

On bonded indian and uniformly parallel insertion systems and their generative power

 Ahmad Firdaus Yosman^{a,*}, Markus Holzer^b, Bianca Truthe^b, Wan Heng Fong^a, Sherzod Turaev^c
^a Department of Mathematical Sciences, Faculty of Science, Universiti Teknologi Malaysia, 81310 UTM Johor Bahru, Johor, Malaysia

^b Institut für Informatik, Universität Giessen, Arndtstr. 2, 35392 Giessen, Germany

^c Department of Computer Science, Kulliyah of Information & Communication Technology, International Islamic University Malaysia, 53100 Jalan Gombak, Selangor, Malaysia

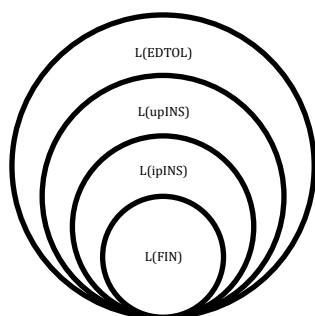
* Corresponding author: firdausyosman@yahoo.com

Article history

Received 21 September 2017

Accepted 22 October 2017

Graphical abstract



Abstract

Insertion is an operation in formal language theory that generalizes the operation of concatenation of words, where its variants allow the operation in different ways. Parallel insertion is a variant of insertion that simultaneously adds words between all letters of a word and also at the right and left extremities. In previous research, restrictions on the applicability have been imposed leading to so-called bonded insertion systems with a sequential and a parallel variant. Motivated by the atomic behavior of chemical compounds in the process of chemical bonding, the generative power of bonded insertion systems has been investigated where a language hierarchy was obtained. In this paper, we introduce new variants of bonded parallel insertion systems, namely bonded Indian parallel insertion systems and bonded uniformly parallel insertion systems. We present some results regarding the generative power of these new systems and a language hierarchy.

Keywords: Bonded parallel insertion systems, bonded Indian parallel insertion systems, bonded uniformly parallel insertion systems, formal languages, generative power

© 2017 Penerbit UTM Press. All rights reserved

INTRODUCTION

A formal language is a set of finite strings of symbols from a finite alphabet. The language theoretical operations which contribute to the forming of a sentence consist of concatenation, quotient and Kleene closure [1].

In [1], Kari introduced a generalization of the concatenation and quotient operations, namely the insertion and deletion operations, respectively. The study of insertion and deletion operations has been conducted extensively ever since its introduction as seen in [2-12].

Concatenating a word v to a word u yields one word, namely uv ; whereas inserting a word v into a word u may take place at an arbitrary position in the word u , resulting in a finite set of words. The operation of inserting one word into another word at a time is called *sequential insertion*. Inserting words into all possible positions at a time, which includes between all the letters in a word and to its right and left extremities, is called *parallel insertion*.

The atomic behaviour of chemical compounds in the process of chemical bonding motivated the work in [13], where the concepts of bonded sequential insertion systems (bSINS-systems) and bonded parallel insertion systems (bPINS-systems) were introduced. These bonded insertion systems describe operations of insertion on bonded alphabets, which is explained in detail in Section 2.

The results from [13] show that the relation

$$L(\text{REG}) \subset L(\text{bSINS}) \subset L(\text{bPINS}) \subset L(\text{EOL})$$

holds. In this case, $L(\text{REG})$, $L(\text{bSINS})$, $L(\text{bPINS})$, and $L(\text{EOL})$ refer to the families of regular languages, of languages generated by

bSINS-systems, of languages generated by bPINS-systems, and of languages generated by extended interactionless Lindenmayer systems, respectively (see, e.g., [13,14]).

In this paper, we explore the bPINS-systems further. By implementing some restrictions on the insertion rules, we introduce the concept of bonded Indian parallel insertion systems and bonded uniformly parallel insertion systems.

