Incomparability of simple and one-sided/regular sticker languages

Peter Weigel and

Institut für Informatik, Fachbereich Mathematik/Informatik,
Martin-Luther-Universität Halle-Wittenberg, D-06120 Halle/Saale, Germany
email: mail@stickersysteme.de, url: http://www.stickersysteme.de

Februar 2005

Abstract

This paper shows that any of the classes $SSL(b)$ and $SSL(n)$ is incomparable to any of the classes $OSL(b)$, $OSL(n)$, $RSL(b)$ and $RSL(n)$. This answers some of the questions left open in [KPG98], [FPR98], [PR98] and [PRS98] concerning the expressive power of sticker systems compared to Chomsky grammars.

Key words: dna-computing, sticker system, chomsky grammar, complexity analysis

1 Introduction

In [Adl94] L. M. Adleman gives a procedure for solving the *Hamiltonian Path Problem (HPP)* based on DNA strands. This procedure, known as *Adleman's Experiment*, can be considered as the basis of the concept of *sticker systems*, which was introduced in [KPG98] as *regular sticker systems*. Sticker systems with capabilities of synchronizing the extension on the left and right were mentioned first in [FPR98] as *bidirectional sticker systems*. In [PR98] both concepts were merged to a new concept called *sticker systems*. The definition of sticker systems from [PR98] and a lot of results and proofs from [KPG98], [FPR98] and [PR98] were thereafter summarized and supplemented in [PRS98].

Sticker systems are one of many ways for theoretical analysis of properties and capabilities of DNA strands, which are interesting for language, computation or complexity theories. Because the constructs of sticker systems, which can be considered as grammars, are completely different from Chomsky grammars, an explicit analysis is needed. In [KPG98], [FPR98], [PR98] and [PRS98] there are a lot of proofs for relations between sticker language families among themselves and Chomsky language families.

Here we will prove the incomparability of $SSL(b)$ and $SSL(n)$ with $OSL(b)$, $OSL(n)$, $RSL(b)$ and $RSL(n)$ and therewith answer the open question of [PRS98] regarding the relation between simple sticker languages and one-sided or rather regular sticker languages. Thereafter, we will use this new result for some interesting conclusions.

This work is based on the diploma thesis [Wei04] and only an abridged version of the analysis given there. A more complex version of the proof from Lemma 3.1 was already published in [KW04].

2 Basic definitions

By $\mathbb{N}$ we denote the set of non-negative integers. The set of all subsets of a set A is denoted by $P(A)$. The *empty set* is denoted by $\emptyset$.

An *alphabet* is a nonempty, finite set of abstract symbols. The elements of an alphabet are called *letters*. Let Σ be an alphabet. A *word* over Σ is a finite sequence of letters of Σ. Σ^* denotes the set of all words over the alphabet Σ including the *empty word* ε. We define $\Sigma^+ := \Sigma^* \setminus \{\varepsilon\}$. A *language* L over the alphabet Σ is a subset of Σ^*. The *complement* of a language L is denoted by L^{co} and defined by $L^{co} := \{w \in \Sigma^* : w \notin L\}$. Let $k, i \in \mathbb{N}$ and $w = a_1 a_2 \ldots a_k \in \Sigma^*$ be a word. We call $w^R := a_k \ldots a_2 a_1$ the *reversion*, $|w| := k$ the *length* and $w[i] := a_i$ the i-th *letter* of w if $1 \leq i \leq k$. Additionally, we define $w[i] := \varepsilon$ for $i < 1$ or $i > k$. The *concatenation* of two words u and v with $u = a_1 \ldots a_k$ and $v = b_1 \ldots b_m$ is defined by $u \cdot v := a_1 \ldots a_k b_1 \ldots b_m$.

Let $k \in \mathbb{N}$. Σ^k denotes the k-fold cartesian product of a nonempty set Σ. The elements of Σ^k are called *vectors*. Let $\Sigma_1, \Sigma_2, \ldots, \Sigma_k$ be a finite sequence of nonempty sets and $x = (x_1, x_2, \ldots, x_k) \in \Sigma_1 \times \Sigma_2 \times \ldots \times \Sigma_k$. We call $x[i] := x_i$ the i-th *component* of x if $1 \leq i \leq k$.

