CONTEXT IN PARALLEL REWRITING*

K. Čulik II and J. Opatrný
Department of Applied Analysis and Computer Science
University of Waterloo
Waterloo, Ontario, Canada

Abstract

Three new types of context sensitive parallel rewriting systems, called global context L-systems, rule context L-systems and predictive context L-systems are introduced in this paper. We investigate the generative power of these new types of context sensitive parallel rewriting systems and we compare it to the generative power of TOL-systems [81], L-systems with interaction [92], regular grammars and context sensitive grammars.

1. Introduction

Parallel rewriting systems were introduced in [59, 60] as a mathematical model for biological developmental systems. Most of the papers related to parallel rewriting have dealt with rewriting systems of context free type, e.g. OL-systems [61], TOL-systems [81], and their generalisations [9], [89].

A generalisation of context sensitive grammars with parallel rewriting known as L-systems with interactions has been studied in [92]. L-systems with interactions have the same basic rules (productions) for rewriting as OL-systems, but with restriction on their use given by right and left "context". A rule may be applied only in the given context.

However, in the case of parallel rewriting it is quite natural to consider different forms of "context". Since we are replacing all symbols at once, we may restrict the use of a rule, $a \to \alpha$ say, by the context adjacent to α after simultaneously replacing all the symbols in a string rather than by the context adjacent to a before the rule was applied. We will call this kind of context, predictive context.

Even more generally, the restriction on the use of a rule may concern rules used on adjacent symbols. We will call this type of restriction rule context.

Clearly, all these generalisations make sense only in the case of parallel rewriting.

We can also consider restrictions on the use of rules, which in distinction to the above are of a global rather than a local character. In a global

* The research was supported by the National Research Council of Canada, Grant No. A7403.

context L-system, in addition to the set of labeled rules, a control set over their labels is given. We can only rewrite a string with a sequence of rules with labels from the control set.

The new types of context sensitive L-systems introduced in this paper also have a natural biological motivation. The development of a cell might be completely independent of the other cells, i.e. in OL-systems, or it might depend on the configuration around the cell before the development takes place i.e. in L-systems with interactions, or it might be restricted in such a way that only compatible cells can occur adjacently, i.e. in predictive context L-systems, or only compatible developments can occur adjacently, i.e. in rule-context L-systems, or even the development of an organism as a whole is restricted by certain patterns, e.g. the development can be different in certain parts of the organisms, i.e. in global context L-systems.

In this paper we investigate the generative power of these new types of L-systems. Among other results it is shown that global context L-systems with regular control sets (regular global context L-systems) are equivalent to rule context L-systems. We also show that the family of regular global context L-languages properly contains the family of languages generated by L-systems with interactions and the family of TOL-languages.

2. Preliminaries

We shall assume that the reader is familiar with the basic formal languages theory, e.g. [102].

Now, we will review the definitions of OL and TOL-systems [91], [81], and L-systems with interactions [92], and we will introduce some notation used throughout the paper.

Definition 1. A table OL-system (TOL-system) is a 3-tuple $G = (\Sigma, P, \sigma)$, where:

(i) Σ is a finite, nonempty set, called the alphabet.

(ii) P is a finite set of tables, $P = \{P_1, P_2, \ldots, P_n\}$ for some $n \geq 1$, where each P_i, $i = 1,2,\ldots,n$ is a finite subset of $\Sigma \times \Sigma^*$. Element (a,α) of P_i, $1 \leq i \leq n$, is called a rule and is usually written in the form $a \to \alpha$. P must satisfy the following condition of completeness. For each $a \in \Sigma$ and i, $1 \leq i \leq n$, there exists $\alpha \in \Sigma^*$ so that $(a,\alpha) \in P_i$.

(iii) $\sigma \in \Sigma^+$, the initial string of G.

Given a TOL-system $G = (\Sigma, P, \sigma)$, we write $\alpha \underset{G}{\Rightarrow} \beta$, where $\alpha \in \Sigma^+$, $\beta \in \Sigma^*$, if there exist $k \geq 1$, $a_1, a_2, \ldots, a_k \in \Sigma$, and $\beta_1, \beta_2, \ldots, \beta_k \in \Sigma^*$ so that $\alpha = a_1 a_2 \ldots a_k$, $\beta = \beta_1 \beta_2 \ldots \beta_k$ and for some table $P_i \in P$, $a_j \to \beta_j \in P_i$ for $1 \leq j \leq k$.

The transitive and reflexive closure of the binary relation $\underset{G}{\Rightarrow}$ is denoted by $\underset{G}{\Rightarrow}^*$.

The language generated by a TOL-system G is denoted by L(G) and is defined to be the set $\{\alpha \in \Sigma^*:\sigma \underset{G}{\Rightarrow}^* \alpha\}$.

Definition 2. A TOL-system $G = (\Sigma,P,\sigma)$ is called an OL-system if P consists of exactly one table of rules, i.e. $P = \{P_1\}$.

Notation. Throughout the paper if r is any binary relation, then r^* denotes the reflexive and transitive closure of r, without repeating it specifically in every case.

