*$, if and only if $xwy \in L \Leftrightarrow xw'y \in L$, for all $x, y \in A^*$. If $L$ is regular, a well known result states that $\mathcal{M}(L)$ is a finite monoid also related to the minimal automaton recognizing $L$.

Historically, this notion arose in the context of variable-length codes. We recall that $X^*$ is a free monoid if and only if $X$ is a *code*, *i.e.* for all $x_1, \ldots, x_n$, $x'_1, \ldots, x'_m \in X$, we have

$$x_1 \cdots x_n = x'_1 \cdots x'_m \Leftrightarrow n = m \text{ and } \forall i \in \{1, \ldots, n\}\ x_i = x'_i.$$

A remarkable class of codes is the class of *biprefix codes* $C$. $C \subseteq A^*$ is biprefix if no word in $C$ is a proper prefix or a proper suffix of another element in $C$, *i.e.* $C \cap CA^+ = C \cap A^+C = \emptyset$. For instance, uniform codes $A^d, d \geq 1$, are biprefix codes.

Here we report two known results on codes. Theorem 6.2 will be used to prove Proposition 6.2. Theorem 6.3 completely describes finite group codes.

**Theorem 6.2** [3,27]**.** *Let $X \subseteq A^+$ be a code. Then $X^*$ is closed under conjugacy relation if and only if $\mathcal{M}(X^*)$ is a group.*

We note that, under the hypothesis that $X$ is a regular language, $\mathcal{M}(X^*)$ is a group if and only if $X$ is a *group code*, *i.e.* a group $G$ and a surjective morphism $\phi$: $A^* \to G$ exist such that $X^* = \phi^{-1}(H)$, where $H$ is a subgroup of $G$ (see [2,3] for this result). Group codes are biprefix codes.

**Theorem 6.3** [3,27]**.** *Let $X \subseteq A^+$ be a group code. $X$ is finite if and only if $X = A^d$, $d \geq 1$.*

**Example 6.5** [2]**.** Group codes exist which are not finite. Let $A = \{a, b\}$ and $M_a = (b^*(ab^*a)^*)^*$ be the set of words with an even number of $a$'s. $M_a$ is generated by $X_a = b \cup ab^*a$, which is a (biprefix) code. Then $M_a = X_a^*$ is a free monoid. Moreover, $X_a$ is a regular group code, where $X_a^* = \phi^{-1}(0)$, $\phi$: $A^* \to \boldsymbol{Z}_2$ is the morphism given by $\phi(a) = 1, \phi(b) = 0$ and with $\{0\}$ being a subgroup of the quotient group $(\boldsymbol{Z}_2, +)$ of the integers modulo 2. Observe that for the submonoid $X^* = \{w \in A^* || w|_a = |w|_b\}$ composed of the words in $A^*$ having as many $a$'s as $b$'s, we have $X^* = \phi^{-1}(0)$ where $\phi: A^* \to \boldsymbol{Z}$ is such that $\phi(a) = 1, \phi(b) = -1$, and with $\{0\}$ a subgroup of $(\boldsymbol{Z}, +)$. $X$ is called a *Dyck code* over $A$ and it is a (non regular) group code.

Finally, Proposition 6.2 shows that the hypothesis contained in Theorem 6.1 is satisfied by regular group codes.

**Proposition 6.2.** *Let $X$ be a regular group code and let $\mathcal{A}$ be a trim deterministic finite state automaton recognizing $X^*$. For every label $c$ of a closed path in $\mathcal{A}$ we have $c \in X^*$.*

*Proof.* Let $X$ be a regular group code and let $\mathcal{A}$ be a trim deterministic finite state automaton recognizing $X^*$. Let $c$ be the label of a closed path in $\mathcal{A}$. Then, $x, y \in A^*$ exist such that $xcy \in X^*$ and $xy \in X^*$. Moreover, since $X^*$ is closed under conjugacy, we have $yxc, yx \in X^*$ and, since $X^*$ is biunitary, we have $c \in X^*$. We recall that a code $X$ is a biprefix code if and only if $X^*$ is biunitary, *i.e.* $X^*$ is both left and right unitary, where a submonoid $X^*$ is right (resp. left) unitary if, for all $u, v \in A^*$, the condition $u, uv \in X^*$ (resp. $u, vu \in X^*$) implies $v \in X^*$ [2]. □

**Remark 6.2.** Thanks to Proposition 6.2, for each regular group code $X$, $X^*$ is cycle closed with respect to each trim deterministic finite state automaton $\mathcal{A}$ recognizing $X^*$.

**Corollary 6.1.** *For each regular group code $X$, $\sim X^* \in Reg^\sim \cap C(Fin, Fin)$.*

**Example 6.6.** Consider $L_1 = \{w \in A^* \mid \exists h, k \in \boldsymbol{N}\ |w|_a = 2k, |w|_b = 2h\}$. Obviously, $L_1 = M_a \cap M_b$, where $M_a, M_b$ are defined in Example 6.5. Thus $L_1$ is a free monoid which is closed under conjugacy, since it is the intersection of two free monoids both closed under conjugacy. $L_1$ is also a regular language, since it is the intersection of two regular languages (Ex. 6.5). Furthermore, since $L_1$ is a free monoid, it follows that $L_1 = X^*$, where $X$ is a code [2]. Moreover, it is well known that when $X$ is a code, $X^*$ is regular if and only if $X$ is regular [2]. Since $X$ is a code which is regular and $X^* = L_1$ is closed under conjugacy relation, by using Theorem 6.2 we have that $X$ is a group code. Consequently, $L_1 = X^*$ is a star language and $\sim L_1 = \sim X^*$ is in $Reg^\sim \cap C(Fin, Fin)$, thanks to Corollary 6.1.

In the proof of Theorem 6.1, we have given the construction of a splicing system $SC_{PA}$ which generates a cycle closed star language. Thanks to Remark 6.2, $L_1$ is a cycle closed star language and the above-mentioned construction for $L_1$ yields $SC_{PA} = (A, I, R')$, where we set $A = \{a, b\}$, $I = \{\sim aa, \sim bb, \sim abab\} \cup 1 = \{\sim aa, \sim bb, \sim abab, \sim baba\} \cup 1$ and $R' = \{1|1\$1|aa, 1|1\$1|bb, 1|1\$1|abab, 1|1\$1|baba\}$ (see also Fig. 3, where the minimal automaton $\mathcal{A}$ recognizing $L_1$ is depicted).

An explicit proof of this result can be also obtained by a slight modification of the argument below which shows that the splicing system $SC_{PA} = (A, I, R)$ generates $\sim L_1$, when we choose $R = \{aa|1\$1|1, bb|1\$1|1, ab|1\$1|1\}$.

Let us prove that $\sim L_1 = C(SC_{PA})$. Since $L_1$ is a star language and $I \subseteq \sim L_1$, we have that $C(SC_{PA}) \subseteq \sim L_1$, by using Proposition 6.1. *Vice versa*, we

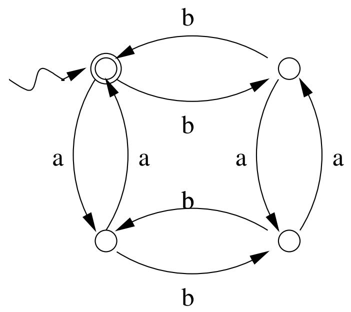


FIGURE 3. Automaton for $L_1$.

show that every $w \in^{\sim} L_1$ is in $C(SC_{PA})$, by using induction on $|w| = n$. We obviously have $w \in^{\sim} L_1 \cap I \subseteq C(SC_{PA})$. Then, suppose that $w \in^{\sim} L_1 \setminus I$ and so $|w|_a = 2k, |w|_b = 2h$, for $h, k \in \boldsymbol{N}$, $|w| \geq 4$. If $aa$ is a factor of $w$ then $w_1, w_2 \in A^+$ exist so that $w =^{\sim} w_1 aa w_2$ with $|w_1|_a + |w_2|_a = |w|_a - 2 = 2k - 2$, $|w_1|_b + |w_2|_b = |w|_b = 2h$. Thus, $^{\sim}w_2w_1 \in^{\sim} L_1$ and $|w_2w_1| < |w|$: by using induction hypothesis, $^{\sim}w_2w_1 \in C(SC_{PA})$. We also have $^{\sim}aa \in I \subseteq C(SC_{PA})$. So, the rule $aa|1\$1|1$, applied to $^{\sim}w_2w_1$ and $^{\sim}aa$, generates $w$. The case in which $bb$ is a factor of $w$ is symmetric to the case above and can be handled with a similar argument.

If neither $aa$ nor $bb$ are factors of $w$, then $w =^{\sim} (ab)^{2k}$ with $k > 1$. On the other hand $^{\sim}abab, ^{\sim}(ab)^{2k-2} \in^{\sim} L_1$. We have $^{\sim}abab \in I \subseteq C(SC_{PA})$ and, by using induction hypothesis, $^{\sim}(ab)^{2(k-1)}$ is in $C(SC_{PA})$: the rule $ab|1\$1|1$, applied to $^{\sim}(ab)^{2(k-1)}$ and $^{\sim}abab$, generates $w$.

Observe that $|R'| > |R|$. Finally, we notice that, by using Remark 3.1, we can prove that $^{\sim}L_1$ is also $C_{PI}$ generated.

As we have already said, we now prove that all star languages $X^*$ with $X$ a finite set have their circularization in $Reg^{\sim} \cap C(Fin, Fin)$. In the proof of this result, we also give a finite splicing system generating $^{\sim}X^*$ the construction of which is simpler than the one given in the proof of Theorem 6.1. For a subset $L$ of $A^*$ we denote $Fact(L)$ the set of all the factors of the elements in $L$.

**Proposition 6.3.** *For all finite sets $X, Y \subseteq A^*$ such that $X^*$ (resp. $X^+$) and $Y$ are closed under the conjugacy relation, with $X \cap Fact(Y) = \emptyset$, we have that $^{\sim}(X^* \cup Y)$ (resp. $^{\sim}(X^+ \cup Y)$) is in $Reg^{\sim} \cap C(Fin, Fin)$.*

*Proof.* It is clear that the finite splicing system $SC_{PA} = (A, I, R)$ generates $^{\sim}(X^* \cup Y)$ (resp. $^{\sim}(X^+ \cup Y)$), when we choose $I = \{^{\sim}x \mid x \in X \cup Y\} \cup 1$ (resp. $I = \{^{\sim}x \mid x \in X \cup Y\}$) and $R = \{x_i|1\$1|x_j \mid x_i, x_j \in X\}$.