PRELIMINARIES

We assume that the reader is familiar with the basic concepts of formal language theory (see, e.g., [15]). Here, we only recall some notations used in the paper.

The cardinality of a set S is denoted by $|S|$; the inclusion of a set A in a set B is denoted by $A \subseteq B$ and the proper inclusion by $A \subset B$.

A set of symbols is called an *alphabet*, denoted by Σ . The set Σ^* of strings is obtained from the operations on symbols from Σ . A *language* L over an alphabet Σ is a subset of Σ^* .

We denote the empty word by λ . For a word w , the length is denoted by $|w|$.

We recall here the definition of a bonding alphabet used in [13]. Let \mathbb{Z} be the set of integers, as well as

$$\mathbb{Z}_0^- = \{0, -1, -2, \dots\} \quad \text{and} \quad \mathbb{Z}_0^+ = \{0, 1, 2, \dots\}.$$

Observe that $\mathbb{Z}_0^+ = \mathbb{N}$.

Let Σ be an alphabet. Then the set $B_\Sigma = \mathbb{Z}_0^+ \times \Sigma \times \mathbb{Z}_0^-$ is a *bonding alphabet* over Σ . An element $(i, a, -j)$ of B_Σ is called a *letter* a with left bond i and right bond $-j$. To simplify the presentation, we write $\left[\begin{smallmatrix} i \\ a \\ -j \end{smallmatrix} \right]$ instead of $(i, a, -j)$ for a letter from B_Σ . Let

$$w = \left[\begin{smallmatrix} i_0 \\ a_1 \\ -i_1 \end{smallmatrix} \right] \left[\begin{smallmatrix} i_1 \\ a_2 \\ -i_2 \end{smallmatrix} \right] \left[\begin{smallmatrix} i_2 \\ a_3 \\ -i_3 \end{smallmatrix} \right] \cdots \left[\begin{smallmatrix} i_{n-2} \\ a_n \\ -i_{n-1} \end{smallmatrix} \right]$$

be a non-empty sequence of letters from B_Σ . The sequence w is called a *bond word* and is said to be *well-formed* if all bonds fit, i.e., $i_{2j-1} + i_{2j} = 0$, for $1 \leq j \leq n-1$. If additionally, $i_0 + i_{2n-1} = 0$ holds, then w is said to be a *balanced word* or, for short, a *word*. In case $i_0 + i_{2n-1} \neq 0$, then the word is said to be *unbalanced*. Moreover, a word is *neutral* if $i_0 = i_{2n-1} = 0$. For a well-formed word

$$w = \left[\begin{smallmatrix} i_0 \\ a_1 \\ -i_1 \end{smallmatrix} \right] \left[\begin{smallmatrix} i_1 \\ a_2 \\ -i_2 \end{smallmatrix} \right] \left[\begin{smallmatrix} i_2 \\ a_3 \\ -i_3 \end{smallmatrix} \right] \cdots \left[\begin{smallmatrix} i_{n-1} \\ a_n \\ -i_n \end{smallmatrix} \right],$$

we say that the word w has the left bond i_0 and the right bond $-i_n$ as the *outer bonds* and i_1, \dots, i_{n-1} as *inner bonds*. If we are not interested in the inner bonds, we shortly write w as

$$\left[\begin{smallmatrix} i_0 \\ a_1 a_2 a_3 \cdots a_n \\ -i_n \end{smallmatrix} \right].$$

The set of all well-formed words built from letters of B_Σ including the empty word is referred to as B_Σ^* and the set of all balanced words built from letters of B_Σ including the empty word is referred to as B_Σ^\otimes . By definition $B_\Sigma^\otimes \subset B_\Sigma^*$. The empty word is the neutral element of both structures B_Σ^* and B_Σ^\otimes . For the empty word, we write $\left[\begin{smallmatrix} i_0 \\ \lambda \\ -i_0 \end{smallmatrix} \right]$ for some number $i_0 \in \mathbb{Z}_0^+$. The empty word is always a balanced word.