Notation. The empty string is denoted by ε. The length of a string α is denoted by $|\alpha|$. For any string α and $k \geq 1$, we define $\text{First}_k(\alpha)$ and $\text{Last}_k(\alpha)$ as follows.

$$\text{First}_k(\alpha) = \underline{\text{if}}\ |\alpha| \geq k\ \underline{\text{then}}\ \text{first } k \text{ symbols of } \alpha\ \underline{\text{else}}\ \alpha.$$

$$\text{Last}_k(\alpha) = \underline{\text{if}}\ |\alpha| \geq k\ \underline{\text{then}}\ \text{last } k \text{ symbols of } \alpha\ \underline{\text{else}}\ \alpha.$$

For any string α, we define

$$\text{First}_0(\alpha) = \varepsilon,\ \text{First}\ (\alpha) = \mathop{}_{k=1}^{|\alpha|} \{\text{First}_k(\alpha)\},$$

$$\text{Last}_0(\alpha) = \varepsilon,\ \text{Last}\ (\alpha) = \mathop{}_{k=1}^{|\alpha|} \{\text{Last}_k(\alpha)\}.$$

Definition 3. A context L-system is a 3-tuple $G = (\Sigma,P,\sigma)$, where

(i) Σ is a finite, nonempty set of symbols, called the alphabet.

(ii) P is a finite subset of $\{\#,\varepsilon\}\cdot\Sigma^* \times \Sigma \times \Sigma^*\cdot\{\#,\varepsilon\} \times \Sigma^*$, called the set of rules, where # is a symbol not in Σ called the endmarker. A rule $(\alpha,a,\beta,\gamma) \in P$ is usually written as $<\alpha,a,\beta> \rightarrow \gamma$.

(iii) $\sigma \in \Sigma^+$, the initial string.

Given a context L-system $G = (\Sigma,P,\sigma)$ we write $\alpha \underset{G}{\Rightarrow} \beta$ for $\alpha \in \Sigma^+$, $\beta \in \Sigma^*$, if there exist $k \geq 0$, $a_1,a_2,\ldots,a_k \in \Sigma$ and $\beta_1,\beta_2,\ldots,\beta_k \in \Sigma^*$ so that $\alpha = a_1a_2\ldots a_k$, $\beta = \beta_1\beta_2\ldots\beta_k$ and for every i, $1 \leq i \leq k$, there exist $m,n \geq 0$ such that $(\text{Last}_m(\#a_1a_2\ldots a_{i-1}),a_i, \text{First}_n(a_{i+1}a_{i+2}\ldots a_k\#),\beta_i) \in P$.

Context L-system G must be strongly complete, i.e. for any $\alpha \in \Sigma^+$ there exists $\beta \in \Sigma^*$ such that $\alpha \underset{G}{\Rightarrow} \beta$.

The language generated by a context L-system G is denoted by L(G) and is defined to be the set $\{\alpha \in \Sigma^*:\sigma \underset{G}{\Rightarrow}^* \alpha\}$.

Note. The definition of a context L-system given above is a simplification and an unessential generalisation of the definition of an L-system with interaction from [92]. It is obvious that both types of systems have the same generative power.

Notation. We say that a language L is a λ-language (where λ may be OL, TOL, context L, etc.) if there exists a λ-system G such that $L = L(G)$.

The family of context L-languages will be denoted by Ω.

If f is a mapping from Σ to subsets of Δ^*, then f can be extended to strings and languages over Σ as follows.

(i) $f(\varepsilon) = \{\varepsilon\}$.

(ii) for $a \in \Sigma$, $\alpha \in \Sigma^*$, $f(\alpha a) = f(\alpha)\cdot f(a)$, where "$\cdot$" is the operation of set concatenation.

(iii) for $L \subseteq \Sigma^*$, $f(L) = \{\alpha : \alpha \in f(\beta) \text{ for } \beta \in L\}$.

We will use these extended mappings later on without repeating the process of extension in every single case.

3. Context sensitive parallel rewriting systems

Now, we will define three different types of context sensitive parallel rewriting systems. All of them are using only one type of symbols, i.e. we are not considering any nonterminals.

First we will give the definition of global context L-systems. A global context L-system has, similarly as an OL-system, a finite set of context free rules, however, each rule has a finite number of labels. The use of rules in a global context L-system is restricted by a language over labels, called the control set.

Definition 4. A global context L-system is a 5-tuple $G = (\Sigma,\Gamma,P,C,\sigma)$, where:

(i) Σ is a finite, nonempty set of symbols, called the alphabet.

(ii) Γ is a finite, nonempty set of symbols, called the labels.

(iii) P is a finite, nonempty subset of $p(\Gamma) \times \Sigma \times \Sigma^*$, where $p(\Gamma)$ denotes the family of nonempty subsets of Γ. Element $(B,a,\alpha) \in P$ is called a rule and is usually written in the form $B{:}a \rightarrow \alpha$.

(iv) $C \subseteq \Gamma^*$, called the control set.

(v) $\sigma \in \Sigma^+$, the initial string.