Let $k \in \mathbb{N}$ and $\lambda : A^k \to B$ be a partial mapping from A^k into B. In general, we define the extension $\lambda : P(A)^k \to P(B)$ by

$$\lambda(X_1, X_2, \ldots, X_k) := \left\{ \lambda(x_1, x_2, \ldots, x_k) : \begin{array}{c} x_i \in X_i \text{ for } 1 \leq i \leq k, \\ \lambda(x_1, x_2, \ldots, x_k) \text{ is defined} \end{array} \right\}.$$

The *reversion* A^R and the *concatenation* $A \cdot B$ of two languages A and B are therewith defined.

Let V be an alphabet and $\rho \subseteq V \times V$ be a symmetrical binary relation. There are $\binom{V^*}{V^*} := \left\{ \binom{u}{v} : u, v \in V^* \right\}$ the set of all *pairs of words* [1] of V^* and $\begin{bmatrix} V^* \\ V^* \end{bmatrix}_\rho := \left\{ \binom{u}{v} \in \binom{V^*}{V^*} : |u| = |v|, (u[i], v[i]) \in \rho \text{ for } 1 \leq i \leq |u| \right\}$ the set of all *pairs of complementary words* of V^*. For $\binom{u}{v} \in \begin{bmatrix} V^* \\ V^* \end{bmatrix}_\rho$ we write $\begin{bmatrix} u \\ v \end{bmatrix}_\rho$. The concatenation $\binom{x_1}{x_2} \cdot \binom{y_1}{y_2}$ is $\binom{x_1 \cdot y_1}{x_2 \cdot y_2}$. Analogously, we write $\begin{bmatrix} x_1 \cdot y_1 \\ x_2 \cdot y_2 \end{bmatrix}_\rho$ for $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_\rho \cdot \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}_\rho$.

The set of all *dominoes* $W_\rho(V)$ is defined by $W_\rho(V) := S_\rho(V) \cup LR_\rho(V)$, where $S_\rho(V) := \left\{ \binom{u}{v} \in \binom{V^*}{V^*} : u = \varepsilon \text{ or } v = \varepsilon \right\}$ is called set of *simple dominoes* and $LR_\rho(V) := S_\rho(V) \times \left(\begin{bmatrix} V^* \\ V^* \end{bmatrix}_\rho \setminus \left\{ \begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}_\rho \right\} \right) \times S_\rho(V)$ is called set of *non-simple dominoes*. The set $WK_\rho(V) := \left\{ \binom{\varepsilon}{\varepsilon} \right\} \times \left(\begin{bmatrix} V^* \\ V^* \end{bmatrix}_\rho \setminus \left\{ \begin{bmatrix} \varepsilon \\ \varepsilon \end{bmatrix}_\rho \right\} \right) \times \left\{ \binom{\varepsilon}{\varepsilon} \right\}$ is called set of *complete dominoes*. The domino $\binom{\varepsilon}{\varepsilon}$ is identified by ε and therewith we can write $\begin{bmatrix} x \\ y \end{bmatrix}_\rho$ instead of $\binom{\varepsilon}{\varepsilon} \begin{bmatrix} x \\ y \end{bmatrix}_\rho \binom{\varepsilon}{\varepsilon}$.

Let $x \in LR_\rho(V)$ and $y \in S_\rho(V)$ be two dominoes. x_1^t, x_1^b, x_2^t, x_2^b, x_3^t, x_3^b, y^t and y^b denote the single components of x and y with $x = \left(\binom{x_1^t}{x_1^b}, \begin{bmatrix} x_2^t \\ x_2^b \end{bmatrix}_\rho, \binom{x_3^t}{x_3^b} \right)$ and $y = \binom{y^t}{y^b}$. Additionally, we call $x_2 := \begin{bmatrix} x_2^t \\ x_2^b \end{bmatrix}_\rho$ the *centerpiece*, $x_1 := \binom{x_1^t}{x_1^b}$ the *left* and $x_3 := \binom{x_3^t}{x_3^b}$ the *right delay* of x. We write $x = x_1 x_2 x_3$ instead of $x = (x_1, x_2, x_3)$. The words y^t and $x^t := x_1^t \cdot x_2^t \cdot x_3^t$ are called *upper strand*. Analogously, y^b and $x^b := x_1^b \cdot x_2^b \cdot x_3^b$ are called *lower strand*. The letters of the single components are called *bases*.

The *structure* of a domino x is defined by the mapping $struct : W_\rho(V) \rightarrow W_{\{(\sharp,\sharp)\}}(\{\sharp\})$, whereby $struct(x)$ arises from x by substituting all bases contained therein by $\sharp$.