The length of a bond word w from B_Σ^* or B_Σ^\otimes is denoted by $|w|$ and is equal to the number of letters in w . In particular, the empty bond word $\left[\begin{smallmatrix} i \\ \lambda \\ -i \end{smallmatrix} \right]$ is of length 0.

According to [13], let Σ be a finite alphabet, $A \subseteq B_\Sigma^\otimes$ be a finite set of *axioms* which contains only neutral words, and $I \subseteq B_\Sigma^\otimes$ be a finite set of *insertion strings*. A *bonded parallel insertion system* (bPINS-system) is a triple $\gamma = (\Sigma, A, I)$, where the derivation relation \Rightarrow_γ is defined as follows: let $\alpha, \beta \in B_\Sigma^\otimes$. Then $\alpha \Rightarrow_\gamma \beta$ if and only if

$$\alpha = \alpha_1 \alpha_2 \cdots \alpha_n$$

for letters $\alpha_i \in B_\Sigma$ with $1 \leq i \leq n$ and there are insertion strings $\alpha'_i \in I$ for $1 \leq i \leq n+1$ such that

$$\beta = \alpha'_1 \alpha_1 \alpha'_2 \alpha_2 \cdots \alpha'_{n-1} \alpha_{n-1} \alpha'_n \alpha_n \alpha'_{n+1}.$$

Since $\beta \in B_\Sigma^\otimes$, the insertion of the strings α'_i is balance preserving.

The reflexive and transitive closure of \Rightarrow_γ is denoted by \Rightarrow_γ^* . If there is no risk of ambiguity, we write \Rightarrow and \Rightarrow^* instead of \Rightarrow_γ and \Rightarrow_γ^* , respectively.

A homomorphism

$$h_{be} : B_\Sigma^\otimes \rightarrow \Sigma^*$$

is defined by

$$h_{be} \left(\left[\begin{smallmatrix} i \\ a \\ -j \end{smallmatrix} \right] \right) = a$$

for every $\left[\begin{smallmatrix} i \\ a \\ -j \end{smallmatrix} \right] \in B_\Sigma$ and is called the *bond erasing homomorphism*. The language generated by a bPINS-system $\gamma = (\Sigma, A, I)$ is defined as

$$L(\gamma) = \{ h_{be}(\beta) \mid \text{there is an axiom } \alpha \in A \text{ such that } \alpha \Rightarrow_\gamma^* \beta \}.$$

The family of all such languages is denoted by $L(\text{bSINS})$.

We now give the definition of an extended deterministically tabled interactionless Lindenmayer system (EDTOL-system) because there are some similarities in the derivation process of such Lindenmayer systems and bonded uniformly parallel insertion systems. For further information on Lindenmayer systems, we refer to [14].

An EDTOL-system is a 4-tuple $\Gamma = (\Sigma, H, \omega, \Delta)$, where Σ is a finite alphabet, H is a finite set of homomorphisms on the set Σ^* (which are called *tables*), $\omega \in \Sigma^*$ is called the *axiom*, and $\Delta \subseteq \Sigma$. Instead of $w \in h(a)$ for $h \in H$ and $a \in \Sigma$, we write $a \rightarrow w$. A word $u \in \Sigma^*$ is derived to a word $v \in \Sigma^*$, written as $u \Rightarrow_\Gamma v$, if and only if there is a homomorphism $h \in H$ such that $v = h(u)$.

The reflexive and transitive closure of \Rightarrow_Γ is denoted by \Rightarrow_Γ^* . If there is no risk of ambiguity, we write \Rightarrow and \Rightarrow^* instead of \Rightarrow_Γ and \Rightarrow_Γ^* , respectively.

The language generated by Γ is defined as

$$L(\Gamma) = \{ x \in \Delta^* \mid \omega \Rightarrow_\Gamma^* x \}$$

The family of all languages generated by an EDTOL-system is denoted by $L(\text{EDTOL})$.