Given a global context L-system $G = (\Sigma,\Gamma,P,C,\sigma)$, we write $\alpha \underset{G}{\Rightarrow} \beta$ for $\alpha \in \Sigma^+$, $\beta \in \Sigma^*$, if there exist $k \geq 1$, $a_1,a_2,\ldots,a_k \in \Sigma$, $\beta_1,\beta_2,\ldots,\beta_k \in \Sigma^*$ and $B_1,B_2,\ldots,B_k \in p(\Gamma)$ so that $\alpha = a_1a_2\ldots a_k$, $\beta = \beta_1\beta_2\ldots\beta_k$, $(B_j,a_j,\beta_j) \in P$, for $j = 1,2,\ldots,k$ and $B_1B_2\ldots B_k \cap C \neq \phi$.[1]

The language generated by a global context L-system G is denoted by L(G) and is defined to be the set $\{\alpha \in \Sigma^* : \sigma \Rightarrow^* \alpha\}$.

[1] $B_1B_2\ldots B_k$ is the concatenation of sets $B_1,B_2,\ldots,B_k$.

A global context L-system G is said to be a λ global context L-system if its control set is of the type λ. In this paper only regular global context L-systems will be studied and their control sets will be denoted by regular expressions.

The family of regular global L-languages will be denoted by Ψ.

Example 1 Let G_1 be a regular global context L-system, $G_1 = \{\{a\},\{S_1,S_2\},P,C,a\}$, where $P = \{\{s_1\}:a \rightarrow aa_1\{s_2\}:a \rightarrow aaa\}$ and C is denoted by regular expression $s_1^*+s_2^*$.

Clearly, at any step in a derivation, we can apply either the production $a \rightarrow aa$ to all symbols in a string, or the production $a \rightarrow aaa$ is used throughout the string. Therefore $L(G_1) = \{a^{2^i 3^j} : i \geq 0,\ j \geq 0\}$.

Since we may consider an L-system as a model of the development of a filamentous organism, it is natural to require that for any stage of the development there exists a next stage of the development. Therefore, a condition of "completeness" is usually included in definitions of all versions of L-systems.

Now, we will give the formal definitions of the completeness and strong completeness for regular global context L-systems.

Definition 5. Let G be a regular global L-system with an alphabet Σ. G is complete if for any $\alpha \in L(G)$, $\alpha \neq \varepsilon$, there exists $\beta \in \Sigma^*$ so that $\alpha \underset{G}{\Rightarrow} \beta$.

Definition 6. Let G be a regular global L-system with an alphabet Σ. G is strongly complete if for any $\alpha \in \Sigma^+$ there exists $\beta \in \Sigma^*$ so that $\alpha \underset{G}{\Rightarrow} \beta$.

Note that in [92] only strongly complete systems were considered (and called complete). However, this is unnecessarily restrictive, there is no biological motivation to require that a next stage of the development is defined also for configurations of cells which can never occur in the development. Moreover, it follows from the next lemma that every complete regular global context L-system can be modified to an equivalent strongly complete regular global context L-system.

Lemma 1. For any regular global context L-system G, there effectively exists an equivalent regular global context L-system G' which is strongly complete.

Proof. Let $G = (\Sigma,\Gamma,P,C,\sigma)$ be a regular global context L-system. Let f be a finite substitution on Γ^* defined by $a \in f(k)$ if and only if there exists a rule $(B,a,\alpha) \in P$ so that $k \in B$. Let $R = f(C)$, let $R_1 = \Sigma^*-R$. Since regular languages are closed under finite substitution and complement, R and R_1 are regular languages. If $\alpha \in R$, then there exists $\beta \in \Sigma^*$ such that $\alpha \underset{G}{\Rightarrow} \beta$. If $R_1 = \phi$ then G is strongly complete.

Suppose that $R_1 \neq \phi$. Let s be a new symbol not in Γ. Let h be a homomorphism defined by $h(a) = s$ for any a in Σ. Let $G' = (\Sigma,\Gamma',P',C',\sigma)$, where $\Gamma' = \Gamma \cup \{s\}$, $C' = C \cup h(R_1)$, and $P' = P \cup \{(\{s\},a,a):a \in \Sigma\}$. From the construction of G' follows that G' is strongly complete and if $\alpha \in R$, and $\alpha \underset{G}{\Rightarrow} \beta$ for some $\beta \in \Sigma^*$, then $\alpha \underset{G'}{\Rightarrow} \beta$, and if $\alpha \in R_1$ then $\alpha \underset{G'}{\Rightarrow} \alpha$. Therefore $L(G') = L(G)$. □

Lemma 2. It is undecidable whether a regular global context L-system is complete.

Proof. We will show that for any instance of Post's Correspondence Problem [102] there exists a regular global context L-system which is complete if and only if the instance of Post's Correspondence Problem (PCP) does not have a solution.

Let $\Sigma = \{a_1,a_2,\ldots,a_n\}$ be a finite alphabet, and let A and B be two lists of strings in Σ^+ with the same number of strings in each list. Say $A = \alpha_1,\alpha_2,\ldots,\alpha_k$ and $B = \beta_1,\beta_2,\ldots,\beta_k$. Let $G = (\Sigma',\Gamma,P,C,\$)$ be a regular global L-system, where $\Sigma' = \Sigma \cup \{\$,\text{¢}\}$, $\Gamma = \{s_1,s_2,s_3,s_4\} \cup \{r_i:i = 1,2,\ldots,n\}$, $P = \{(\{s_1\},\$,\alpha_i \; \$ \; \beta_i^r):i = 1,2,\ldots,n\} \cup \{(\{r_i\},a_i,\varepsilon):i = 1,2,\ldots,n\} \cup$ $\cup \{(\{s_3\},a_i,a_i):i = 1,2,\ldots,n\} \cup \{(\{s_2\},\$,\text{¢})\} \cup \{ (\{s_4\},\text{¢},\text{¢})\}$, where β_i^r denotes the reverse of β_i, and C is denoted by $s_3^*s_1s_3^* + s_3^+s_2s_3^+ + s_3^+s_4 + s_4s_3^+ + s_3^+s_4s_3^+ +$

$+ s_3^*r_1s_4r_1s_3^* + s_3^*r_2s_4r_2s_3^* + \ldots + s_3^*r_ns_4r_ns_3^*$.