The *length of the delay* of a domino is defined by the mapping $d : W_\rho(V) \rightarrow \mathbb{N}$ with

$$d(x) := \begin{cases} max \left\{ |x_1^t|, |x_1^b|, |x_3^t|, |x_3^b| \right\} & \text{if } x \in LR_\rho(V), \\ max \left\{ |x^t|, |x^b| \right\} & \text{if } x \in S_\rho(V). \end{cases}$$

Let $x, y \in W_\rho(V)$ be two dominoes. The *sticking* of x and y is defined by the

[1] Therewith $\binom{V^*}{V^*}$ is just another term of $V^* \times V^*$.

mapping $\mu_\rho : W_\rho(V) \times W_\rho(V) \to W_\rho(V)$ with

$$\mu_\rho(x,y) := \begin{cases} x_1 \left(x_2 \cdot \begin{bmatrix} u \\ v \end{bmatrix}_\rho \cdot y_2 \right) y_3 & \text{if } x \in LR_\rho(V), y \in LR_\rho(V), \\ & x_3 \cdot y_1 = \begin{bmatrix} u \\ v \end{bmatrix}_\rho, \\ x_1 \left(x_2 \cdot \begin{bmatrix} u \\ v \end{bmatrix}_\rho \right) w & \text{if } x \in LR_\rho(V), y \in S_\rho(V), \\ & x_3 \cdot y = \begin{bmatrix} u \\ v \end{bmatrix}_\rho \cdot w, w \in S_\rho(V), \\ w \left(\begin{bmatrix} u \\ v \end{bmatrix}_\rho \cdot x_2 \right) x_3 & \text{if } x \in S_\rho(V), y \in LR_\rho(V), \\ & x \cdot y_1 = w \cdot \begin{bmatrix} u \\ v \end{bmatrix}_\rho, w \in S_\rho(V), \\ \text{undefined} & \text{otherwise.} \end{cases}$$

Because μ_ρ is associative, we write $x \cdot_\rho y$ instead of $\mu_\rho(x,y)$.

A *sticker system* is a construct

$$\gamma = (V, \rho, A, D)$$

with an alphabet V, a symmetrical binary relation $\rho \subseteq V \times V$, a finite set $A \subseteq LR_\rho(V)$ and a finite set $D \subseteq W_\rho(V) \times W_\rho(V)$. The relation ρ is called *complementarity* of V. The elements of A are called *axioms* and the elements of D are called *rules*.

Let $x, y \in W_\rho(V)$ be two dominoes. We write $x \to_\gamma y$ if and only if there is a rule $(u,v) \in D$ with $y = u \cdot_\rho x \cdot_\rho v$. We write $x \to_\gamma^k y$ for $x = x_0 \to_\gamma x_1 \to_\gamma x_2 \to_\gamma \cdots \to_\gamma x_k = y$ with $k \in \mathbb{N}$ and $x_i \in W_\rho(V)$ for $0 \leq i \leq k$ or rather $x \to_\gamma^* y$ if and only if there is such a k and call this a *derivation* if and only if $x \in A$ and a *complete derivation* if and only if it is $y \in WK_\rho(V)$, additionally.

Let $C^0(\gamma) := A$ and $C^k(\gamma) := \left\{ y \in W_\rho(V) : \exists x \in C^{k-1}(\gamma) : x \to_\gamma y \right\}$ with $k \in \mathbb{N}$ and $k \geq 1$. $C^*(\gamma) := \bigcup_{k \in \mathbb{N}} C^k(\gamma)$ denotes the set of *dominoes generated by* γ, $LM(\gamma) := C^*(\gamma) \cap WK_\rho(V)$ the *language of molecules generated by* γ and $L(\gamma) := \{x^t : x \in LM(\gamma)\}$ the *language generated by* γ.

It is $\varepsilon \notin L(\gamma)$ for every sticker system $\gamma = (V, \rho, A, D)$. Now we extend the definition of sticker systems and allow $\binom{\varepsilon}{\varepsilon} \in A$. Thereby we have to ensure, that this special axiom will never be used for derivations. It is $\varepsilon \in L(\gamma)$ if and only if $\binom{\varepsilon}{\varepsilon} \in A$.