The methodology of this paper consists of utilizing all of the aforementioned definitions to produce and prove new definitions and theorems as shown in the next section.

RESULTS AND DISCUSSION

In a derivation step of a bonded parallel insertion system, at every position in a word, an insertion string is inserted if the system contains a suitable insertion string for the respective bonds. For different positions with the same bonds, different insertion strings may be chosen. We introduce here two variants of such a system in which the insertion is more restrictive.

In one other variant, the so-called *bonded Indian parallel insertion systems*, only one bond and a suitable insertion string is chosen for insertion in a derivation step. At all positions of other bonds, no insertion takes place in this derivation step. The name is chosen due to the similarity to Indian parallel grammars [14], where every occurrence of a non-terminal in a sentential form is replaced according to the same rule in a single derivation step.

In another variant, the so-called *bonded uniformly parallel insertion systems*, only one insertion string is chosen for each bond in a single derivation step and these insertion strings are inserted everywhere where they fit. In another derivation step, a different string can be chosen for inserting at every position of the same bond. In this way, it is not possible to insert different strings at different positions of the same bond in the same derivation step.

We now formally define the two variants. The systems themselves do not differ; the difference is in the derivation process.

Let Σ be a finite alphabet, $A \subseteq B_{\Sigma}^{\otimes}$ be a finite set of *axioms* that contains only neutral words, and $I \subseteq B_{\Sigma}^{\otimes}$ be a finite set of *insertion strings*. A *bonded Indian parallel insertion system* (ipINS-system) and a *bonded uniformly parallel insertion system* (upINS-system) are triples $\gamma = (\Sigma, A, I)$, where the derivation relation \Rightarrow_{γ} of an ipINS-system $\gamma = (\Sigma, A, I)$ is defined as follows: let $\alpha \in A$ and $\beta \in B_{\Sigma}^{\otimes}$. Then we write $\alpha \Rightarrow_{\gamma} \beta$ if and only if there is a number $n \geq 1$, an insertion word $\delta \in I$ with a left bond i_{δ} , and a set S of non-empty bond words $\left[i_0 a_{1-i_{\delta}} \right], \left[i_{\delta} a_{2-i_{\delta}} \right], \dots, \left[i_{\delta} \alpha_{n-1-i_{\delta}} \right], \left[i_{\delta} \alpha_{n-i_n} \right]$ of the set B_{Σ}^* such that

- $\alpha = \left[i_0 a_{1-i_{\delta}} \right] \left[i_{\delta} a_{2-i_{\delta}} \right] \dots \left[i_{\delta} \alpha_{n-1-i_{\delta}} \right] \left[i_{\delta} \alpha_{n-i_n} \right]$,
- $\beta = \delta' \left[i_0 a_{1-i_{\delta}} \right] \delta \left[i_{\delta} a_{2-i_{\delta}} \right] \delta \dots \delta \left[i_{\delta} \alpha_{n-1-i_{\delta}} \right] \delta \left[i_{\delta} \alpha_{n-i_n} \right] \delta''$,

where δ' is equal to δ if $i_0 = i_{\delta}$ and $\left[i_0 \lambda_{-i_0} \right]$ otherwise; as well as δ'' is equal to δ if $i_n = i_{\delta}$ and $\left[i_n \lambda_{-i_n} \right]$ otherwise, and

- the string δ cannot be inserted somewhere else in the word α (every bond word from the set S does not have i_{δ} as an inner bond).