Clearly, $\$ \underset{G}{\Rightarrow}^* \alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_j}\$\beta_{i_j}^r\beta_{i_{j-1}}^r\ldots\beta_{i_1}^r \underset{G}{\Rightarrow} \alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_j}\text{¢}$ $\beta_{i_j}^r\beta_{i_{j-1}}^r\ldots\beta_{i_1}^r$ for $j \geq 1$, $i_1,i_2,\ldots,i_j$ being integers smaller or equal to k. If $\alpha a \text{¢} a\beta \in L(G)$, where $\alpha,\beta \in \Sigma^*$, $a \in \Sigma$, then $\alpha a \text{¢} a\beta \underset{G}{\Rightarrow} \alpha \text{¢} \beta$. If $\alpha a \text{¢} b\beta \in L(G)$, where $\alpha,\beta \in \Sigma^*$, $a,b \in \Sigma$ and $a \neq b$ then $\alpha a \text{¢} b\beta \underset{G}{\Rightarrow} \alpha a \text{¢} b\beta$ is the only possible derivation in G from $\alpha a \text{¢} b\beta$. Therefore $\$ \underset{G}{\Rightarrow}^* \text{¢}$ if and only if the instance of PCP has a solution. Since $s_4 \notin C$, G is complete if and only if the instance of PCP does not have a solution. Thus it is not decidable whether G is complete. □

Now we will give the definition of a rule context L-system. A rule context L-system has a finite set of context free rules, each rule having a finite number of labels. For each rule p there are restrictions on what rules might be used on the symbols adjacent to the symbol on which p is used. These restrictions are specified by a finite number of triples.

Definition 7. A rule context L-system is a 5-tuple $G = (\Sigma,\Gamma,P,C,\sigma)$, where:

(i) Σ is a finite, nonempty set of symbols, called the alphabet.

(ii) Γ is a finite, nonempty set of symbols, called the labels.

(iii) P is a finite subset of $p(\Gamma) \times \Sigma \times \Sigma^*$, called the set of rules. Rule (B,a,α) in P is usually written in the form $B:a \rightarrow \alpha$.

(iv) C is a finite subset of $\{\#,\varepsilon\}\Gamma^* \times \Gamma \times \Gamma^* \{\#,\varepsilon\}$, called the *context set*, where # is a special symbol not in Γ, called the *endmarker*.

(v) $\sigma \in \Sigma^+$, the *initial string*.

Given a rule context L-system $G = (\Sigma,\Gamma,P,C,\sigma)$, we write $\alpha \underset{G}{\Rightarrow} \beta$ for $\alpha \in \Sigma^+$, $\beta \in \Sigma^*$ if there exist $k \geq 1$, $a_1,a_2,\ldots,a_k \in \Sigma^* \beta_1,\beta_2,\ldots,\beta_k \in \Sigma^*$ and $s_1,s_2,\ldots,s_k \in \Gamma$ so that $\alpha = a_1a_2\ldots a_k$, $\beta = \beta_1\beta_2\ldots\beta_k$ and for every i, $1 \leq i \leq k$, there exist $(B_i,a_i,\beta_i) \in P$ and $m,n \geq 0$ so that $s_i \in B_i$ and $(\mathrm{Last}_m(\#s_1s_2\ldots s_{i-1})s_i, \mathrm{First}_n(s_{i+1}s_{i+2}\ldots s_k\#)) \in C$.

The language generated by a rule context L-system G is denoted by $L(G)$ and is defined to be the set $\{\alpha \in \Sigma\ \sigma \underset{G}{\Rightarrow}^* \alpha\}$.

The family of rule context L-languages will be denoted by Φ.

Example 2. Let G_2 be a rule context L-system, $G_2 = (\{a\},\{s_1,s_2,s_3,s_4\},P,C,a\}$, where $P = \{\{s_1\}:a \rightarrow a^3,\{s_2\}:a \rightarrow a,\{s_3\}:a \rightarrow a^4,\{s_4\}:a \rightarrow a^2\}$ and $C = \{(\#,s_1,\#),(\#,s_4,\#),(\#,s_1,s_2),(s_1,s_2,\#),(s_2,s_1,s_2),(s_1,s_2,s_1),(\#,s_4,s_3),$ $(s_3,s_1,\#),(s_1,s_4,s_3),(s_4,s_3,s_1),(s_3,s_1,s_4)\}$. Let α be a string in a^*. If the length of string α is divisible by 3, then according to control set C we can apply on α only rules with labels s_1,s_3,s_4 and the only string we can derive in G from α is the string $\alpha\alpha\alpha$. If the length of α is even then we can derive in G from α only the string $\alpha\alpha$. From the initial string of G_2 we can derive strings aa and aaa. Therefore $L(G_2) = \{a^{2^n}:n \geq 0\} \cup \{a^{3^n}:n \geq 0\}$.