Indeed, no rule in $R$ can be applied to circular words $w_1, w_2$ with $w_1 \in^{\sim} Y$ or $w_2 \in^{\sim} Y$. Furthermore, $^{\sim}Y \subseteq I \subseteq C(SC_{PA})$ and we can easily prove that $^{\sim}X^* \subseteq C(SC_{PA})$ (resp. $^{\sim}X^+ \subseteq C(SC_{PA})$). Finally, we can show that $C(SC_{PA}) \setminus ^{\sim}Y = {}^{\sim}X^*$ (resp. $C(SC_{PA}) \setminus ^{\sim}Y =^{\sim} X^+$) with a slight variation of the argument contained in Proposition 6.1. □

All the examples of star languages $X^*$ which are cycle closed and which we have reported above, are such that $X$ is a (regular group) code. As a result, the problems below arise.

**Problem 6.1.** *Do star languages $X^*$ exist which do not satisfy the hypothesis contained in Theorem 6.1 (i.e. which are not cycle closed with respect to $\mathcal{A}$, for all trim deterministic finite automaton $\mathcal{A}$ recognizing $X^*$), with $X$ being not a finite set and such that $\sim X^*$ is $C_{PA}$ generated?*

**Problem 6.2.** *Are there cycle closed star languages $X^*$ which are not generated by a finite set or by a regular group code $X$?*

## 7. The case of a one-letter alphabet

In this section, we will consider the special case of a one-letter alphabet $A = \{a\}$. Here, each language $L \subseteq a^*$ is closed under the conjugacy relation since we have $\sim w = \{w\}$, so we can identify each word $w$ with the circular word $\sim w$ and each language $L$ with its circularization $\sim L$. Using Lemmata 7.2 and 7.3 as preliminary results, we give a characterization of $Reg^{\sim} \cap C(Fin, Fin)$ for one-letter alphabets, and so we provide an answer to Problem 4.1 in this particular case (Prop. 7.1). Observe that circular splicing systems with an initial finite set and an empty set $R$ of rules generates only finite sets. This particular situation is not significant for a characterization of $Reg^{\sim} \cap C(Fin, Fin)$, so in the next part of this section we will often suppose $R \neq \emptyset$. As usual we will refer to Paun's systems, but we end this section with Remark 7.2 regarding Pixton circular splicing systems. For a subset $G$ of $\boldsymbol{N}$, we set $a^G = \{a^g \mid g \in G\}$. In the lemma which follows we gathered results which can be easily proved.

**Lemma 7.1.** *Let $G$ be a finite subset of $\boldsymbol{N}$. Then, we have $1 \in G$ if and only if $(a^G)^+ = a^+$. Furthermore, the following identities between regular expressions hold*

$$a^* = 1 + a + (a^2 + a^3)^+, \quad a^+ = a + (a^2 + a^3)^+.$$

**Lemma 7.2.** *Let $SC_{PA} = (A, I, R)$ be a circular splicing system with $A = \{a\}$, $R = \{r_1, \ldots, r_t\}$ and $r_i = a^{m_{1,i}} | a^{m_{2,i}} \$ a^{m_{3,i}} | a^{m_{4,i}}$, for $i \in \{1, \ldots, t\}$. For all $i \in \{1, \ldots, t\}$, for all $x, y \in a^*$ with $|x| \geq m_{1,i} + m_{2,i}$, $|y| \geq m_{3,i} + m_{4,i}$, the circular word $\sim xy$ is generated by means of the rule $r_i$ and starting from $\sim x$, $\sim y$. As a result, denote $m = \max\, \{m_{1,i} + m_{2,i},\ m_{3,i} + m_{4,i} \mid i \in \{1, \ldots, t\}\}$. For all $x, y \in a^*$, with $|x| \geq m$, $|y| \geq m$, if $\sim x$, $\sim y \in C(SC_{PA})$, then $\sim xy \in C(SC_{PA})$.*

*Proof.* The conclusion follows by using Paun's definition of circular splicing in the special case $A = \{a\}$. □

Some notations and elementary notions from group theory will be used and recalled below (see for instance [14]). Let $\boldsymbol{N}$ be the set of the nonnegative integers, let $n \in \boldsymbol{N}$, $n \geq 2$ and consider the *cyclic group* of order $n$ which we realize, as usual, as the quotient group $\boldsymbol{Z}_n$ of the integers modulo $n$ (*i.e.*, the group of the congruence classes modulo $n$). For subsets $T_1, T_2$ of $\boldsymbol{N}$, $T_1 = T_2 (\mathrm{mod}\ n)$ means that the relation $\{(t_1, t_2) \mid t_1 \in T_1,\ t_2 \in T_2,\ t_1 \equiv t_2 (\mathrm{mod}\ n)\}$ is a bijection between $T_1$ and $T_2$.

For a subset $T$ of $\boldsymbol{N}$, $t \in \boldsymbol{N}$, we set $t + T = \{t + t' \mid t' \in T\}$. Thus, $T$ is a *periodic* subset of $\boldsymbol{Z}_n$ if $t \in \boldsymbol{N}$ exists with $t + T = T\ (\mathrm{mod}\ n)$, $t \not\equiv 0 (\mathrm{mod}\ n)$. In this case $t$ is a *period* of $T$. It is well known that $G'$ is a subgroup of $\boldsymbol{Z}_n$ if and only if there exist $p, r \in \boldsymbol{N}$ such that $n = pr$ and $G' = \{pk \mid k \in \boldsymbol{N},\ 0 \le k \le r - 1\}$. As a result, for each $t \in \boldsymbol{N}$, $t \equiv 0 (\mathrm{mod}\ p)$, we have $G' = t + G' (\mathrm{mod}\ n)$ and, if $G' \neq \{0\}$, $p$ is a period of $G'$. Nevertheless periodic subsets of $\boldsymbol{Z}_n$ exist which are not subgroups. For instance, $\{0, 2, 4\}$ and $\{1, 3, 5\}$ are both periodic subsets of $\boldsymbol{Z}_6$ ($\{0, 2, 4\} + 2 = \{0, 2, 4\}\ (\mathrm{mod}\ 6)$ and $\{1, 3, 5\} + 2 = \{1, 3, 5\}\ (\mathrm{mod}\ 6)$) but the former is a subgroup of $\boldsymbol{Z}_6$ whereas the latter is not a subgroup ($5 + 1 \notin \{1, 3, 5\}$).

**Lemma 7.3.** *Let $SC_{PA} = (A, I, R)$ be a circular splicing system with $A = \{a\}$, $1 \notin I$, $R = \{r_1, \ldots, r_t\}$ and $r_i = a^{m_{1,i}} | a^{m_{2,i}} \$ a^{m_{3,i}} | a^{m_{4,i}}$, for $i \in \{1, \ldots, t\}$. Denote $m = \max\ \{m_{1,i} + m_{2,i},\ m_{3,i} + m_{4,i} \mid i \in \{1, \ldots, t\}\}$, $L = C(SC_{PA})$ and $L_2 = \{w \in L \mid |w| \ge m\}$. Then, $L = L_1 \cup L_2$ where $L_1 = L \setminus L_2$ is a finite subset of $a^+$. Furthermore, either $L_2 = \emptyset$ or there exists $n \in \boldsymbol{N}$, $n \ge 2$, and a subgroup $G'$ of $\boldsymbol{Z}_n$ such that $L_2 = (a^G)^+$, where $G = G' (\mathrm{mod}\ n)$, $a^G \cap Fact(L_1) = \emptyset$ and $n = \min\{m \mid m \in G\}$.*

*Proof.* As a preliminary step, we observe that, if $L = a^+$, then, in view of Lemma 7.1, we have $L = a^+ = a + (a^2 + a^3)^+$. Thus, $L$ satisfies the conditions reported in the statement with $L_1 = \{a\}$, $n = 2$, $G' = \{0, 1\}$, $G = \{2, 3\}$.

Let us denote $m = \max\ \{m_{1,i} + m_{2,i},\ m_{3,i} + m_{4,i} \mid i \in \{1, \ldots, t\}\}$, $L = C(SC_{PA})$ and $L_2 = \{w \in L \mid |w| \ge m\}$. Then, we obviously have $L = L_1 \cup L_2$ where $L_1 = L \setminus L_2$ is a finite subset of $a^*$.

If $L_2 = \emptyset$, the proof is ended. Otherwise, let $n \in \boldsymbol{N}$ with $n = \min\{|l| \mid l \in L_2\}$. We have $n \ge 1$ since $1 \notin L$ (see Lem. 2.1). For all $j \in \{0, \ldots, n-1\}$ such that $(a^n)^* a^j \cap L_2 \neq \emptyset$, let $h_j = \min\{h \mid a^{nh+j} \in L_2\}$. We have already observed that $h_0 = 1$ and $h_j > 0$, for all $j$'s. Let $G \subseteq \boldsymbol{N}$ with $G = \{h_j n + j \mid j \in \{0, \ldots, n-1\},\ (a^n)^* a^j \cap L_2 \neq \emptyset\}$.

Let us prove that $L_2 = (a^G)^+$. We show that $(a^G)^+ \subseteq L_2$, by using induction on the length of $a^g \in (a^G)^+$. Indeed, either $a^g \in a^G \subseteq L_2$ or $a^g = a^{g_1} a^{g_2}$ with $a^{g_1} \in (a^G)^+$, $a^{g_2} \in (a^G)^+$ and $|a^{g_1}| < |a^g|$, $|a^{g_2}| < |a^g|$. In the last case, by using induction hypothesis, $a^{g_1}, a^{g_2} \in L_2$ and, by using Lemma 7.2, since $|a^{g_1}| \ge m$, $|a^{g_2}| \ge m$, we have that $a^g = a^{g_1} a^{g_2} \in L_2$. So, $(a^G)^+ \subseteq L_2$. In order to prove that $L_2 \subseteq (a^G)^+$, let $a^l \in L_2$ and let $h, j$ be such that $l = nh + j$, with $h \in \boldsymbol{N}$ and $j \in \{0, \ldots, n-1\}$. We already know that $h \ge h_j$. Since $a^{h_j n + j} \in (a^G)^+$ and $a^n \in (a^G)^+$, we have $a^{h_j n + j} (a^n)^* \subseteq (a^G)^+$ and so $a^l = a^{nh+j} \in a^{h_j n + j} (a^n)^* \subseteq (a^G)^+$. Clearly we have $a^G \cap Fact(L_1) = \emptyset$.

Finally, if $n = 1$ then $L_1 = \emptyset$, $1 \in G$ and $L = (a^G)^+ = a^+$ satisfies the conditions reported in the statement, as already observed above. Otherwise, $n \ge 2$ and let us prove that $G$ is a set of representatives of the elements of a subset $G'$ of $\boldsymbol{Z}_n$ where $G'$ is a (periodic) subgroup of the finite cyclic group of order $n$.