A rule $(u,v) \in D$ is called *simple* if and only if both dominoes are simple, *left-sided* if and only if $v = \varepsilon$, *right-sided* if and only if $u = \varepsilon$ and *one-sided* if and only if it is left-sided or right-sided. A derivation $x_0 \to_\gamma^* x_k$ is called *delay bounded by the bound* $d \in \mathbb{N}$ if and only if $d(x_i) \leq d$ for $0 \leq i \leq k$. A sticker system $\gamma = (V, \rho, A, D)$ is called *simple* if and only if all rules of D are simple, *one-sided* if and only if all rules in D are one-sided, *regular* if and only if all rules in D are right-sided and *with bounded delay* if and only if there is a constant $d \in \mathbb{N}$, such that for every domino $x \in LM(\gamma)$ there is at least one delay bounded derivation with the delay bound d.

$ASL(n)$ denotes the family of languages generated by sticker systems. Restriction to sticker systems with bounded delay is denoted by substituting n by b. Restrictions to simple, one-sided, regular, simple and one-sided or simple and regular sticker systems are denoted by substituting A by S, O, R, SO or SR.

2.2 Chomsky grammars

By CS, CF, LIN and REG we denote the families of languages, which are generated by *context-sensitive*, *context-free*, *linear* and *regular* Chomsky grammars e.g. defined in [WW86, Section 4.1.1].

Lemma 2.1 ([WW86, Theorem 4.6]) *$REG \subset LIN \subset CF \subset CS$.*

3 Complementarity lemma

Lemma 3.1 (cf. [PRS98, Lemma 5.8]) *For every sticker system $\gamma = (V, \rho, A, D)$ there exists an effectively constructable sticker system $\gamma' = (V, \rho', A', D')$ with $L(\gamma) = L(\gamma')$ and $\rho' = \{(x, x) : x \in V\}$. Additionally, the transformation from γ to γ' preserves any property[2] of rules and derivations defined in this publication or in [PRS98].*

Proof. The proof is a transcription and generalization of the proof of [PRS98, Lemma 5.8] concerning Watson-Crick finite automata.

Let $\gamma = (V, \rho, A, D)$ be a sticker system.

[2] The transformation of γ to γ' preserves the properties simple, one-sided, right-sided, with bounded delay, ...

The mapping $\lambda_\rho : \binom{V^*}{V^*} \to P\left(\binom{V^*}{V^*}\right)$ is defined as follows:

$$\lambda_\rho\left(\binom{a}{b}\right) := \begin{cases} \left\{\binom{a}{a}\right\} & \text{if } a \in V, b \in V, (a,b) \in \rho, \\ \left\{\binom{a}{\varepsilon}\right\} & \text{if } a \in V, b = \varepsilon, \\ \left\{\binom{\varepsilon}{\varepsilon}\right\} & \text{if } a = \varepsilon, b = \varepsilon, \\ \left\{\binom{\varepsilon}{c} : (c,b) \in \rho\right\} & \text{if } a = \varepsilon, b \in V, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

If we define the extensions of λ_ρ on $S_\rho(V)$ and $\begin{bmatrix}V^*\\V^*\end{bmatrix}_\rho$ by $\lambda_\rho(u{\cdot}v) := \lambda_\rho(u){\cdot}\lambda_\rho(v)$, on $LR_\rho(V)$ by $\lambda_\rho(x_1x_2x_3) := \lambda_\rho(x_1) \times \lambda_\rho(x_2) \times \lambda_\rho(x_3)$ and on sets of dominoes by $\lambda_\rho(M) := \bigcup_{w\in M} \lambda_\rho(w)$, then the transformation λ_ρ of dominoes is structure-preserving with no substitutions in the upper strand, and a basis α of the lower strand migrates to a basis β if and only if β could be placed in the upper strand directly over the basis α with respect to the complementarity ρ or rather if it is already placed there.

Let $\gamma' = (V', \rho', A', D')$ be the sticker system with $V' := V$, $\rho' := \{(x,x) : x \in V'\}$, $A' := \lambda_\rho(A)$ and $D' := \bigcup_{(u,v)\in D} \lambda_\rho(u) \times \lambda_\rho(v)$. Then $L(\gamma) = L(\gamma')$.