Meanwhile, the derivation relation \Rightarrow_{γ} of a upINS-system $\gamma = (\Sigma, A, I)$ is defined as follows: let $\alpha \in A$ and $\beta \in B_{\Sigma}^{\otimes}$. Then we write $\alpha \Rightarrow_{\gamma} \beta$ if and only if $\alpha = \alpha_1 \alpha_2 \dots \alpha_n$ for a number $n \geq 1$ and non-empty subwords $\alpha_i \in B_{\Sigma}^*$ with $1 \leq i \leq n$ and there are balanced words $\delta_i \in B_{\Sigma}^{\otimes}$ for $0 \leq i \leq n$ such that

- $\beta = \delta_0 \alpha_1 \delta_1 \alpha_2 \dots \delta_{n-2} \alpha_{n-1} \delta_{n-1} \alpha_n \delta_n$ and $\delta_i \in I$ for $1 \leq i \leq n-1$. $\delta_0 \in I$ if there is an insertion word in I which fits to the left bond of α_1 . Otherwise, $\delta_0 = \left[i_0 \lambda_{-i_0} \right]$. $\delta_n \in I$ if there is an insertion word in I which fits to the right bond of α_n . Otherwise, $\delta_n = \left[i_n \lambda_{-i_n} \right]$,
- whenever $\delta_j \alpha_i \dots \alpha_{j-1} \delta_i$ ($1 \leq i < j \leq n+1$) is also well-formed, then $\delta_i = \delta_j$, and
- there is no insertion string which can be inserted inside some word α_i with $1 \leq i \leq n$ (for any decomposition $\alpha_i = u_i v_i$ with $1 \leq i \leq n$ and insertion string $\zeta \in I$, the bond word $u_i \zeta v_i$ is not well-formed).

The reflexive and transitive closure of \Rightarrow_{γ} for both variants is denoted by \Rightarrow_{γ}^* . If there is no risk of ambiguity, we write \Rightarrow and \Rightarrow^* instead of \Rightarrow_{γ} and \Rightarrow_{γ}^* , respectively.

The language generated by an ipINS-system or a upINS-system $\gamma = (\Sigma, A, I)$ is defined as

$$L(\gamma) = \{ h_{bc}(\beta) \mid \text{there is an axiom } \alpha \in A \text{ such that } \alpha \Rightarrow_{\gamma}^* \beta \}.$$

The family of all languages generated by an ipINS-system is denoted by $L(\text{ipINS})$; the family of all languages generated by a upINS-system is denoted by $L(\text{upINS})$.

In order to clarify our notation, we give examples for the new variants.

Example 1

Let $\gamma_1 = (\{a\}, A, I)$ be an ipINS-system with $A = \{ \left[i_0 a_0 \right], \left[i_0 a_{-1} \right] \left[i_1 a_0 \right] \}$ and $I = \{ \left[i_1 a_{-2} \right] \left[i_2 a_{-1} \right] \}$.

Since the system contains only one insertion string, this system generates the same language no matter whether it is considered as an ipINS-system (where only one insertion string is inserted at every possible position in a derivation step), a upINS-system (where, for every different bond, only one insertion string is inserted at every possible position in a derivation step), or a bPINS-system (where at every position some suitable insertion string is inserted in a derivation step). As shown in [13], the language $\{ a^{2^n} \mid n \geq 0 \}$ is generated by the system γ_1 .

Example 2

Let $\gamma_2 = (\{a, b\}, A, I)$ be a upINS-system with $A = \{ \left[i_0 a_{-3} \right] \left[i_3 b_0 \right], \left[i_0 a_{-1} \right] \left[i_1 a_{-3} \right] \left[i_3 b_{-2} \right] \left[i_2 b_0 \right] \}$ and $I = \{ \left[i_1 a_{-3} \right] \left[i_3 a_{-1} \right], \left[i_2 b_{-3} \right] \left[i_3 b_{-2} \right] \}$.