Now, we will show that the family of rule context L-languages is equal to the family of regular global context L-systems.

Theorem 1. $\Psi = \Phi$.

Proof. Let $G_1 = (\Sigma,\Gamma,P,C,\sigma)$ be a rule context L-system. Let k,m be positive integers such that if $(\alpha,a,\beta) \in C$, then $|\alpha| < k$ and $|\beta| < m$. Let $L = \mathrm{First}(\#\Gamma^{k-1}) \cup \Gamma^k$, $R = \mathrm{Last}(\Gamma^{m-1}\#) \cup \Gamma^m$. Let A be a finite automaton, $A = (K,\Gamma,\delta,q_0,F)$, where $K = (L \times \Gamma \times R) \cup \{q_0\}$, $F = K \cap ((\Gamma \cup \{\#\})^* \times \Gamma \times \{\#\})$, and δ is defined as follows.

(i) If $(\#,p,\beta\#) \in C$, where $\beta \in \Gamma^*$ then $(\#,p,\beta\#) \in \delta(q_0,p)$.

(ii) If $(\#,p,\beta) \in C$ where $\beta \in \Gamma^*$, then $(\#,p,\beta\gamma_1\#) \in \delta(q_0,p)$ and $(\#,p,\beta\gamma_2) \in \delta(q_0,p)$, for every $\gamma_1,\gamma_2 \in \Gamma^*$ such that $\beta\gamma_1\#$, $\beta\gamma_2 \in R$.

(iii) If $(\alpha s,p,\beta) \in C$, where $\alpha \in \Gamma^* \cup \{\#\}\Gamma^*$, $\beta \in \Gamma^*$,$s,p \in \Gamma$ then $(\mathrm{Last}_k(\gamma_1\alpha s),p,\beta\gamma_2 q) \in \delta((\gamma_1\alpha,s,p\beta\gamma_2),p)$, and $(\mathrm{Last}_k(\gamma_1\alpha s),p,\beta\gamma_3\#) \in \delta((\gamma_1\alpha,s,p\beta\gamma_3\#),p)$ for any $q \in \Gamma \cup \{\#\}$,$\gamma_2,\gamma_3 \in \Gamma^*$, $\gamma_1 \in \#\Gamma^* \cup \Gamma^+$ such that $\beta\gamma_2 q,\beta\gamma_3\# \in R$ and $\gamma_1\alpha \in L$.

(iv) If $(\varepsilon,p,\beta) \in C$, where $p \in \Gamma$, $\beta \in \Gamma^*$, then
$(\mathrm{Last}_k(\gamma_1 s),p,\beta\gamma_2 q) \in \delta((\gamma_1,s,p\beta\gamma_2),p\}$ and
$(\mathrm{Last}_k(\gamma_1 s),p,\beta\gamma_3 \#) \in \delta((\gamma_1,s,p\beta\gamma_3\#),p)$ for any
$s \in \Gamma$, $q \in \Gamma \cup \{\#\}$, $\gamma_1 \in L$, γ_2, $\gamma_3 \in \Gamma^*$such that $\beta\gamma_2 q,\beta\gamma_3\# \in R$.

(v) If $(\alpha s,p,\beta\#) \in C$, where $\alpha \in \Gamma^* \cup \{\#\}\Gamma^*$, $\beta \in \Gamma^*$, $s,p \in \Gamma$ then
$(\mathrm{Last}_k(\gamma_1\alpha s),p,\beta\#) \in \delta((\gamma_1\alpha,s,p\beta\#),p)$ for any $\gamma_1 \in \#\Gamma^* \cup \Gamma^+$ such
that $\gamma_1\alpha \in L$.

(vi) If $(\varepsilon,p,\beta\#) \in C$, where $\beta \in \Gamma^*$, $p \in \Gamma$, then
$(\mathrm{Last}_k(\gamma_1 s),p,\beta\#) \in \delta((\gamma_1,s,p\beta\#),p)$ for any $\gamma_1 \in L$.

$L(A)$ is a regular language and, clearly, α is in $L(A)$ if and only if α is a string of labels of rules which can be simultaneously applied to a string in Σ^* according to context set C. Therefore, the regular global context L-system $G_2 = (\Sigma,\Gamma,P,L(A),\sigma)$ will also generate language $L(G_1)$ and thus $\Phi \subseteq \Psi$.

Now, we will show the other inclusion. Let $G = (\Sigma,\Gamma,P,Q,\sigma)$ be a regular global context L-system. Let $A = (K,\Gamma,\delta,q_0,F)$ be a finite automaton such that $\delta(q,\varepsilon) = \phi$ for any $q \in K$ and $L(A) = Q$. Let G_3 be a rule context L-system, $G_3 = (\Sigma,\Gamma_3,P_3,C_3,\sigma)$, where $\Gamma_3 = \Gamma \times K$, $P_3 = \{(A \times K,a,\alpha):(A,a,\alpha) \in P\}$, and C_3 is defined as follows.