To this aim one can prove $\lambda_\rho(u \cdot_\rho v) = \lambda_\rho(u) \cdot_{\rho'} \lambda_\rho(v)$ for any $u, v \in W_\rho(V)$ and thereby show, that the transformation λ_ρ is an homomorphism regarding the sticking $\cdot_\rho$ and $\cdot_{\rho'}$. Thereafter, one can show the relation $\lambda_\rho(C^k(\gamma)) = C^k(\gamma')$ and consequently $\lambda_\rho(C^*(\gamma)) = C^*(\gamma')$ by using induction over $k \in \mathbb{N}$. Because λ_ρ is structure-preserving with no substitutions in the upper strand, we can conclude $L(\gamma) = L(\gamma')$.

4 Simple sticker systems are more powerful than one-sided/regular sticker systems

Theorem 4.1 *$SSL(b) \not\subseteq REG$.*

Proof. This result is already known, but not mentioned in [PRS98]. For the sake of completeness, we will give a proof.

Let $\gamma = (V, \rho, A, D)$ be the sticker system with $V = \{a, b\}$, $\rho = \{(x,x) : x \in V\}$, $A = \left\{\begin{bmatrix}a\\a\end{bmatrix}, \begin{bmatrix}b\\b\end{bmatrix}, \begin{bmatrix}aa\\aa\end{bmatrix}, \begin{bmatrix}bb\\bb\end{bmatrix}, \binom{\varepsilon}{\varepsilon}\right\}$ and $D = \left\{\left(\binom{a}{\varepsilon}, \binom{a}{\varepsilon}\right), \left(\binom{b}{\varepsilon}, \binom{b}{\varepsilon}\right), \left(\binom{\varepsilon}{a}, \binom{\varepsilon}{a}\right), \left(\binom{\varepsilon}{b}, \binom{\varepsilon}{b}\right)\right\}$. Then $L(\gamma) \in SSL(b) \setminus REG$.

Obviously, γ is a simple sticker system. Additionally, it is a sticker system with bounded delay, because generations of the upper strand and lower strand are independent of each other and so every derivation can be transformed into an equivalent derivation with a delay bounded by $d = 1$ by resorting rule usages. Consequently, there is $L(\gamma) \in SSL(b)$.

Let $S_1 := \left\{w \in \{a,b\}^* : w = w^R\right\}$. For every word $w \in S_1$ one can find a suitable complete derivation of γ. On the other hand, one can show that every complete derivation of γ is a derivation of a word $w \in S_1$. Therewith we get $L(\gamma) = S_1$. One can also simply prove $S_1 \notin REG$ by using the Pumping Lemma for regular Chomsky grammars e.g. presented in [HU79, Lemma 3.1].

Theorem 4.2 ([PRS98, Theorem 4.1 + Theorem 4.4])

$$REG = OSL(b) = OSL(n) = RSL(b) = RSL(n).$$

Corollary 4.3

$$SSL(n) \supseteq SSL(b) \not\subseteq REG = OSL(b) = OSL(n) = RSL(b) = RSL(n).$$

5 One-sided/regular sticker systems are more expressive than simple sticker systems

Let $d \in \mathbb{N}$, then the order relations $\leq$ and $<$ on $\mathbb{N}^d$ are defined by

$$\begin{aligned} x \leq y &\iff \forall\, 1 \leq i \leq d : x[i] \leq y[i], \\ x < y &\iff x \leq y \text{ and } x \neq y. \end{aligned}$$

Lemma 5.1 ([Dic13], [Hig52]) *Let $d \in \mathbb{N}$. For $(\mathbb{N}^d, <)$ there is no infinite set of pairwise incomparable elements of $\mathbb{N}^d$.*

Proof. This result was firstly proved in [Dic13] and later generalized in [Hig52]. Nevertheless we give a proof, because this result is the core of Theorem 5.2.

We show by induction on the dimension d, that every set of pairwise incomparable elements of $\mathbb{N}^d$ is finite.

Let $d = 0$. Because of $\mathbb{N}^d = \{\varepsilon\}$, every subset of $\mathbb{N}^d$ is finite.

Let $d > 0$. The empty set is finite. Let M be a nonempty set of pairwise incomparable elements of $\mathbb{N}^d$. Because M is not empty, there exists an element $x \in M$.

Let $T(i,k) := \{y \in M : y[i] = k\}$ with $i \in \{1, ..., d\}$ and $k \in \{0, ..., x[i]\}$ and let $T = \bigcup_{i,k} T(i,k)$. Obviously, $T \subseteq M$. Let $y \in M$, then there exists an i with $y[i] \leq x[i]$. Otherwise there would be $x < y$, a contradiction. Consequently, $y \in T(i, y[i]) \subseteq T$. $M = T$ is proved.