We already know that $n \in G$. Furthermore, for each $g_1, g_2 \in G$, we have $a^{g_1} a^{g_2} \in (a^G)^+ = L_2$. Thus, for $h \in \boldsymbol{N}$ and $k \in \{0, \ldots, n-1\}$ such that

$g_1 + g_2 = hn + k$, we have $h \geq h_k$ with $h_k n + k \in G$. So, denote $[t]$ the class modulo $n$ of $t \in \{0, \ldots, n-1\}$ and let $G'$ be defined as the set of the (different) classes modulo $n$ of the elements in $G$. We know that $[0] \in G'$. Furthermore, for all $[g'_1], [g'_2] \in G'$, let $g_1, g_2$ the corresponding elements in $G$. We have proved that $g \in G$ exist so that $g \equiv g_1 + g_2 (\text{mod } n)$, *i.e.*, $[g'_1 + g'_2] = [g'_1] + [g'_2] \in G'$. Thus $G'$ is a submonoid of a finite group (namely $\boldsymbol{Z}_n$): a classical result from group theory states that $G'$ is a subgroup (of $\boldsymbol{Z}_n$, indeed for each $[g] \in G'$, $G'$ being a submonoid, we have $(n-1)[g] \in G'$ and $[(n-1)g] = (n-1)[g]$ is the inverse of $[g]$). □

**Lemma 7.4.** *Let $SC_{PA} = (A, I, R)$ be a circular splicing system with $A = \{a\}$, $R = \{r_1, \ldots, r_t\}$ and $r_i = a^{m_{1,i}} | a^{m_{2,i}} \$ a^{m_{3,i}} | a^{m_{4,i}}$, for $i \in \{1, \ldots, t\}$. Denote $m = \max \{m_{1,i} + m_{2,i},\ m_{3,i} + m_{4,i} \mid i \in \{1, \ldots, t\}\}$, $L = C(SC_{PA})$ and $L_2 = \{w \in L \mid |w| \geq m\}$. Then, $L = L_1 \cup L_2$ where $L_1 = L \setminus L_2$ is a finite subset of $a^*$. Furthermore, either $L_2 = \emptyset$ or there exists $n \in \boldsymbol{N}$, $n \geq 2$, and a subgroup $G'$ of $\boldsymbol{Z}_n$ such that $L_2 = (a^G)^+$, where $G = G'(\text{mod } n)$, $a^G \cap Fact(L_1) = \emptyset$ and $n = \min\{m \mid m \in G\}$.*

*Proof.* The conclusion holds when either $L = \{1\}$ or $L \subseteq a^+$, in the latter case by using Lemmata 7.3 and 2.1. Otherwise, set $L' = L \setminus \{1\}$. By using once again Lemma 2.1, we have that $L'$ is $C_{PA}$ generated. Then, in view of Lemma 7.3, $L' = L'_1 \cup L'_2$ where $L'_1 = L' \setminus L'_2$ is a finite subset of $a^+$. Furthermore, either $L'_2 = \emptyset$ or there exists $n \in \boldsymbol{N}$, $n \geq 2$, and a subgroup $G'$ of $\boldsymbol{Z}_n$ such that $L'_2 = (a^G)^+$, where $G = G'(\text{mod } n)$, $a^G \cap Fact(L'_1) = \emptyset$ and $n = \min\{m \mid m \in G\}$. Thus, it is easy to prove that $L = L_1 \cup L_2$ satisfies the condition reported in the statement when we set $L_1 = L'_1 \cup \{1\}$ and $L_2 = L'_2$. □

**Proposition 7.1.** *A subset $L = {}^\sim L$ of $a^*$ is $C_{PA}$ generated if and only if either $L$ is a finite set or there exist a finite subset $L_1$ of $a^*$, positive integers $p, r, n$, with $n = pr \geq 2$ and a subgroup $G' = \{pk \mid k \in \boldsymbol{N},\ 0 \leq k \leq r-1\}$ of $\boldsymbol{Z}_n$ such that $L = L_1 \cup (a^G)^+$, where $G = G'(\text{mod } n)$, $a^G \cap Fact(L_1) = \emptyset$ and $n = \min\{m \mid m \in G\}$.*

*Proof.* Thanks to Proposition 6.3 and to Lemma 7.4, the conclusion easily follows. □

A simpler characterization of the $C_{PA}$ generated circular languages over $\{a\}$ is given below.

**Corollary 7.1.** *A subset $L$ of $a^*$ is $C_{PA}$ generated if and only if there exist finite subsets $L_1$, $a^G$ of $a^*$ and a positive integer $n$ such that $L = L_1 \cup (a^G)^+$ and, when $a^G \neq \emptyset$ and $L_1 \neq \emptyset$, $\max\{|l| \mid l \in L_1\} < n = \min\{|l| \mid l \in a^G\}$.*

*Proof.* Thanks to Propositions 6.3 and 7.1, the conclusion easily follows. □

**Example 7.1.** Let $L = \{a^3, a^4\} \cup \{a^6, a^{14}, a^{16}\}^+$. Then $L$ satisfies the hypothesis contained in Proposition 7.1. Following the proof of Proposition 6.3, we have that $L$ is generated by the splicing system $SC_{PA} = (\{a\}, I, R)$, where $I = \{a^3, a^4, a^6, a^{14}, a^{16}\}$ and $R = \{a^6|1\$1|a^6, a^6|1\$1|a^{14},\ a^6|1\$1|a^{16},\ a^{14}|1\$1|a^{14}, a^{14}|1\$1|a^{16},\ a^{16}|1\$1|a^{16}\}$.

We end this section with some observations. We begin with a result concerning the descriptional complexity of a circular splicing system which generates a circular language $L \subseteq a^*$ (Prop. 7.2). We then make a comparison between the two cases $|A| = 1$ and $|A| > 1$, with respect to Problem 4.1 for Paun's systems (Rem. 7.1). Comments on Pixton's systems with $|A| = 1$ are given in Remark 7.2.

**Proposition 7.2.** *Let $L \subseteq a^*$ be a $C_{PA}$ generated language. Then, there exists a (minimal) splicing system $(\{a\}, I, R)$ generating $L$ with either $R = \emptyset$ or $R = \{r\}$ containing only one rule.*

*Proof.* Let $L \subseteq a^*$ be a $C_{PA}$ generated language. Then, using Corollary 7.1 and with the same notations, $L = L_1 \cup (a^G)^+$ and a positive integer $n$ exists such that, when $a^G \neq \emptyset$, $n = \min\{|l| \mid l \in a^G\}$, $L_1 = \{l \in L \mid |l| < n\}$. If $a^G = \emptyset$ then the circular splicing system $(\{a\}, I, R)$ where $R = \emptyset$ and $I = L_1$ obviously generates $L = L_1$.

Thus, suppose that $a^G \neq \emptyset$. Consider the circular splicing system $(\{a\}, I, R)$ where $I = L_1 \cup a^G$ and $R = \{a^n \mid 1 \$ 1 \mid a^n\}$. Obviously, $I$ is a finite set and $I \subseteq L$.

Let us prove that $L = C(SC_{PA})$. We show that $L \subseteq C(SC_{PA})$. Obviously, $I \subseteq C(SC_{PA})$ and, for each $a^g \in L \setminus I$, we prove that $a^g \in C(SC_{PA})$ by using induction on $|a^g|$. Since $a^g \in L \setminus I$, then $a^g \in (a^G)^+ \setminus a^G$ which in turn implies $a^g = a^{g_1}a^{g_2}$ with $a^{g_1} \in (a^G)^+$, $a^{g_2} \in (a^G)^+$, $|a^{g_1}| < |a^g|$, $|a^{g_2}| < |a^g|$. Furthermore, $a^{g_1} \in (a^G)^+$ implies $g_1 \geq n$ and, analogously, $a^{g_2} \in (a^G)^+$ implies $g_2 \geq n$. By using induction hypothesis, $a^{g_1}, a^{g_2} \in C(SC_{PA})$ and, by using Lemma 7.2, $a^g = a^{g_1}a^{g_2} \in C(SC_{PA})$.

We now prove that $C(SC_{PI}) = \bigcup_{i \geq 0} \sigma^i(I) \subseteq L$, with a slight variation of the argument contained in Proposition 6.1 and by using induction on the minimal $i$ such that $a^g \in \sigma^i(I)$. Let $a^g \in \sigma^i(I)$ and let $h, j$ be such that $g = nh + j$, with $h \in \boldsymbol{N}$ and $j \in \{0, \ldots, n-1\}$. Since $\sigma^0(I) = I \subseteq L$, we can suppose $i > 0$. Furthermore, since $\sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$ and by using induction hypothesis, we can suppose that $a^g$ has been generated by using the rule in $R$ and starting from two words $a^{nh_1+j_1}, a^{nh_2+j_2} \in \sigma^{i-1}(I)$. Then, $h_1 \geq 1$, $h_2 \geq 1$ (otherwise the rule in $R$ cannot be applied) and $a^g = a^{nh_1+j_1}a^{nh_2+j_2}$. By using induction hypothesis, $a^{nh_1+j_1}, a^{nh_2+j_2} \in L$. Then $a^{nh_1+j_1}, a^{nh_2+j_2} \in (a^G)^+$ (since $h_1, h_2 \geq 1$), and thus $a^g = a^{nh_1+j_1}a^{nh_2+j_2} \in (a^G)^+$, since $(a^G)^+$ is a submonoid. $\square$

**Example 7.2.** We already know that $\sim L = L = \{a^3, a^4\} \cup \{a^6, a^{14}, a^{16}\}^+$ is in $Reg^\sim \cap C(Fin, Fin)$ (Ex. 7.1). In virtue of Proposition 7.2, $L = \{a^3, a^4\} \cup \{a^6, a^{14}, a^{16}\}^+$ has a minimal circular splicing system which generates it, namely, $SC_{PA} = (\{a\}, I, R)$ where $I = \{a^3, a^4, a^6, a^{14}, a^{16}\}$ and $R = \{a^n|1\$1|a^n\} = \{a^6|1\$1|a^6\}$. Looking at the proof of Proposition 7.2, and using the same argument as in this proof, we can see that another minimal splicing system can be defined, namely, $SC_{PA} = (\{a\}, I, R)$, where $I = \{a^3, a^4, a^6, a^{12}, a^{14}, a^{16}, a^{18}, a^{20}, a^{22}\}$ and $R = \{a^{12}|1\$1|a^{12}\}$. As a matter of fact, if $L$ satisfies the hypothesis of Proposition 7.1, for every $h \geq 1$, the circular splicing system $(\{a\}, I_h, R_h)$, where $R_h = \{a^{hn}|1\$1|a^{hn}\}$, generates $L$ if we choose a sufficiently large $I_h$.