(i) If $\delta(q_0,a) \neq \phi$, where $a \in \Gamma$, then $(\#,(a,q_0),\varepsilon) \in C_3$.

(ii) If $r \in \delta(q,a)$ and $\delta(r,b) \neq \phi$ where $a,b \in \Gamma$ and $q,r \in K$, then
$((a,q),(b,r),\varepsilon) \in C_3$.

(iii) If $r \in \delta(q,a)$ and $r \in F$, where $q \in K$, $a \in \Gamma$, then $(\varepsilon,(q,a),\#) \in C_3$.

It can be easily verified that $\alpha \underset{G_3}{\Rightarrow} \beta$ if and only if $\alpha \underset{G}{\Rightarrow} \beta$. Therefore
$L(G_3) = L(G)$. □

Let the completeness and strong completeness is defined for rule context L-systems in the same way as for regular global context L-systems. Since rule context L-systems are effectively equivalent to regular global context L-systems, Lemma 1 and Lemma 2 also hold when replacing in them a regular global context L-system by a rule context L-system.

Since any triple in the context set in a rule context L-system implicitly includes also a restriction on the adjacent symbols, it is quite obvious that the family of rule context L-languages includes context L-languages. We will show in the next theorem that this inclusion is proper.

Theorem 2. $\Omega \subsetneq \phi$.

Proof. Let $G = (\Sigma,P,\sigma)$ be a context L-system. We construct a rule context L-system $G' = (\Sigma,\Sigma,P',C,\sigma)$, where $P' = \{(\{a\},a,\beta):(\alpha_1,a,\alpha_2,\beta) \in P,\ a \in \Sigma,\ \beta \in \Sigma^*,\ \alpha_1 \in \{\#,\varepsilon\}\Sigma^*, \alpha_2 \in \Sigma^*\{\#,\varepsilon\}\}$, and $C = \{(\alpha,a,\beta):(\alpha,a,\beta,\gamma) \in P$ for some $\gamma \in \Sigma^*\}$. We have constructed the rule context L-system so that all rules for a symbol a in Σ have the same label a, and the context set of G' allows to obtain in G' exactly the same derivations as in G. Therefore L(G) = L(G'). Thus we have shown that $\Omega \subseteq \Phi$ and it remains to show that the inclusion is proper. In Example 2 the language $L = \{a^{2^n}:n \geq 0\} \cup \{a^{3^n}:n \geq 0\}$ is generated by a rule contex L-system. It has been shown in [92], that L is not in Ω. □

Now, we will give the definition of a predictive context L-system. In a predictive context L-system the use of a rule is restricted by the context of the right hand side of the rule after the simultaneous replacement of all the symbols in a string.

Definition 8. A predictive context L-system G is a 3-tuple (Σ,P,σ), where

(i) Σ is a finite, nonempty set of symbols, called the alphabet.

(ii) P is a finite subset of $\Sigma \times \{\#,\varepsilon\}\Sigma^* \times \Sigma^* \times \Sigma^*\ \{\#,\varepsilon\}$, called the set of rules, where # is a special symbol not in Σ, called the endmarker. A rule $(a,\beta_1,\alpha,\beta_2)$ in P is usually written in the form $a \rightarrow <\beta_1,\alpha,\beta_2>$. (We assume that "<" and ">" are symbols not in Σ.)

(iii) $\sigma \in \Sigma^+$, the initial string.

Given a predictive context L-system $G = (\Sigma,P,\sigma)$, we write $\alpha \Rightarrow \beta$ for $\alpha \in \Sigma^+$, $\beta \in \Sigma^*$ if there exist $k \geq 1$, $a_1,a_2,\ldots,a_k \in \Sigma$ and $\beta_1,\beta_2,\ldots,\beta_k \in \Sigma^*$ so that $\alpha = a_1a_2\ldots a_k$, $\beta = \beta_1\beta_2\ldots\beta_k$ and for every i, $1 \leq i \leq k$ there exist $m,n \geq 0$ such that

$$(a_i,\mathrm{Last}_m(\#\beta_1\beta_2\ldots\beta_{i-1}),\beta_i,\mathrm{First}_n(\beta_{i+1}\beta_{i+2}\ldots\beta_k\#)) \in P.$$

The language generated by a predictive context L-system G is denoted by L(G) and is defined to be the set $\{\alpha \in \Sigma^*:\sigma \underset{G}{\Rightarrow}^* \alpha\}$.

The family of predictive context L-languages is denoted by Π.

Example 3. Let G be the predictive context L-system ({a,b,c},P,abc), where $P = \{a \rightarrow <\varepsilon,a,bc>,\ b \rightarrow <a,b,c>,\ c \rightarrow <b,c,a>,\ c \rightarrow <ab,cabc,\#>,\ a \rightarrow <\varepsilon,aa,bb>,$ $b \rightarrow <aa,bb,\varepsilon>,\ c \rightarrow <bb,cc,\varepsilon>,\ a \rightarrow <\varepsilon,a,a>,\ b \rightarrow <b,b,\varepsilon>,\ c \rightarrow <c,c,\varepsilon>$.

Using the first four rules in P we can generate from the string abc the string $(abc)^m$, $m \geq 1$. If we decide to use a rule which would double a symbol, then, clearly, we have to double each symbol throughout the whole string $(abc)^m$.