We have $y[i] = k$ for all elements y of $T(i,k)$. Consequently, all these elements are located in a space with the dimension $d - 1$ (we hide the component i by projection). Furthermore, all these elements are pairwise incomparable, because of $T(i,k) \subseteq T = M$. By using the induction hypothesis we conclude the finiteness of $T(i,k)$.

The set T is a finite union of finite sets, because there exists only a finite amount of sets $T(i,k)$ and any of these sets is finite. Consequently, T and therewith M is finite.

Theorem 5.2 *$REG \not\subseteq SSL(n)$.*

Proof. Let $L := \{a\}^* \cup \{b\}$. Obviously, $L \in REG$.

Assume $L \in SSL(n)$. Then there would exist a simple sticker system $\gamma = (V, \rho, A, D)$ with $L(\gamma) = L$ and $\rho = \{(x,x) : x \in V\}$ by Lemma 3.1.

The word b or rather the domino $\begin{bmatrix} b \\ b \end{bmatrix}_\rho$ can only be generated by an axiom. Because all the other generatable complete dominoes only contain a's, we can assume that all rules and all other axioms only contain a's as well, without any restrictions.

In order to stick a rule d with a domino x, there must be ensured some structure and complementary conditions. Because there exist only simple rules, there are no structure conditions. Because all rules are rules over a one-letter alphabet and the only axiom containing other letters does not have delays, there are no complementary conditions as well. Consequently, at any time every rule is applicable and the generated domino is independent from the order of rule usages.

Let $A = \{a_1, ..., a_n\}$, $D = \{d_1, ..., d_m\}$ and $\gamma_i := (V, \rho, \{a_i\}, D)$ with $1 \leq i \leq n$. Then $L(\gamma) = \bigcup_i L(\gamma_i)$. Because this union is finite and $|L(\gamma)| = \infty$, there is a γ_k with $|L(\gamma_k)| = \infty$.

Let P be the set of all complete derivations of γ_k with first only usages of rule d_1, then rule d_2, then .... Such a derivation can be described as a tuple

$(c_1, \ldots, c_m) \in \mathbb{N}^{|D|}$. Thereby c_i stands for the number of times rule d_i is used. Consequently, we can assume $P \subseteq \mathbb{N}^{|D|}$, without any restrictions. By $mol(x)$ we denote the complete domino generated by the derivation x starting with the axiom a_k. If x and y are two derivations from P with $x < y$, then $mol(y)$ can be derived from $mol(x)$.

For each complete domino $x \in L(\gamma_k)$ there exists a corresponding derivation in P. Because of $|L(\gamma_k)| = \infty$, it is $|P| = \infty$. Because of Lemma 5.1 there is no infinite set of pairwise incomparable elements of $\mathbb{N}^{|D|}$. Consequently, there are two comparable elements $x, y \in P$ with $x < y$. So we conclude $mol(y) = \binom{a^r}{a^r} \cdot mol(x) \cdot \binom{a^s}{a^s}$ with $r, s \in \mathbb{N}$, $r \neq 0$ or $s \neq 0$ and $mol(x) \rightarrow^*_{\gamma_k} mol(y)$.

The rules used to derivate $mol(y)$ from $mol(x)$, now applied to the axiom $\begin{bmatrix} b \\ b \end{bmatrix}_\rho$, constitute in γ a derivation of the domino $\begin{bmatrix} a^r b a^s \\ a^r b a^s \end{bmatrix}_\rho$. Consequently, $w = a^r b a^s \in L(\gamma) = L$. A contradiction.

Remark 5.3 *The relation $L \notin SSL(b)$ with $L := \{a\}^* \cup \{b\}$ can be shown easier: Assume there exists a simple sticker system γ with $L(\gamma) = L$ and with a by $d \in \mathbb{N}$ bounded delay. Then, without any restrictions, there would be only $(2d+1)^2$ pairwise different delays. Because of $|L| = \infty$, $|A| < \infty$ and $|D| < \infty$, there are arbitrarily long derivations, which are delay bounded by d. For every derivation with at least a length of $(2d+1)^2 + 1$, there is at least one delay which appears at least twice. Thereby, like in Theorem 5.2, we have found a (lot of) delay restoring rule sequence(s) and therewith, we can derivate a word w with $w \notin L$. A contradiction.*