**Example 7.3.** As another example, let us consider the rational language $L = \{a^3, a^7\} \cup \{a^4, a^5\}^+$. We can see that even if $L$ is expressed in a form that does not correspond to the hypothesis of Proposition 7.1, $L$ is $C_{PA}$ generated by choosing $SC_{PA} = (\{a\}, I, R)$, where $I = \{a^3, a^4, a^5, a^7, a^8, a^9, a^{10}, a^{12}, a^{13}, a^{14}, a^{15}\}$ and $R = \{a^8|1\$1|a^8\}$. Furthermore, following the proof of Lemma 7.3, we have $L = \{a^3, a^4, a^5, a^7\} \cup \{a^8, a^9, a^{10}, a^{12}, a^{13}, a^{14}, a^{15}\}^+ = \{a^3, a^4, a^5, a^7\} \cup \{a^8, a^9, a^{10}, a^{12}, a^{13}, a^{14}, a^{15}, a^{19}\}^+$ and in this form, $L$ satisfies the hypothesis of Proposition 7.1.

A decidability question naturally follows on from the last example: given a language $L$, with $L \subseteq a^*$, can we decide whether $L$ is a $C_{PA}$ generated language? If no hypothesis is made over $L$, the answer to this question is no, thanks to the Rice theorem [21]. On the contrary, in [10] the authors proved that this question is decidable when we restrict ourselves to regular languages.

**Remark 7.1.** The computational power of Paun circular splicing systems dramatically decreases when we restrict ourselves to alphabets of cardinality one. Indeed, in Proposition 7.1, we have proven that in this case $Reg^{\sim} \cap C(Fin, Fin) = C(Fin, Fin)$ is a class of languages satisfying the hypotheses of Proposition 6.3 since each $L \in C(Fin, Fin)$ has the form $L = L_1 \cup (a^G)^+$, $L_1$, $G$ being finite sets and $a^G \cap Fact(L_1) = \emptyset$. We already know that there exist languages $L$ having this form and which are not cycle closed with respect to the minimal automaton recognizing $L$ [10]. Furthermore, with the same notations as in Proposition 7.1 and in virtue of Theorem 6.3, $a^G$ is a group code (and consequently $(a^G)^*$ is generated by a group code) if and only if $a^G = a^n$, *i.e.*, $(a^n)^* a^j \cap (a^G)^+ = \emptyset$, for all $j \in \{1, \ldots, n-1\}$.

**Remark 7.2.** Languages exist which are $C_{PI}$ generated but not $C_{PA}$ generated even in the case of a one-letter alphabet (thus showing already in this case that Problem 3.1 does not have a positive answer). Indeed, we will see that $cyclic(aa) = (a^2)^* a$ is $C_{PI}$ generated (Prop. 8.6) but $cyclic(aa)$ is not $C_{PA}$ generated (Ex. 8.3). As in the case of a Paun system each rule in a Pixton system can be applied to a pair of words $x, y$ provided that $|x|, |y|$ are large enough.

## 8. Regular $C_{PA}$ generated languages: cyclic and weak cyclic languages

We have already seen that regular circular $C_{PA}$ generated languages exist which are not the circularization of a star language, namely $^{\sim}(ab)^*$ (see Ex. 6.2). The aim of this section is to give other examples of such languages. Precisely, we construct a class of these languages called *cyclic languages* (Def. 8.1). After some preliminary definitions, we prove this result in Propositions 8.2 and 8.3. The same family of cyclic languages allows us to give a negative answer to the natural question whether the full linearization of a regular circular $C_{PA}$ generated language is always cycle closed (Prop. 8.4). As usual, in this section we implicitly suppose that Paun's definition for circular splicing is adopted.

In this section we will often consider prefixes (resp. proper prefixes) $x$ of $w$ which are not the empty word. Thus, we set $Pref(w) = \{x \in A^+ \mid \exists y \in A^+ : xy = w\}$ and $Suff(w) = \{x \in A^+ \mid \exists y \in A^+ : yx = w\}$. Proper prefixes of the words $w'$ such that $w' \sim w$ are also needed, and so we set $Pref_c(w) = \{x \in A^+ \mid \exists w' \sim w : x \in Pref(w')\}$. Finally, a word $w \in A^*$ is *unbordered* if, for each $x \in A^+$, $w \notin xA^*x$. As a matter of fact, in [12], an unbordered word is defined as a word satisfying the property contained in Lemma 8.1 the proof of which is reported below for the sake of completeness. The following proposition is also needed.

**Proposition 8.1** [23]**.** *Two words $x, y \in A^+$ are conjugate if and only if there exists $z \in A^*$ such that $xz = zy$. More precisely, the above equality holds if and only if there exist $u, v \in A^*$ such that $x = uv$, $y = vu$, $z \in u(vu)^*$.*

**Lemma 8.1.** *For every unbordered word $w$ with $|w| \geq 2$, we have $Pref(w) \cap Suff(w) = \emptyset$.*

*Proof.* By contradiction, suppose that $z \in Pref(w) \cap Suff(w)$. Then, there exist $x, y \in A^+$ such that $xz = w = zy$. Thus, by using Proposition 8.1, words $u, v \in A^*$ exist and a nonnegative integer $t$ also exists such that $x = uv$, $y = vu$, $z = u(vu)^t$. Furthermore, since $z \neq 1$, we have either $u \neq 1$ or $v \neq 1$, $t \geq 1$. Consequently we have $w = u(vu)^{t+1}$ with $u \neq 1$ or $v \neq 1$, $t \geq 1$, which is in contradiction with the hypothesis of $w$ being unbordered. □

**Definition 8.1** *Cyclic languages***.** For each $w \in A^+$, the cyclic language $cyclic(w)$ is defined as follows:

$$cyclic(w) = \,^{\sim}(w^* Pref_c(w)) = \cup_{p \in Pref_c(w)} \,^{\sim}(w^* p).$$

**Remark 8.1.** Notice that, for each $w \in A^+$, $cyclic(w)$ is a regular circular language. Indeed, we have

$$\begin{aligned} Lin(cyclic(w)) = & \sum_{p_1 p_2 = p,\; p \in Pref_c(w)} p_2 w^* p_1 \\ & + \sum_{w_1 w_2 = w,\; p \in Pref_c(w)} w_2 w^* p w^* w_1. \end{aligned}$$

**Example 8.1.** Let $w = abc$. Then, $Pref_c(abc) = \{a, ab, b, bc, c, ca\}$ and

$$\begin{aligned} cyclic(w) = \; & ^{\sim}((abc)^* a) \cup \,^{\sim}((abc)^* ab) \cup \,^{\sim}((abc)^* b) \cup \,^{\sim}((abc)^* bc) \\ & \cup \,^{\sim}((abc)^* c) \;\cup\; ^{\sim}((abc)^* ca). \end{aligned}$$

**Remark 8.2.** Notice that words $w, w'$ exist such that $w \sim w'$ and $cyclic(w) \neq cyclic(w')$. For example, $cyclic(abc) \neq cyclic(cab)$ since $abca \in cyclic(abc) \setminus cyclic(cab)$ (if $abca \in cyclic(cab)$ then we would also have $abca \sim caba$ which is impossible). Analogously, $cyclic(aba) \neq cyclic(baa)$ since $abab \in cyclic(aba) \setminus cyclic(baa)$.

The technical lemma below will be useful for proving our main result.

**Lemma 8.2.** *For every $x, w \in A^+$, $x \in Pref_c(w)$ if and only if either $x$ is a proper factor of $w$ or $x = sp$, with $s$ being a suffix of $w$, $p$ being a prefix of $w$ and $|sp| < |w|$. Furthermore, for every $x \in Pref_c(w)$, $x_1, x_2 \in A^+$, if $x = x_1x_2$ then $x_1, x_2 \in Pref_c(w)$.*

*Proof.* Let $x, w \in A^+$. Suppose that either $x$ is a proper factor of $w$ or $x = sp$, with $s$ being a suffix of $w$, $p$ being a prefix of $w$ and $|sp| < |w|$. Then, obviously we have $x \in Pref_c(w)$.

Conversely, suppose that $x \in Pref_c(w)$. Thus, there exist $u \in A^+$, $y_1, y_2 \in A^*$ such that $w = y_1y_2$, $xu = w' = y_2y_1 \sim w$. Two cases can occur: either $|y_2| \geq |x|$ or $|y_2| < |x|$.

In the first case, there exists $t \in A^*$ such that $y_2 = xt$. Consequently, we have $w = y_1xt$ with $y_1t \neq 1$ (otherwise $x = w = w'$) and $x$ is a proper factor of $w$. In the second case, there exists $t \in A^+$ such that $x = y_2t$. Consequently, we have $y_1 = tu$ and $w = tuy_2$. So $x = sp$, with $s = y_2$ being a suffix of $w$, $p = t$ being a prefix of $w$ and $|sp| < |w|$ (since $u \neq 1$).

In order to prove the second part of the statement, suppose that $x = x_1x_2$ with $x \in Pref_c(w)$, $x_1, x_2 \in A^+$. By using the first part of the statement, we have that either $x$ is a proper factor of $w$ or $x = sp$, with $s$ being a suffix of $w$, $p$ being a prefix of $w$ and $|sp| < |w|$. In the first case $x_1, x_2$ are also proper factors of $w$ and so $x_1, x_2 \in Pref_c(w)$. In the second case, we have $x_1x_2 = sp$. Consequently, either $x_1$ is a prefix of $s$ or $x_2$ is a suffix of $p$. If $x_1$ is a prefix of $s = x_1s'$ then $x_1$ is a proper factor of $w$ and we have $x_2 = s'p$, with $s'$ being a suffix of $w$, $p$ being a prefix of $w$ and $|s'p| < |sp| < |w|$. So, we have $x_1, x_2 \in Pref_c(w)$. Otherwise, $x_2$ is a suffix of $p = p'x_2$ and $x_1 = sp'$, with $s$ being a suffix of $w$, $p'$ being a prefix of $w$ and $|sp'| < |sp| < |w|$ and once again, $x_2$ being a proper factor of $w$, we have $x_1, x_2 \in Pref_c(w)$. □

We will now prove the main result of this section, namely that $cyclic(w)$ is in $Reg^\sim \cap C(Fin, Fin)$ provided that $w$ is an unbordered word with $|w| > 2$. The proof of this result is broken up into several intermediate results. Lemmata 8.3 and 8.4 allow us to find a finite circular splicing system such that all the circular words in $cyclic(w)$ are generated by using its rules, as proved in Lemma 8.5. Next, in Lemma 8.8 (by using Lemmata 8.1, 8.6 and 8.7 as preliminary results), we prove that if a circular word $^\sim x$ is generated by the above-mentioned finite circular splicing system then $^\sim x \in cyclic(w)$.