Therefore, $(abc)^m \underset{G}{\Rightarrow} (a^2b^2c^2)^m$ and from any string of the form $(a^ib^ic^i)^m$, where $i > 1$, $m \geq 1$ only the string $(a^{i+1}b^{i+1}c^{i+1})^m$ can be generated. Thus $L(G) = \{(a^ib^ic^i)^m : i \geq 1, m \geq 1\}$.

We can define the completeness and strong completeness for predictive context L-systems in the same way as for regular global context L-systems. We can prove that it is undecidable whether a predictive context L-system is complete. However, in this case we cannot show that for every predictive context L-system it is possible to construct an equivalent strongly complete predictive context L-system. We can only show that every complete predictive context L-system can be made strongly complete.

<u>Lemma 3</u>. The language $L = \{a^{3^i 2^j} : i \geq 0, j \geq 0\}$ is not a predictive context L-language.

<u>Proof:</u> Since the proof of this lemma is very tedious, we will present it only informally.

Suppose that there exists a predictive context L-system $G = (\Sigma,P,\sigma)$ such that $L(G) = \{a^{3^i 2^j} : i \geq 0, j \geq 0\}$. Then there exists exactly one integer j, $j \geq 0$ such that $a \to \langle a^i,a^j,a^h \rangle \in P$ for some integers i,h. Since L is infinite, $j \geq 1$. To be able to generate all string a^{3^i} for $i \geq 1$, j has to be a power of three, i.e. $j = 3^p$, $p \geq 1$. But then we cannot generate in G all strings a^{2^i} for $i \geq 1$. □

<u>Theorem 3</u>. $\Pi \subsetneq \Psi$.

<u>Proof:</u> Let $G = (\Sigma,P,\sigma)$ be a predictive context L-system. Let k, m be natural numbers such that $|\alpha| < k$, $|\gamma| < m$ for any $(a,\alpha,\beta,\gamma) \in P$. Let A be a finite automaton, $A = (K,P,\delta,q_0,F)$, where $K = (\mathrm{First}(\#\Sigma^{k-1}) \cup \Sigma^k) \times (\mathrm{Last}(\Sigma^{m-1}\#) \cup \Sigma^m) \cup \cup \{q_0\}$, $F = K \cap (\Sigma \cup \{\#\})^* \times (\#)$, and δ is defined as follows.

(i) If $p = (a,\#,\beta,\gamma\#) \in P$ where $a \in \Sigma$, and $\beta,\gamma \in \Sigma^*$, then $(\mathrm{Last}_k(\#\beta),\gamma\#) \in \delta(q_0,p)$.

(ii) If $p = (a,\#,\beta,\gamma) \in P$, where $a \in \Sigma$, and $\beta,\gamma \in \Sigma^*$, then $(\mathrm{Last}_k(\#\beta),\gamma\delta_1\#) \in \delta(q_0,p)$ for any $\delta_1 \in \Sigma^*$ such that $|\delta_1| \leq m-|\gamma|-1$, and $(\mathrm{Last}(\#\beta),\gamma\delta_2) \in \delta(q_0,p)$ for any $\delta_2 \in \Sigma^*$ such that $|\delta_2| = m-|\gamma|$.

(iii) If $p = (a,\alpha,\beta,\gamma) \in P$, where $\alpha \in \Sigma^* \cup \#\Sigma^+$ and $\beta,\gamma \in \Sigma^*$, then $(\mathrm{Last}_k(\gamma_1\alpha\beta),\gamma\gamma_2\delta) \in \delta((\gamma_1\alpha,\beta\gamma\gamma_2),p)$ for any $\gamma_1 \in \Sigma^* \cup \#\Sigma^*, \gamma_2 \in \Sigma^*$, $\delta \in \Sigma^* \cup \Sigma^*\#$ such that $(\gamma_1\alpha, \beta\gamma_2) \in K$ and $|\delta| = |\beta|$, and also $(\mathrm{Last}_k(\gamma_1\alpha\beta),\gamma\gamma_2\#) \in \delta((\gamma_1\alpha,\beta\gamma\gamma_2\#),p)$ for any $\gamma_1 \in \Sigma^* \cup \#\Sigma^*$, and $\gamma_2 \in \Sigma^*$ such that $(\gamma_1\alpha,\beta\gamma\gamma_2\#) \in K$.

(iv) If $p = (a,\alpha,\beta,\gamma\#) \in P$, where $\alpha \in \Sigma^* \cup \#\Sigma^+$ and $\beta,\gamma \in \Sigma^*$, then $(\mathrm{Last}_k(\gamma_1\alpha\beta),\gamma\#) \in \delta((\gamma_1\alpha,\beta\gamma\#),p)$ for any $\gamma_1 \in \Sigma \cup \#\Sigma^*$ such that $(\gamma_1\alpha,\beta\gamma\#) \in K$.

It follows from the construction of automaton A that if $p_1p_2\ldots p_n \in L(A)$, where $p_i \in P$, $p_i = (a_i,\alpha_i,\beta_i,\gamma_i)$ for $1 \le i \le n$, then $a_1a_2\ldots a_n \overset{*}{\underset{G}{\Rightarrow}} \beta_1\beta_2\ldots\beta_n$ and vice versa. Therefore, the regular global context L-system $G' = (\Sigma,P,P',L(A),\sigma)$, where $P' = \{(\{a,\beta,\alpha,\gamma\},a,\alpha):(a,\beta,\alpha,\gamma) \in P\}$, generates also the language $L(G)$. Thus $\Pi \subseteq \Psi$.