Remark 5.4 *In [PRS98, Theorem 4.7] there is given a proof for $L_{bab} \in REG \setminus SOSL(n)$ with $L_{bab} := \{ba^n b : n \in \mathbb{N}\}$. Analogously to Theorem 5.2, this result can be improved to $L_{bab} \notin SSL(n)$. In contrast to Theorem 5.2 now there exist rules containing b's. Fortunately, the (maximal four) uses of these rules per complete derivation can be moved to the end of the derivation and therewith hidden from our attention, because of $|D| < \infty$ and analogously to the restriction to one axiom.*

Corollary 5.5

$$OSL(b) = OSL(n) = RSL(b) = RSL(n) = REG \not\subseteq SSL(n) \supseteq SSL(b).$$ [3]

[3] Beside the incomparability of $SSL(b)$ and $SSL(n)$ with $OSL(b) = OSL(n) = RSL(b) = RSL(n) = REG$ this even shows the incomparability of $SSL(n)$ with LIN and CF by application of [PRS98, Theorem 4.2].

6 Conclusions

Now we will give some results, which are direct or indirect conclusions of the previous two sections. This collection isn't complete.

Conclusion 6.1 *$SOSL(b) \not\subseteq SRSL(n)$.* [4]

Proof. Let L_{ab} be the set $\{a^n b : n \in \mathbb{N}\}$. Obviously, $L_{ab} \in SOSL(b)$.
Assume $L_{ab} \in SRSL(n)$. Then there would be a simple regular sticker system γ with $L(\gamma) = L_{ab}$. Analogous to $L_{bab} \notin SSL(n)$ in Remark 5.4 one can show, that the domino $\begin{bmatrix} ba^s \\ ba^s \end{bmatrix}$ or rather the word $w = ba^s$ for a $s \geq 1$ would be derivable. A contradiction.

Conclusion 6.2 *The language families $SRSL(n)$ and $SRSL(b)$ are not closed under reversion.*

Proof. Let L_{ba} be the set $\{ba^n : n \in \mathbb{N}\}$. Obviously, $L_{ba} \in SRSL(b)$. According to Lemma 6.1 there is $L_{ba}^{co} = L_{ab} \notin SRSL(n)$.

Lemma 6.3 *The language families $ASL(x)$, $SSL(x)$, $OSL(x)$ and $SOSL(x)$ with $x \in \{b, n\}$ are closed under reversion.*

(Without a proof. A proof can be found by reversing axioms and rules.)

Conclusion 6.4 *The language families $SSL(x)$ and $SOSL(x)$ with $x \in \{b, n\}$ are not closed under intersection.*

Proof. Let L_{baba} be the language $L_{baba} := \{ba^n ba^{2m} : n, m \in \mathbb{N}\}$ and $L_{bab} := \{ba^n b : n \in \mathbb{N}\}$. There is a simple, regular sticker system γ with bounded delay with

$$\gamma = \left(\{a, b\}, \{(a, a), (b, b)\}, \left\{ \begin{bmatrix} bb \\ bb \end{bmatrix}, \begin{bmatrix} b \\ b \end{bmatrix} \begin{pmatrix} a \\ \varepsilon \end{pmatrix}, \begin{bmatrix} ba \\ ba \end{bmatrix} \begin{pmatrix} a \\ \varepsilon \end{pmatrix} \right\}, D \right),$$
$$D = \left\{ \left(\begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}, \begin{pmatrix} \varepsilon \\ aa \end{pmatrix} \right), \left(\begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}, \begin{pmatrix} aa \\ \varepsilon \end{pmatrix} \right), \left(\begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}, \begin{pmatrix} \varepsilon \\ ab \end{pmatrix} \right), \left(\begin{pmatrix} \varepsilon \\ \varepsilon \end{pmatrix}, \begin{pmatrix} b \\ \varepsilon \end{pmatrix} \right) \right\}$$

and $L(\gamma) = L_{baba}$. Therewith the termed language families contain the language L_{baba}. Because of Lemma 6.3 these language families even contain L_{baba}^R. But they do not contain $L_{bab} = L_{baba} \cap L_{baba}^R$ because of Remark 5.4.

[4] The inclusions $SRSL(b) \subset SOSL(b)$ and $SRSL(n) \subset SOSL(n)$ are some direct results of this conclusion.