Let us briefly sketch what we will prove in Lemmata 8.3 and 8.4. We must generate all the words with the form $w^tp_g$ where $t$ is a positive integer and $p_g \in Pref_c(w)$. In the next part of this section we will suppose that $|p_g| = g$. In Lemma 8.3 we consider the case $g = 1$, whereas the case $g > 1$ is examined in Lemma 8.4.

Intuitively, a circular word $^\sim x$ is generated by a rule $r$ in a circular splicing system if one of the representatives of $^\sim x$ is the concatenation of two other words each being in turn a representative of a generated word and having a suffix and a prefix related to the rule $r$. If $w = ps$ is a factorization of $w$, we will show in Lemma 8.3 that we can define splicing rules such that

(1) $w^t p_1 \sim (w^g p_1 p)(sw^{g'})$ and $\sim w^t p_1$ is generated starting from $\sim w^g p_1 p$, $\sim sw^{g'} \in cyclic(w)$, if $p_1$ is a suffix of $w$;
(2) $w^t p_1 \sim (w^g p)(sw^{g'} p_1)$ and $\sim w^t p_1$ is generated starting from $\sim w^g p$, $\sim p_1 sw^{g'} \in cyclic(w)$, if $p_1$ is a factor of $w$, with the additional hypothesis that $p_1 s$ is once again a suffix of $w$;
(3) $w^t p_1 \sim (w^g p)(sp_1 w^{g'})$ and $\sim w^t p_1$ is generated starting from $\sim w^g p$, $\sim sp_1 w^{g'} \in cyclic(w)$, if $p_1$ is a prefix of $w$.

**Lemma 8.3.** *Let $w \in A^+$ be an unbordered word with $|w| > 2$. Let $p_1 \in Pref_c(w)$ (i.e., let $p_g \in Pref_c(w)$ with $g = 1$). Then, either $p_1 \in Pref(w) \setminus Suff(w)$ or $p_1 \in Suff(w) \setminus Pref(w)$ or $p', s \in A^+$ exist such that $w = p' p_1 s$. Furthermore, the following statements hold.*

(1) *If $p_1 \in Suff(w) \setminus Pref(w)$, let $p, s \in A^+$ be such that $w = ps$ and $|s| \geq 2$ (so, $|p_1 p| < |w|$). Then, the circular word $\sim w^t p_1 \in cyclic(w)$ is generated by means of the rule*

$$R_{p_1} = wp_1 p \mid w \ \$ \ w \mid sw$$

*and starting from $\sim w^{t_1 - 1} p_1 pw \in cyclic(w)$ and $\sim w^{t_2 - 1} sw \in cyclic(w)$, with $t_1 + t_2 = t - 1$, $t_1 \geq 2$, $t_2 \geq 2$.*
(2) *If $p', s \in A^+$ exist such that $w = p' p_1 s$ (so $p_1 s$ is once again a proper suffix of $w$), let $p = p' p_1$. Then, the circular word $\sim w^t p_1 \in cyclic(w)$ is generated by means of the rule*

$$R_{p_1} = wp \mid w \ \$ \ wp_1 \mid sw$$

*and starting from $\sim w^{t_1 - 1} pw \in cyclic(w)$ and $\sim w^{t_2 - 1} p_1 sw \in cyclic(w)$, with $t_1 + t_2 = t - 1$, $t_1 \geq 2$, $t_2 \geq 2$.*
(3) *If $p_1 \in Pref(w) \setminus Suff(w)$, let $p, s \in A^+$ be such that $w = ps$ and $|p| \geq 2$ (so $|sp_1| < |w|$). Then, the circular word $\sim w^t p_1 \in cyclic(w)$ is generated by means of the rule*

$$R_{p_1} = wp \mid w \ \$ \ w \mid sp_1 w$$

*and starting from $\sim w^{t_1 - 1} pw \in cyclic(w)$ and $\sim w^{t_2 - 1} sp_1 w \in cyclic(w)$, with $t_1 + t_2 = t - 1$, $t_1 \geq 2$, $t_2 \geq 2$.*

*Proof.* The first part of the statement easily follows by using Lemmata 8.1 and 8.2. Let $t_1, t_2$ be positive integers with $t_1 \geq 2$, $t_2 \geq 2$. Let $p_1 \in Suff(w) \setminus Pref(w)$ and let $p, s \in A^+$ be such that $w = ps$, with $|s| \geq 2$ (so, $|p_1 p| < |w|$). Obviously, once again by using Lemma 8.2, $\sim w^{t_1 - 1} p_1 pw \in cyclic(w)$ and $\sim w^{t_2 - 1} sw \in cyclic(w)$. Suppose now that $p', s \in A^+$ exist such that $w = p' p_1 s$ (so $p_1 s$ is once again a proper suffix of $w$) and let $p = p' p_1$. Then, by using the same Lemma 8.2 we can conclude that $\sim w^{t_1 - 1} pw \in cyclic(w)$ and $\sim w^{t_2 - 1} p_1 sw \in cyclic(w)$. Finally, let $p_1 \in Pref(w) \setminus Suff(w)$, let $p, s \in A^+$ be such that $w = ps$ and $|p| \geq 2$ (so $|sp_1| < |w|$). Then, once again in virtue of Lemma 8.2, ${}^\sim w^{t_1-1}pw \in cyclic(w)$ and ${}^\sim w^{t_2-1}sp_1w \in cyclic(w)$. Thus, by using Paun's definition of circular splicing, it is easy to see that the conclusion holds. □

Lemma 8.4 below deals with rules for generating circular words ${}^\sim w^t p_g$ where $t$ is a positive integer and $p_g \in Pref_c(w)$, $g > 1$. In the proof of this lemma we will also use a part of the statement contained in Lemma 8.2. If $p_g = ps$ is a factorization of $p_g$, we will show in Lemma 8.4 that we can define splicing rules such that $w^t p_g \sim (w^g p)(sw^{g'})$ and ${}^\sim w^t p_g$ is generated starting from ${}^\sim w^g p$ and ${}^\sim sw^{g'}$.

**Lemma 8.4.** *Let $w \in A^*$ be a word with $|w| > 2$. Let $p_g \in Pref_c(w)$ with $g > 1$ and let $p, s \in A^+$ be such that $p_g = ps$. The circular word ${}^\sim w^t p_g \in cyclic(w)$ is generated by means of the rule*

$$R_{p_g} = wp \mid w \ \$ \ w \mid sw$$

*and starting from ${}^\sim w^{t_1-1}pw \in cyclic(w)$ and ${}^\sim w^{t_2-1}sw \in cyclic(w)$, with $t_1+t_2 = t$, $t_1 \geq 2$, $t_2 \geq 2$.*

*Proof.* Let $t_1, t_2$ be positive integers with $t_1 \geq 2$, $t_2 \geq 2$ and let $p, s \in A^+$ be such that $p_g = ps$. By using Lemma 8.2, we have ${}^\sim w^{t_1-1}pw \in cyclic(w)$ and ${}^\sim w^{t_2-1}sw \in cyclic(w)$. Then, by using Paun's definition of circular splicing, it is easy to see that the conclusion holds. □

**Lemma 8.5.** *Let $w \in A^*$ be an unbordered word with $|w| > 2$, let $SC_{PA} = (A, I, R)$ be the circular splicing system defined by*

$$I = \{{}^\sim(w^t Pref_c(w)) \mid 0 \leq t \leq 4\}, \quad R = \cup_{p_g \in Pref_c(w)} R_{p_g}$$

*where the $R_{p_g}$'s are defined according to Lemmata 8.3 and 8.4. Then, $cyclic(w) \subseteq C(SC_{PA})$.*

*Proof.* We prove that ${}^\sim w^t Pref_c(w) \subseteq C(SC_{PA})$ by induction on $t$. If $t \in \{0, 1, 2, 3, 4\}$ we have ${}^\sim w^t Pref_c(w) \subseteq I$ and so ${}^\sim w^t Pref_c(w) \subseteq C(SC_{PA})$. Let us suppose that ${}^\sim w^j Pref_c(w) \subseteq C(SC_{PA})$, for $0 \leq j < t$, $t \geq 5$, and let us prove that ${}^\sim w^t Pref_c(w) \subseteq C(SC_{PA})$. Indeed, for all $p_g \in Pref_c(w)$ with $g = 1$ (respectively $g > 1$), in virtue of Lemma 8.3 (resp. Lem. 8.4) the circular word ${}^\sim w^t p_g \in cyclic(w)$ is generated by means of the rule $R_{p_g}$, starting from ${}^\sim w^{t_1-1}p'w \in cyclic(w)$ and ${}^\sim w^{t_2-1}s'w \in cyclic(w)$, with $t_1 + t_2 = t - 1$ (resp. $t_1 + t_2 = t$), $t_1 \geq 2$, $t_2 \geq 2$. By induction hypothesis over $t$, ${}^\sim w^{t_1-1}p'w \in C(SC_{PA})$ and ${}^\sim w^{t_2-1}s'w \in C(SC_{PA})$. Consequently, ${}^\sim w^t p_g \in C(SC_{PA})$, by definition of circular splicing. □

**Example 8.2.** Let us consider $cyclic(abc)$ (Ex. 8.1). By using Lemma 8.5, each word in $cyclic(abc)$ is generated by the circular splicing system $SC_{PA} = (\{a, b, c\}, I, R)$ defined by $I = \{{}^\sim((abc)^t Pref_c(abc)) \mid 0 \leq t \leq 4, \}$, $R = R_a \cup R_b \cup R_c \cup R_{ab} \cup R_{bc} \cup R_{ca}$, where $w = abc$, $R_a = wab|w\$w|caw$, $R_b = wab|w\$wb|cw$,

$R_c = wca|w\$w|bcw$, $R_{ab} = wa|w\$w|bw$, $R_{bc} = wb|w\$w|cw$, $R_{ca} = wc|w\$w|aw$. Thanks to Lemma 8.8, we will see that $cyclic(abc) = C(SC_{PA})$.

Using the intermediate results reported below, we now prove that every word generated by the circular splicing system defined in Lemma 8.5, is an element of $cyclic(w)$.

**Lemma 8.6.** *Let $w \in A^*$ be an unbordered word with $|w| \geq 2$. For all $p \in A^+$ with $|p| < |w|$, we have $w \notin Suff(w)p$.*

*Proof.* By contradiction, suppose that there exist $p \in A^+$ with $|p| < |w|$ and $s \in Suff(w)$ such that $w = sp$. Thus, we also have $s \in Pref(w) \cap Suff(w)$, in contradiction with Lemma 8.1. □

**Lemma 8.7.** *Let $w \in A^*$ be an unbordered word with $|w| \geq 2$. For all $p, p' \in A^+$ with $|p| < |w|$, $|p'| < |w|$, for all $h \in A^*$, $q \in \boldsymbol{N}$, $q \geq 2$, if $hwp'w \sim w^q p$ then $h = w^{q-2}$ and $p = p'$. In particular the result holds when $p, p' \in Pref_c(w)$.*

*Proof.* Suppose that, for $p, p' \in A^+$ with $|p| < |w|$, $|p'| < |w|$, $h \in A^*$, $q \in \boldsymbol{N}$, $q \geq 2$, we have $hwp'w \sim w^q p$. Two cases can occur:

(1) $hwp'w = w_2' w^g p w^{g'} w_1'$ with $w_1' w_2' = w$, $w_1', w_2' \in A^*$ and $g + g' = q - 1$;
(2) $hwp'w = p_2 w^q p_1$ with $p_1 p_2 = p$, $p_1, p_2 \in A^*$.