The words of the set A yield the words ab and $aabb$. The word $\left[i_0 a_{-1} \right] \left[i_1 a_{-3} \right] \left[i_3 b_{-2} \right] \left[i_2 b_0 \right]$ has two positions where insertions can be applied; at every such position, only one insertion string fits. After inserting, the word $aaaabbbb$ is obtained which has four insertion positions, two of each bond for which an insertion string exists. Inside the insertion strings, no insertion is possible. Hence, in each derivation step, from a word $a^{2^n} b^{2^n}$ (which has 2^{n-1} possible insertion positions for every insertion string) for some natural number $n \geq 1$, the word $a^{2^{n+1}} b^{2^{n+1}}$ is obtained which has 2^n possible insertion positions for every insertion string. Other words are not generated. Thus, the language generated is $\{ a^{2^n} b^{2^n} \mid n \geq 0 \}$.

If the system γ_2 is considered as an ipINS-system, then the word $\left[i_0 a_{-1} \right] \left[i_1 a_{-3} \right] \left[i_3 b_{-2} \right] \left[i_2 b_0 \right]$ would yield either the word $aaaabb$ or $aabbbb$ in one derivation step. This is because either the first insertion string is inserted at every possible position or the second one but not both at the same time. Hence, as an ipINS-system, the system γ_2 generates another language than as a upINS-system.

In the sequel, we will prove a hierarchy of language families relating the families of the languages generated by ipINS-system and upINS-system to the family of finite languages and the family of the languages generated by extended deterministically tabled interactionless Lindenmayer systems (EDTOL-systems).

Theorem 1

The proper inclusion $L(\text{FIN}) \subset L(\text{ipINS})$ holds.

Proof.

Any finite language can be generated by an ipINS-system by taking all its words as axioms with bonds zero and giving no insertions words. A witness for the properness is the infinite language $\{a^{2^n} \mid n \geq 0\}$ which is generated by γ_1 from Example 1. □

Bonded Indian parallel insertion systems form a specialization of bonded uniformly parallel insertion systems, as we will show in the next theorem.

Theorem 2

The proper inclusion $L(\text{ipINS}) \subset L(\text{upINS})$ holds.

Proof.

An ipINS-system γ can be simulated by a upINS-system Γ which is obtained by extending γ with empty insertion strings. Then any derivation in γ can be simulated in Γ by using the same insertion string and empty insertion strings for all other bonds. Every derivation in Γ can be simulated in γ by sequentially inserting the insertion strings for different bonds.

A witness for the properness is the language $L = \{a^{2^n} b^{2^n} \mid n \geq 0\}$

which is generated by the upINS-system γ_2 from Example 2 but not by an ipINS-system since such a system would increase the number of occurrences of either a or b but not both in one step. □

As the last result in our paper, we show the following relation to EDTOL-systems.

Theorem 3

The proper inclusion $L(\text{upINS}) \subset L(\text{EDTOL})$ holds.

Proof.

The idea for the simulation of a upINS-system $\gamma = (\Sigma, A, I)$ by an EDTOL-system is as follows. We represent a well-formed word of the set B_Σ^* by a sequence of symbols, where each position where an insertion word can be inserted is represented by a non-terminal symbol, and each letter of Σ is represented by itself and is considered as a terminal symbol. The tables of the constructed EDTOL-system ensure that at every insertion position of the same bond, the same insertion string is inserted (the non-terminal representing the insertion position is replaced by a word representing the insertion string). Also, there is a table which eliminates all the non-terminal symbols at once (which simulates the bond erasing homomorphism h_{be}).

Let

$$B = \{ \ell \mid \text{there are } [{}_\ell a_r] \in B_\Sigma \text{ and } u, v \in B_\Sigma^*, \text{ with } u[{}_\ell a_r]v \in A \cup I \}.$$

We now separate the insertion strings according to the positions where they can be inserted: for each $b \in B$, let I_b be the set of all insertion strings which have b as their left bonds:

$$I_b = \{ [{}_b w_{-b}] \mid [{}_b w_{-b}] \in I \}$$

Only those bonds, for which an insertion string exists, need to be considered: $B_I = \{ b \mid I_b \neq \emptyset \}$. These and only these bonds are called *expandable*.