Conclusion 6.5 *The language families $SSL(x)$, $SOSL(x)$ and $SRSL(x)$ with $x \in \{b, n\}$ are not closed under union, not closed under complement, not closed under intersection with regular Chomsky languages and not closed under concatenation (with a single letter).*

Proof. We define $L_0 := \{a\}^* \cup \{b\}$, $L_1 := \{ba^n b : n \in \mathbb{N}\}$, $L_2 := \{a, b\}^*$, $L_3 := \{a\}^*$, $L_4 := \{b\}$, $L_5 := \{u \cdot b \cdot v : u, v \in \{a, b\}^*, |u \cdot v| \geq 1\}$, $L_6 := \{ba^n : n \in \mathbb{N}\}$, $L_7 := L_3 \cup L_4$, $L_8 := L_5^{co}$, $L_9 := L_2 \cap L_7$ and $L_{10} := L_6 \cdot L_4$. Obviously, the termed language families contain L_2, L_3, L_4, L_5 and L_6. (The proof of $L_5 \in SRSL(b)$ goes by construction of a simple, regular sticker system with bounded delay. The construction is similar to that one in the proof of Conclusion 6.4.) On the other hand they don't contain $L_0 = L_7 = L_8 = L_9$, because of Theorem 5.2, and because of Remark 5.4 they don't contain $L_1 = L_{10}$.

Conclusion 6.6 *$SSL(x) \cup OSL(x) \subset ASL(x)$. ($x \in \{b, n\}$)*

Proof. Let $L_1 := \{a\}^* \cup \{b\}$, $L_2 := \left\{w \in \{c, d\}^* : w = w^R\right\}$ and $L_3 := L_1 \cup L_2$ be three languages. Then there are $L_1 \notin SSL(x)$ and $L_1 \in OSL(x)$. Additionally, there are $L_2 \in SSL(x)$ and $L_2 \notin OSL(x)$. Therewith, one can easily prove $L_3 \notin SSL(x) \cup OSL(x)$. In contrast, it is $L_3 \in LIN \subseteq ASL(x)$ according to [PRS98, Theorem 4.5].

7 Acknowledgments

We are grateful to Ludwig Staiger and Dietrich Kuske for their helpful advices and suggestions.

References

[Adl94] L. M. Adleman. *Molecular computation of solutions to combinatorial problems.* Science, Vol. 226, November 1994, 1021–1024.

[KPG98] L. Kari, G. Păun, G. Rozenberg, A. Salomaa, S. Yu. *DNA-Computing, sticker systems and universality.* Acta Informatica, 35, 5, 1998, 401–420.

[FPR98] R. Freund, G. Păun, G. Rozenberg, A. Salomaa. *Bidirectional sticker systems.* Third Annual Pacific Conference on Biocomputing, Hawaii, 1998 / World Scientific, Sigapore, 1998, 535–546.

[PR98] G. Păun, G. Rozenberg. *Sticker systems.* Theoretical Computer Science, 204, 1998, 183–203.

[PRS98] G. Păun, G. Rozenberg, A. Salomaa. *DNA-Computing. New Computing Paradigms.* Springer, Berlin Heidelberg, 1998.

[Wei04] Peter Weigel. *Ausdrucksstärke von Stickersystemen. Untersuchung der Ausdrucksstärke von Stickersystemen durch Vergleich mit Chomskygrammatiken und Mehrkopfautomaten.* Diplomarbeit, Martin-Luther-Universität Halle-Wittenberg, Institut für Informatik, Halle/Saale, Oktober 2004.

[KW04] D. Kuske, P. Weigel. *The rôle of the complementarity relation in Watson-Crick automata and sticker systems.* Developments in Language Theory: 8th International Conference, DLT 2004. Auckland, New Zealand, December 13-17. Proceedings. / Lecture Notes in Computer Science, Springer, Heidelberg, 3340, 2004, 272 – 283.

[Dic13] L. Dickson. *Finiteness of the odd perfect and primitive abundant numbers with n distinct prime factors.* American Journal of Mathematics, 35, 1913, 413–426.

[Hig52] G. Higman. *Ordering by divisibility in abstract algebras.* Proceedings of the London Mathematical Society, (3) 2(7), 1952, 326–336.

[WW86] K. Wagner, G. Wechsung. *Computational Complexity.* Deutscher Verlag der Wissenschaften, Berlin, 1986.

[HU79] J. E. Hopcroft, J. D. Ullman. *Introduction to automata theory, languages and computation.* Addison-Wesley, 1979.