*Case (1).* In the first case, $w_1'$ is a prefix of $w$. Then, in virtue of Lemma 8.1, we have either $w_1' = 1$ or $w_1' = w$ and so we must have

$$hwp'w = w^{g+1} p w^{g'} \quad (*)$$

or

$$hwp' = w^g p w^{g'}. \quad (**)$$

We notice that we cannot have equality $(*)$ with $g' \neq 1$ (otherwise we would also have either $sp' = w$ or $sp = w$, with $s \in Suff(w)$, in contradiction with Lem. 8.6) nor can we have equality $(**)$ with $g' \neq 0$ (otherwise, once again, we would also have $sp' = w$ with $s \in Suff(w)$, in contradiction with Lem. 8.6). Thus, suppose that either $(*)$ holds with $g' = 1$ or $(**)$ holds with $g' = 0$. In both cases we have $hwp' = w^{q-1}p$. If $p = p'$ then $h = w^{q-2}$ and the proof is ended. Otherwise, either $p'$ is a proper suffix of $p$ or $p$ is a proper suffix of $p'$. As a result, we also have either $hw = w^{q-1}p''$ with $p'' \in Pref(p)$ or $hwp'' = w^{q-1}$ with $p'' \in Pref(p')$. In both cases we have $w = sp''$ with $0 < |p''| < |w|$, $s \in Suff(w)$, in contradiction with Lemma 8.6.

*Case (2).* Assume that $hwp'w = p_2 w^q p_1$ with $p_1 p_2 = p$, $p_1, p_2 \in A^*$. Then, we have either $p_1 \neq 1$ or $p_1 = 1$. Correspondingly, we also have either $w = sp_1$ or $w = sp'$ with $s \in Suff(w)$. In both cases we are in contradiction with Lemma 8.6. □

**Lemma 8.8.** *Let $w \in A^*$ be an unbordered word with $|w| > 2$, let $SC_{PA} = (A, I, R)$ be the circular splicing system defined by*

$$I = \{\sim(w^t Pref_c(w)) \mid 0 \leq t \leq 4\}, \quad R = \cup_{p_g \in Pref_c(w)} R_{p_g}$$

*where the $R_{p_g}$'s are defined as in Lemmata 8.3 and 8.4. Then, $C(SC_{PA}) \subseteq cyclic(w)$.*

*Proof.* Let us prove that $C(SC_{PA}) \subseteq cyclic(w)$ for the circular splicing system $SC_{PA} = (A, I, R)$ defined in the statement above. Let $\sim z \in C(SC_{PA})$, we will prove that $\sim z \in cyclic(w)$, by using induction over $|z|$. Since $I \subseteq cyclic(w)$, let us suppose that $\sim z$ is generated by using one of the rules in $R$. Then, $\sim z = \sim u_2 h u_1 u_4 k u_3$ with $u_1|u_2\$u_3|u_4 \in R$ and $\sim h u_1 u_2 \in C(SC_{PA})$, $\sim k u_3 u_4 \in C(SC_{PA})$. Since for every rule in $R$ we have $u_i \neq 1$, for $i \in \{1, 2, 3, 4\}$, then $|hu_1u_2| < |z|$, $|ku_3u_4| < |z|$ and so, by using induction hypothesis, $\sim hu_1u_2 \in cyclic(w)$, $\sim ku_3u_4 \in cyclic(w)$. Looking at the form of rules in $R$, there exist non-negative integers $q, j$ and $p, p', t, t' \in Pref_c(w)$ such that $hu_1u_2 = hwp'w \sim w^q p$, $ku_3u_4 = kwt'w \sim w^j t$. Consequently, $q \geq 2$, $j \geq 2$, and in virtue of Lemma 8.7, $h = w^{q-2}$, $p = p'$, $k = w^{j-2}$, $t = t'$. Finally, $z \sim u_2hu_1u_4ku_3$, and by looking at the form of the rules in $R$, we see that there exists $v \in Pref_c(w)$ such that for $\sim u_2hu_1u_4ku_3$ one of the following four cases occur: $u_2hu_1u_4ku_3 \sim w^q vww^j$ or $u_2hu_1u_4ku_3 \sim w^q ww^j v$ or $u_2hu_1u_4ku_3 \sim w^q wvw^j$ or $u_2hu_1u_4ku_3 \sim w^q vw^j$. Thus $\sim z = \sim u_2hu_1u_4ku_3 \in cyclic(w)$ and the conclusion holds. □

In virtue of Lemmata 8.5 and 8.8, we can state the result below.

**Proposition 8.2.** *For all unbordered words $w \in A^*$, with $|w| > 2$, $cyclic(w)$ is $C_{PA}$ generated.*

The next two propositions allow us to compare cyclic languages with the classes of languages introduced in Section 6. Indeed, Proposition 8.3 shows that the class of cyclic languages is different from the class of the languages which are circularizations of the Kleene closure of regular languages. Proposition 8.4 shows that the class of the regular linearizations of cyclic languages is different from the class of the cycle closed languages.

**Proposition 8.3.** *There exist unbordered words $w \in A^*$, with $|w| > 2$, such that $cyclic(w) \neq \sim X^*$, for every regular language $X \subseteq A^*$.*

*Proof.* Let $w = abc$. By contradiction, suppose that a regular language $X$ exists such that $cyclic(abc) = \sim X^*$. Since $\sim a \in cyclic(abc) = \sim X^*$, $\sim ab \in cyclic(abc) = \sim X^*$, we have $a$, $ab \in X^*$ or $a$, $ba \in X^*$. Consequently, $aab \in X^*$ or $aba \in X^*$ and so $\sim aab \in \sim X^* = cyclic(abc)$, which is a contradiction. □

**Remark 8.3.** Notice that we can easily prove that $cyclic(abc) \neq \sim(X^+ \cup Y)$, for every pair of finite subsets $X, Y$ of $A^*$ such that $X^+$ and $Y$ are closed under the conjugacy relation. Indeed, by contradiction, suppose that such a pair $X, Y$ exists with $cyclic(abc) = \sim(X^+ \cup Y)$. Thus, a positive integer $n$ also exists such that $\sim(abc)^n a \in cyclic(abc) \setminus \sim Y$, $\sim(abc)^n ab \in cyclic(abc) \setminus \sim Y$. Consequently, $\sim(abc)^n a$, $\sim(abc)^n ab \in \sim X^+$. Since $X^+$ is closed under the conjugacy relation, we have $(abc)^n a$, $(abc)^n ab \in X^+$ and so $(abc)^n a (abc)^n ab \in X^+$. Now, we have a contradiction since $\sim(abc)^n a (abc)^n ab \in \sim(X^+ \cup Y) \setminus cyclic(abc)$.

The following observation will be used in the proof of Proposition 8.4.

**Lemma 8.9.** *For all words $w, w' \in A^+$ such that $|w'| = g|w|$, $g$ being a positive integer, we have $^{\sim}w' \notin cyclic(w)$.*

*Proof.* By contradiction, suppose that for $w, w' \in A^+$ we have $|w'| = g|w|$ and $^{\sim}w' \in cyclic(w)$ with $g$ being a positive integer. Using Definition 8.1, there exists a nonnegative integer $q$, and $p \in Pref_c(w)$ also exists such that $^{\sim}w' = ^{\sim}w^q p$. Then, by using the above equalities we get $|w'| = g|w| = q|w| + t$, where $0 < t = |p| < |w|$ which is a contradiction. □

**Proposition 8.4.** *There exist unbordered words $w \in A^*$, with $|w| > 2$, such that, for each regular linearization $L$ of $cyclic(w)$, $L$ is not a cycle closed language. In particular the result holds for $L = Lin(cyclic(w))$.*

*Proof.* For the sake of simplicity, take $w = abc$ (the argument below works for all words $w$).

By contradiction, suppose that a regular linearization $L$ of $cyclic(abc)$ exists such that $L$ is a cycle closed language. Obviously we have $L \subseteq A^+$. Let $\mathcal{A}= (Q, \{a, b, c\}, \delta, q_0, F)$ be the minimal finite state automaton recognizing $L$ and let $v$ be a simple cycle in $\mathcal{A}$ ($v$ exists since $cyclic(abc)$ is not a finite set, so $L$ is not a finite set and we can apply Lem. 6.2). We now prove that $v \notin L$, so $L$ is not a cycle closed language (see Def. 6.2). Since $\mathcal{A}$ is trim, there exist $u, z \in A^*$ such that $uv^*z \in L$. In particular, $uz \in L$ which implies $uz \neq 1$ and $|uz| = g|abc| + t$, with $g, t \in \boldsymbol{N}$, $0 < t < |abc|$ (Lem. 8.9). On the other hand, $g', t' \in \boldsymbol{N}$ also exist such that $|v| = g'|abc| + t'$, $0 \leq t' < |abc|$. If $t' = 0$, in view of Lemma 8.9, we have $^{\sim}v \notin cyclic(abc) = ^{\sim}L$, so $v \notin L$ and we have ended the proof. Otherwise, $t, t' \in \{1, 2\}$ and, given that $uvz \in L$, using Lemma 8.9 once again, we have $t + t' \pmod 3 \in \{1, 2\}$, *i.e.*, $t = t' = 1$ or $t = t' = 2$. In both cases we have $uvvz \in L$ with $|uvvz| = 0 \pmod 3$, which contradicts Lemma 8.9. □

We have proven that $cyclic(w)$ is $C_{PA}$ generated under the hypotheses that $w$ is an unbordered word and $|w| > 2$. A natural question is whether the same result holds when we remove the above-mentioned hypotheses. We are not able to answer this question with respect to the first hypothesis. On the contrary, Example 8.3 shows that the hypothesis $|w| > 2$ in Proposition 8.2 is necessary.