Let $b_1, b_2, \dots, b_{|B_I|}$ be the elements of the set B_I with $b_1 < b_2 < \dots < b_{|B_I|}$. Furthermore, let

$$T = I_{b_1} \times I_{b_2} \times \dots \times I_{b_{|B_I|}}.$$

We now construct an EDTOL-system $\Gamma = (N \cup \Sigma, H, S, \Sigma)$ which simulates the derivation of the upINS-system γ . The set N of new symbols is defined as

$$N = \{ S \} \cup \{ X_b \mid b \in B_I \}.$$

Let $f : B_\Sigma^* \rightarrow (N \cup \Sigma)^*$

be a mapping which transforms a bond word into its representation for the EDTOL-system: every expandable bond $b \in B_I$ is represented by X_b , the other bonds are omitted, and the basic letters are kept as they are.

We now define the tables of the EDTOL-system.

For every axiom $w \in A$, we create a start table

$$h_{S,w} = \{ S \rightarrow f(w) \} \cup \{ X_b \rightarrow X_b \mid b \in B_I \} \cup \{ a \rightarrow a \mid a \in \Sigma \}.$$

For every tuple $(w_{b_1}, w_{b_2}, \dots, w_{b_{|B_I|}}) \in T$, we create a table

$$h_{(w_{b_1}, \dots, w_{b_{|B_I|}})} = \{ S \rightarrow S \} \cup \{ X_{b_i} \rightarrow f(w_{b_i}) \mid 1 \leq i \leq |B_I| \} \cup \{ a \rightarrow a \mid a \in \Sigma \}.$$

For eliminating the non-terminal symbols, we create the table

$$h_\lambda = \{ S \rightarrow S \} \cup \{ X_b \rightarrow \lambda \mid b \in B_I \} \cup \{ a \rightarrow a \mid a \in \Sigma \}.$$

Finally, we set

$$H = \{ h_{S,w} \mid w \in A \} \cup \left\{ h_{(w_{b_1}, w_{b_2}, \dots, w_{b_{|B_I|}})} \mid (w_{b_1}, w_{b_2}, \dots, w_{b_{|B_I|}}) \in T \right\} \cup \{ h_\lambda \}.$$

This completes the description of the EDTOL-system Γ .

From the construction, it is not hard to see that any uniformly parallel insertion of words from I into the current bond word can be simulated by an application of an appropriate table of rules described above at the correct corresponding positions within the sentential form derived by the EDTOL-system. The details of the proof showing that $L(\Gamma) = L(\gamma)$ are left to the reader. Thus, $L(\text{upINS}) \subseteq L(\text{EDTOL})$.

A witness for the properness is the language $L = \{ a^{2^n} b^{n-1} c \mid n \geq 1 \}$, which is generated by the deterministically interactionless Lindenmayer system (DOL-system) G , defined as

$$G = \{ \{ a, b, c \}, \{ a \rightarrow aa, b \rightarrow bc, c \rightarrow bc \}, aac \}.$$

In [13], it was shown that the language L cannot be generated by a bPINS-system, where the argumentation did not use the possibility of a bPINS-system to use different insertion strings for different positions of the same bond. Hence, the same argumentation shows that also no upINS-system can generate the language L . □

CONCLUSION

In this paper, two new variants of bonded parallel insertion systems have been introduced, namely bonded Indian parallel insertion systems and bonded uniformly parallel insertion systems. The hierarchy of language families relating the families of the languages generated by ipINS-system and upINS-system to the family of finite languages and the family of the languages generated by extended deterministically tabled interactionless Lindenmayer systems (EDT0L-systems) is illustrated in Fig. 1. An arrow from an entry X to an entry Y represents the proper inclusion $X \subset Y$. The labels on the edges refer to the theorem where the respective inclusion is proven.

The results are not only of interest for research in theoretical computer science, but also have practical application in the advancement of research in biochemistry and DNA computing due to the more accurate modeling of DNA recombination by considering real-world atomic behavior of chemical bonding.