**Example 8.3.** Let $w \in (\{a, b\})^2$. We can see that $cyclic(w) \notin Reg^{\sim} \cap C(Fin, Fin)$. Indeed, by contradiction, suppose that the contrary holds. We observe that every circular word $w'$ in $cyclic(w)$ has an odd length, *i.e.*, there exists a nonnegative integer $n$ such that $|w'| = 2n + 1$. Since $cyclic(w)$ is not a finite language, for a sufficiently large $n$ there exist two circular words $x, y \in cyclic(w)$, $|x| = 2k + 1, |y| = 2h + 1$, with $h, k \in \boldsymbol{N}$, such that $w'$ is obtained by using a rule and by starting from $x, y$. On the other hand, looking at the definition of circular splicing, we see that the length of $w'$ is the sum of the lengths of $x$ and $y$. However, this sum is an even number and this is a contradiction.

We end this section with Propositions 8.5 and 8.6. Proposition 8.5 suggests that we can modify $cyclic(w)$ and define a similar class of regular circular languages (*weak cyclic languages*) which are $C_{PA}$ generated. Below, we will only define the weak cyclic language $Wcyclic(abc)$. Observe that $Wcyclic(abc)$ differs from $cyclic(abc)$ only in the last subset $\sim(abc)^*ac$. Proposition 8.6 shows that $cyclic(w)$ is $C_{PI}$ generated, for each $w \in (\{a,b\})^2$. In the proof of Proposition 8.6, we use Lemma 8.10 as a preliminary observation.

**Proposition 8.5.** *The regular circular language*

$Wcyclic(abc) = \ {\sim}(abc)^*a \cup \ {\sim}(abc)^*b \cup \ {\sim}(abc)^*c \cup \ {\sim}(abc)^*ab \cup \ {\sim}(abc)^*bc \cup {\sim}(abc)^*ac$

*is* $C_{PA}$ *generated by the splicing system* $SC_{PA} = (A, I, R)$ *where we set* $w = abc$, $A = \{a, b, c\}$, $I = \{\sim w^i a, \sim w^i b, \sim w^i c, \sim w^i ab, \sim w^i bc, \sim w^i ac \mid 0 \leq i \leq 4\}$, $R = \{R_a, R_b, R_c, R_{ab}, R_{bc}, R_{ac}\}$ *and* $R_a = wab|w\$wa|cw$, $R_b = wab|w\$wb|cw$, $R_c = wa|cw\$w|bcw$, $R_{ab} = wa|w\$w|bw$, $R_{bc} = wb|w\$w|cw$, $R_{ac} = wa|w\$w|cw$.

*Proof.* Let us denote $Wcyclic(abc) = \ {\sim}(abc)^* Pref_{wc}$, where $Pref_{wc} = \{a, b, c, ab, bc, ac\}$. We prove that ${\sim}w^t Pref_{wc}(w) \subseteq C(SC_{PA})$ by using induction on $t$.

If $t \in \{0, 1, 2, 3, 4\}$ we have ${\sim}w^t Pref_{wc}(w) \subseteq I$ and so ${\sim}w^t Pref_{wc}(w) \subseteq C(SC_{PA})$. Let us suppose that ${\sim}w^j Pref_{wc}(w) \subseteq C(SC_{PA})$, for $0 \leq j < t$, $t \geq 5$, and let us prove that ${\sim}w^t Pref_{wc}(w) \subseteq C(SC_{PA})$. Indeed, for all $p_g \in Pref_{wc}(w)$ with $|p_g| = 1$ the circular word ${\sim}w^t p_g \in Wcyclic(w)$ is generated by means of the rule $R_{p_g}$, starting from ${\sim}w^{t_1-1}pw \in Wcyclic(w)$ and ${\sim}w^{t_2-1}sw \in Wcyclic(w)$, with $t_1 + t_2 = t - 1$, $t_1 \geq 2$, $t_2 \geq 2$ (in virtue of Lem. 8.3 for $R_b$, by using Paun's definition of circular splicing for $R_a$ and $R_c$, with $p = ab$, $s = ac$ for $R_a$ and $p = ac$, $s = bc$ for $R_c$). By using induction hypothesis over $t$, ${\sim}w^{t_1-1}pw \in C(SC_{PA})$ and ${\sim}w^{t_2-1}sw \in C(SC_{PA})$, consequently ${\sim}w^t p_g \in C(SC_{PA})$, by using the definition of circular splicing.

Furthermore, for all $p_g \in Pref_{wc}(w)$ with $|p_g| > 1$ the circular word ${\sim}w^t p_g \in Wcyclic(w)$ is generated by means of the rule $R_{p_g}$, starting from ${\sim}w^{t_1-1}pw \in Wcyclic(w)$ and ${\sim}w^{t_2-1}sw \in Wcyclic(w)$, with $t_1 + t_2 = t$, $ps = p_g$, $t_1 \geq 2$, $t_2 \geq 2$ (in virtue of Lem. 8.4 for $R_{ab}$ and $R_{bc}$, and by using Paun's definition of circular splicing for $R_{ac}$). By using induction hypothesis over $t$, ${\sim}w^{t_1-1}pw \in C(SC_{PA})$ and ${\sim}w^{t_2-1}sw \in C(SC_{PA})$, consequently ${\sim}w^t p_g \in C(SC_{PA})$, by using the definition of circular splicing.

Let us prove that $C(SC_{PA}) \subseteq Wcyclic(w)$ for the circular splicing system $SC_{PA} = (A, I, R)$ defined in the statement above. Let ${\sim}z \in C(SC_{PA})$. We will prove that ${\sim}z \in Wcyclic(w)$, by using induction over $|z|$. Since $I \subseteq Wcyclic(w)$, let us suppose that ${\sim}z$ is generated by using one of the rules in $R$. Then, ${\sim}z = {\sim}u_2hu_1u_4ku_3$ with $u_1|u_2\$u_3|u_4 \in R$ and ${\sim}hu_1u_2 \in C(SC_{PA})$, ${\sim}ku_3u_4 \in C(SC_{PA})$. Since for every rule in $R$ we have $u_i \neq 1$, for $i \in \{1, 2, 3, 4\}$, then $|hu_1u_2| < |z|$, $|ku_3u_4| < |z|$ and so, by using induction hypothesis, ${\sim}hu_1u_2 \in Wcyclic(w)$, ${\sim}ku_3u_4 \in Wcyclic(w)$. Looking at the form of rules in $R$, there exist nonnegative integers $q, j$ and $p, p', t, t' \in Pref_{wc}(w)$ such that $hu_1u_2 = hwp'w \sim w^q p$, $ku_3u_4 = kwt'w \sim w^j t$. Consequently, $q \geq 2$, $j \geq 2$, and in virtue of Lemma 8.7, $h = w^{q-2}$, $p = p'$, $k = w^{j-2}$, $t = t'$. Finally, $z \sim u_2hu_1u_4ku_3$ and, by looking at the form of the rules in $R$, we see that there exists $v \in Pref_{wc}(w)$ such that for ${\sim}u_2hu_1u_4ku_3$ one of the following three cases occur: $u_2hu_1u_4ku_3 \sim w^q v w^j$

or $u_2hu_1u_4ku_3 \sim w^q ww^j v$ or $u_2hu_1u_4ku_3 \sim vw^q ww^j$. Thus $^\sim z = ^\sim u_2hu_1u_4ku_3 \in Wcyclic(w)$ and the conclusion holds. □

**Lemma 8.10.** *We have* $Lin(cyclic(ab)) = \{(ab)^s a(ab)^t, b(ab)^s a(ab)^t a, (ab)^s b(ab)^t, b(ab)^s b(ab)^t a \mid s,t \geq 0\}$. *Furthermore,* $cyclic(ab) = cyclic(ba)$.

*Proof.* It is easy to observe that $cyclic(ab) = ^\sim(ab)^j a \cup ^\sim(ab)^j b$ and $Lin(cyclic(ab)) = \{(ab)^s a(ab)^t,\ b(ab)^s a(ab)^t a,\ (ab)^s b(ab)^t,\ b(ab)^s b(ab)^t a \mid s,t \geq 0\}$.

Furthermore, we notice that:

- $(ab)^s a(ab)^t \sim a(ba)^{s+t}$, $^\sim a(ba)^{s+t} \in cyclic(ba)$;
- $(ab)^s b(ab)^t = \begin{cases} a(ba)^{s-1}b(ba)^t b,\ ^\sim a(ba)^{s-1}b(ba)^t b \in cyclic(ba) & \text{for } s > 0, \\ (ba)^t b,\ ^\sim(ba)^t b \in cyclic(ba) & \text{for } s = 0; \end{cases}$
- $b(ab)^s a(ab)^t a = (ba)^{s+1}a(ba)^t$, $^\sim(ba)^{s+1}a(ba)^t \in cyclic(ba)$;
- $b(ab)^s b(ab)^t a = (ba)^s b(ba)^{t+1}$, $^\sim(ba)^s b(ba)^{t+1} \in cyclic(ba)$.

Consequently, $cyclic(ab) \subseteq cyclic(ba)$. Symmetrically, $cyclic(ba) \subseteq cyclic(ab)$ (it suffices to substitute the $a$'s with the $b$'s and *vice versa* in the above equalities). □

**Proposition 8.6.** *For each* $w \in (\{a,b\})^2$, $cyclic(w)$ *is* $C_{PI}$ *generated.*

*Proof.* We prove the statement for $w = ab$ (case 1) and for $w = aa$ (case 2). The other cases (when $w = ba$ or $w = bb$) can be obtained by substituting the $a$'s with the $b$'s and *vice versa* in cases 1 and 2.

*Case* 1. Suppose $w = ab$. We will prove that $cyclic(ab) = C(SC_{PI})$ by considering $SC_{PI} = (\{a,b\}, I, R)$ with $I = \{^\sim a, ^\sim b, ^\sim aba, ^\sim abb\}$ and $R = R_a \cup R_b$, with $R_a = \{r_a, \overline{r}_a\}$, $r_a = (aab, ab; abab)$, $\overline{r}_a = (ab, aab; 1)$; $R_b = \{r_b, \overline{r}_b\}$, $r_b = (abb, ab; abab)$, $\overline{r}_b = (ab, abb; 1)$. Firstly, let us show that $C(SC_{PI}) = \bigcup_{i \geq 0} \sigma^i(I) \subseteq cyclic(ab)$, by using induction on the the minimal $i$ such that $^\sim w' \in \sigma^i(I)$. Clearly $I \subseteq cyclic(ab)$. Since $\sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$ and by using induction hypothesis, we can suppose that there exist $^\sim w_1, ^\sim w_2 \in \sigma^{i-1}(I)$ such that $(^\sim w_1, ^\sim w_2) \vdash_{r_x, \overline{r}_x} {}^\sim w'$, for $r_x, \overline{r}_x \in R_x$, $x \in \{a,b\}$. By induction hypothesis, $^\sim w_1, ^\sim w_2 \in cyclic(ab)$. If $x = a$ (*i.e.*, we consider $r_a, \overline{r}_a$), then $^\sim w_1 = ^\sim \alpha\epsilon = {}^\sim aab\epsilon \in cyclic(ab)$ and $^\sim w_2 = ^\sim \alpha'\epsilon' = {}^\sim ab\epsilon' \in cyclic(ab)$. This means that $aab\epsilon$, $ab\epsilon' \in \{(ab)^s a(ab)^t, b(ab)^s a(ab)^t a, (ab)^s b(ab)^t, b(ab)^s b(ab)^t a \mid s,t \geq 0\}$ (Lem. 8.10). This implies $\epsilon \in (ab)^*$ and $ab\epsilon' \in \{(ab)^s a(ab)^t,\ (ab)^s b(ab)^t \mid s,t \geq 0\}$. Thus, $^\sim w' = ^\sim \epsilon abab\epsilon' \in cyclic(ab)$.