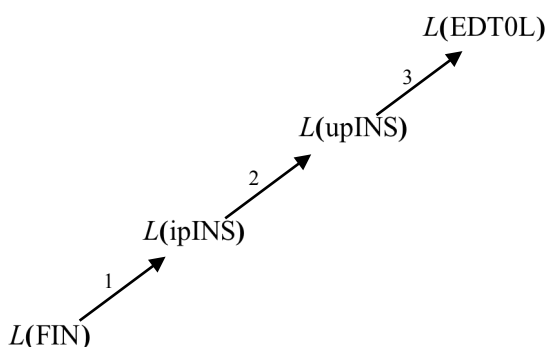


Fig. 1 Hierarchy of language families.

ACKNOWLEDGEMENT

The first author would like to thank the Ministry of Higher Education Malaysia (MOHE) for his MyBrainSC scholarship. The fourth author would like to acknowledge MOHE and Research Management Centre (RMC) of Universiti Teknologi Malaysia (UTM) for the financial funding through Research University Grant Vote No. 13H18. The fifth author would like to thank the International Islamic University Malaysia for his financial funding of Research Initiative Grant Scheme RIGS16-368-0532. The results presented in this paper were obtained during the first author's research attachment at the University of Giessen.

REFERENCES

- Kari, L. 1991. On insertion and deletion in formal languages. Ph.D. thesis. University of Turku.
- Cui, B., Kari, L., Seki, S. 2011. Block insertion and deletion on trajectories. *Theoretical Computer Science*. 412, 714-728.
- Daley, M., Kari, L., Gloor, G., and Siromoney, R. 1999, 'Circular contextual insertions/deletions with applications to biomolecular computation' in Proceedings of International Symposium on String Processing and Information Retrieval (SPIRE 1999), Cancun, Mexico, pp. 47-54.
- Ito, M., Kari, L., and Thierrin, G. 1997. Insertion and deletion closure of languages. *Theoretical Computer Science*. 183. 3-19.
- Ito, M., Kari, L., and Thierrin, G. 2000. Shuffle and scattered deletion closure of languages. *Theoretical Computer Science*. 245(1), 115-133.
- Kari, L. 1992. Insertion and deletion of words: determinism and reversibility. *Mathematical Foundations of Computer Science*. 315-327.
- Kari, L. 1993. Generalized derivatives. *Fundamenta Informaticae*. 18(1), 61-77.
- Kari, L., Mateescu, A., Paun, G., and Salomaa, A. 1993. Deletion sets. *Fundamenta Informaticae*. 19, 355-370.

- Kari, L., Mateescu, A., Paun, G., and Salomaa, A. 1995. On parallel deletions applied to a word. *RAIRO – Theoretical Informatics and Applications*. 29(2), 129-144.
- Kari, L., Paun, G., Thierrin, G., and Yu, S. 1999, 'At the crossroads of DNA computing and formal languages: characterizing RE using insertion-deletion systems' in Proceedings of 3rd DIMACS Workshop on DNA Based Computing, University of Pennsylvania, Pennsylvania, pp. 329-347.
- Kari, L., and Thierrin, G. 1996. Contextual insertions/deletions and computability. *Information and Computation*. 131(1), 47-61.
- Krassovitskiy, A. 2011. Complexity and modeling power of insertion-deletion systems. Ph.D. thesis. Universitat Rovira i Virgili.
- Fong W. H., Holzer, M., Truthe, B., Turaev, S., and Yosman, A. F. 2016, 'On bonded sequential and parallel insertion systems' in Eighth Workshop on Non-Classical Models of Automata and Applications (NCMA) Proceedings, Debrecen, Austria, pp. 163-178.
- Rozenberg, G., and Salomaa, A. (1980). *The Mathematical Theory of L-systems*. Cambridge, MA: Academic Press.
- Rozenberg, G., and Salomaa, A. (1997). *Handbook of Formal Languages*. Berlin: Springer-Verlag.