Otherwise, if $x = b$ (*i.e.*, we consider $r_b, \overline{r}_b$), then $^\sim w_1 = ^\sim \alpha\epsilon = {}^\sim abb\epsilon \in cyclic(ab)$, $^\sim w_2 = {}^\sim \alpha'\epsilon' = ^\sim ab\epsilon' \in cyclic(ab)$. We already know that $ab\epsilon' \in \{(ab)^s a(ab)^t,\ (ab)^s b(ab)^t \mid s,t \geq 0\}$ and, as above, $^\sim abb\epsilon \in cyclic(ab)$ implies $\epsilon \in (ab)^*$ (Lem. 8.10). Thus, $^\sim w' = ^\sim \epsilon abab\epsilon' \in cyclic(ab)$.

*Vice versa*, let us prove that $cyclic(ab) \subseteq C(SC_{PI})$, by induction on the length of $^\sim w' \in cyclic(ab)$. If $^\sim w' = ^\sim(ab)^i a$, with $i \in \{0,1\}$, then $^\sim w' \in I$ (similarly, if $^\sim w' = ^\sim(ab)^i b$, with $i \in \{0,1\}$, the same holds). Suppose that the statement holds for every $^\sim w' = ^\sim(ab)^i a$ and for every $^\sim w' = ^\sim(ab)^i b$, with $0 \leq i < n$, $n > 1$. Thus, $^\sim(ab)^{n-1}a \in C(SC_{PI})$ and we also have $^\sim aba \in I \subseteq C(SC_{PI})$. Consequently, starting from $^\sim(ab)^{n-1}a$ and $^\sim aba$, and by applying $r_a, \overline{r}_a$, we can generate $^\sim(ab)^{n-2}ababa = ^\sim(ab)^n a$. Analogously, starting from $^\sim(ab)^{n-1}b \in$

$C(SC_{PI})$ and $\sim abb \in I \subseteq C(SC_{PI})$, and by applying $r_b, \overline{r}_b$, we can generate $\sim(ab)^{n-2}ababb = \sim(ab)^n b$.

*Case* 2. Suppose $w = aa$. We now show that $cyclic(aa) = C(SC_{PI})$, for $SC_{PI} = (\{a\}, I, R)$ with $I = \{\sim a, \sim aaa\}$ and $R = \{r, \overline{r}\}$, $r = (aaa, aa; aaaa)$, $\overline{r} = (aa, aaa; 1)$. Notice that we have $cyclic(aa) = (a^2)^* a$.

Let us firstly show that $C(SC_{PI}) = \bigcup_{i \geq 0} \sigma^i(I) \subseteq cyclic(aa)$, by using induction on the minimal $i$ such that $\sim w' \in \sigma^i(I)$. Clearly $I \subseteq cyclic(aa)$. Since $\sigma^i(I) = \sigma^{i-1}(I) \cup \sigma'(\sigma^{i-1}(I))$ and by using induction hypothesis, we can suppose that there exist $\sim w_1, \sim w_2 \in \sigma^{i-1}(I)$ such that $(\sim w_1, \sim w_2) \vdash_{r, \overline{r}} \sim w'$, for $r, \overline{r} \in R$. Thus, $\sim w_1, \sim w_2 \in C(SC_{PI})$ and, by using induction hypothesis, $\sim w_1, \sim w_2 \in cyclic(aa)$. Then $\sim w_1 = \sim \alpha \epsilon = \sim aaa\epsilon \in cyclic(aa)$ and $\sim w_2 = \sim \alpha' \epsilon' = \sim aa\epsilon' \in cyclic(aa)$. Thus, we have $\epsilon \in (aa)^*$, $\epsilon' \in (aa)^* a$ and so $\sim w' = \sim \epsilon aaaa\epsilon' \in cyclic(aa)$.

*Vice versa*, let us prove that $cyclic(aa) \subseteq C(SC_{PI})$ by using induction on the length of $\sim w' \in cyclic(aa)$. If $\sim w' = \sim(aa)^i a$, with $i \in \{0, 1\}$, then $w \in I$. Suppose that the statement holds for every $w' \in \sim(aa)^i a$, with $0 \leq i < n$, $n > 1$. Thus, $\sim (aa)^{n-1} a \in C(SC_{PI})$ and we also have $\sim aaa \in I \subseteq C(SC_{PI})$. Consequently, starting from $\sim(aa)^{n-1} a$ and $\sim aaa$, and by applying $r, \overline{r}$, we can generate $\sim(aa)^n a$. $\square$

The results proved in this paper lead us to the conclusion that it is not easy to characterize the regular $C_{PA}$ generated languages. These results also suggest further investigation of the analogous problem of finding those regular circular languages which are generated by Pixton circular splicing systems.

*Acknowledgements*. Many thanks to J. Berstel and A. Restivo for suggesting references [3, 27] during the conference Words99 [6]. The authors also wish to thank A. Bertoni for useful discussions. Our attention was drawn to the case of a one-letter alphabet by a question asked by A. Restivo during a meeting in Milan and the authors thank him for this useful question. Last but not least, many thanks to the anonymous referees and to one of them in particular for his/her helpful and constructive criticism which allowed us to improve one the main results of this paper.

## References

[1] L.M. Adleman, Molecular computation of solutions to combinatorial problems. *Science* **226** (1994) 1021–1024.

[2] J. Berstel and D. Perrin, *Theory of codes.* Academic Press, New York (1995).

[3] J. Berstel and A. Restivo, Codes et sousmonoides fermes par conjugaison. *Sem. LITP* **81-45** (1981) 10.

[4] P. Bonizzoni, C. De Felice and R. Zizza, *The structure of reflexive regular splicing languages via Schützenberger constants* (submitted).

[5] P. Bonizzoni, C. De Felice, G. Mauri and R. Zizza, DNA and circular splicing, in *Proc. of DNA* 2000, edited by A. Condon and G. Rozenberg. Springer, *Lect. Notes Comput. Sci.* **2054** (2001) 117–129.

[6] P. Bonizzoni, C. De Felice, G. Mauri and R. Zizza, Linear and circular splicing, in *WORDS99* (1999).

[7] P. Bonizzoni, C. De Felice, G. Mauri and R. Zizza, Decision Problems on Linear and Circular Splicing, in *Proc. of DLT* 2002, edited by M. Ito and M. Toyama. Springer, *Lect. Notes Comput. Sci.* **2450** (2003) 78–92.
[8] P. Bonizzoni, C. De Felice, G. Mauri and R. Zizza, Regular Languages Generated by Reflexive Finite Linear Splicing Systems, in *Proc. of DLT* 2003, edited by Z. Esik and Z. Fulop. Springer, *Lect. Notes Comput. Sci.* **2710** (2003) 134–145.
[9] P. Bonizzoni, C. De Felice, G. Mauri and R. Zizza, *Linear splicing and syntactic monoid* (submitted).
[10] P. Bonizzoni, C. De Felice, G. Mauri and R. Zizza, *On the power of circular splicing* (submitted).
[11] P. Bonizzoni, C. Ferretti, G. Mauri and R. Zizza, Separating some splicing models. *Inform. Process. Lett.* **79** (2001) 255–259.
[12] C. Choffrut and J. Karhumaki, Combinatorics on Words, in *Handbook of Formal Languages*, edited by G. Rozenberg and A. Salomaa. Springer, Vol. 1 (1996) 329–438.
[13] K. Culik and T. Harju, Splicing semigroups of dominoes and DNA. *Discrete Appl. Math.* **31** (1991) 261–277.
[14] L. Fuchs, *Abelian groups.* Pergamon Press, Oxford, London, New York and Paris (1960).
[15] R.W. Gatterdam, Splicing systems and regularity. *Intern. J. Comput. Math.* **31** (1989) 63–67.
[16] R.W. Gatterdam, Algorithms for splicing systems. *SIAM J. Comput.* **21** (1992) 507–520.
[17] T. Head, Formal Language Theory and DNA: an analysis of the generative capacity of specific recombinant behaviours. *Bull. Math. Biol.* **49** (1987) 737–759.
[18] T. Head, Circular suggestions for DNA Computing, in *Pattern Formation in Biology, Vision and Dynamics*, edited by A. Carbone, M. Gromov and P. Pruzinkiewicz. World Scientific, Singapore and London (2000) 325–335.
[19] T. Head, Gh. Paun and D. Pixton, Language theory and molecular genetics: generative mechanisms suggested by DNA recombination, in *Handbook of Formal Languages*, edited by G. Rozenberg and A. Salomaa. Springer, Vol. 2 (1996) 295–360.
[20] T. Head, G. Rozenberg, R. Bladergroen, C. Breek, P. Lommerse and H. Spaink, Computing with DNA by operating on plasmids. *BioSystems* **57** (2000) 87–93.
[21] J.E. Hopcroft, R. Motwani and J.D. Ullman, *Introduction to Automata Theory, Languages, and Computation.* Addison-Wesley, Reading, Mass. (2001).
[22] S.M. Kim, Computational modeling for genetic splicing systems. *SIAM J. Comput.* **26** (1997) 1284–1309.
[23] M. Lothaire, *Combinatorics on Words*, Encyclopedia of Math. and its Appl. Addison Wesley Publishing Company (1983).
[24] G. Paun, On the splicing operation. *Discrete Appl. Math.* **70** (1996) 57–79.
[25] G. Paun, G. Rozenberg and A. Salomaa, *DNA computing, New Computing Paradigms.* Springer-Verlag (1998).
[26] D. Pixton, Regularity of splicing languages. *Discrete Appl. Math.* **69** (1996) 101–124.
[27] C. Reis and G. Thierren, Reflective star languages and codes. *Inform. Control* **42** (1979) 1–9.
[28] R. Siromoney, K.G. Subramanian and A. Dare, Circular DNA and Splicing Systems, in *Proc. of ICPIA*. Springer, *Lect. Notes Comput. Sci.* **654** (1992) 260–273.

Communicated by A. Restivo.
Received July 25, 2002. Accepted February 19, 2004.