In Example 1 we have shown that the language $L = \{a^{3^i2^j}:i \ge 0,\ j \ge 0\}$ is a regular global context L-language. However, by Lemma 3, L is not a predictive context L-language. Thus, the inclusion is proper. □

It has been shown in [92] that the family of regular languages is included in the family of context L-systems. It is easy to modify this proof to show that all regular languages containing a nonempty string are also included in the family of predictive context L-languages.

Let the family of regular languages be denoted by REGULAR.

Theorem 4. REGULAR-$\{\{\varepsilon\},\phi\} \subsetneq \Pi$.

Proof. Similar to the proof that REGULAR-$\{\{\varepsilon\},\phi\}$ is included in Ω in [92]. □

Now, we will compare the generative power of TOL-systems with that of context sensitive L-systems. The family of TOL-languages will be denoted by TOL.

Theorem 5. TOL $\underset{\not\subseteq}{\not\supseteq}$ Π.

Proof. TOL does not include all finite sets as shown in [81]. Therefore, it follows from Theorem 4 that $\Pi \not\subseteq$ TOL. We have shown in Lemma 3 that the language $L = \{a^{3^i2^j}:i \ge 0,j \ge 0\}$ is not a predictive context L-language. However, L is generated by TOL-system $G = (\{a\},\{\{a \rightarrow aa\},\{a \rightarrow aaa\},a)$. Therefore, TOL $\not\subseteq \Pi$. □

Theorem 6. TOL $\subsetneq \Psi$.

Proof. Let $G = (\Sigma,\mathcal{P},\sigma)$ be a TOL-system, where $\mathcal{P} = \{P_1,P_2,\ldots,P_n\}$. Let $G' = (\Sigma,\Gamma,P',Q,\sigma)$ be a regular global context L-system, where $\Gamma = \{s_1,s_2,\ldots,s_n\}$, Q is denoted by $s_1^+ + s_2^+ +\ldots+s_n^+$, and P' is defined as follows.

$P' = \{(A,a,\alpha):a \in \Sigma,\ \alpha \in \Sigma^*,\ (a,\alpha) \in P_i$ for some i, $1 \le i \le n$ and $A = \{s_j \in \Gamma:(a,\alpha) \in P_j\}\}$, i.e. a rule p has label s_j if and only if p is in the table P_j. Since the control set Q allows to use at one step in a derivation only rules which all are from the same table of P we have $L(G) = L(G')$. Thus TOL $\subseteq \Psi$.

It follows from Theorem 3 and Theorem 5 that the inclusion is proper. □

Lemma 4. The language $L = \{(a^n b^n c^n)^m : n,m \geq 1\}$ is not a context L-language.

Proof. We will give here only an informal proof to keep the paper short.

Suppose that there exists a context L-system $G = (\Sigma,P,\sigma)$ generating the language $L = \{(a^n b^n c^n)^m : n,m \geq 1\}$. Since G can generate all strings $a^n b^n c^n$ for $n \geq 0$, there exists exactly one integer i, $i \geq 1$ such that $<a^k,a,a^j> \to a^i$ is in P for some integer k,j. Similarly, for rules involving only symbol b and only symbol c. Therefore, there exists a constant c such that for any $n \geq c$, if $(a^n b^n c^n)^m \underset{G}{\Rightarrow}^* \alpha$ then $\alpha = (a^k b^k c^k)^m$ for some integer $k \geq n$. Thus there exists an integer $j \geq 1$ such that the following holds. There exist infinitely many strings in L of type $(a^j b^j c^j)^m$ for some integer m and $(a^j b^j c^j)^m \underset{G}{\Rightarrow} (a^{j_1} b^{j_2} c^{j_2})^{n_1}$, $(a^j b^j c^j)^m \underset{G}{\Rightarrow} (a^{j_2} b^{j_2} c^{j_2})^{m_2}$ and $m_1 \neq m_2$, $j_1 \neq j_2$. Then we can generate in G also strings not in L, which is a contradiction to $L = L(G)$. □

Since we have shown in Example 3 that the language $L = \{(a^n b^n c^n)^m : n,m \geq 1\}$ is a predictive context L-language, it is clear that context L-languages do not include all predictive context L-languages.

Theorem 7. $\Pi \not\subseteq \Omega$.

Proof. It follows directly from Lemma 4 and Example 3. □

Now, we will compare the generative power of context sensitive grammars with that of predictive context L-systems and regular global context L-systems.

Theorem 8. For each type 0 language L over alphabet T, there exists a predictive context L-system G such that $L = L(G) \cap T^*$.

Proof. Let L be generated by a type 0 grammar $G_1 = (N,T,P,S)$. Let $G = (\Sigma,P',S)$ be a predictive context L-system, where $\Sigma = T \cup N \cup \{(p,p):P \in P\} \cup \{(p,A):p \in P \text{ and } A \in N \cup T\}$, and P' is constructed as follows.

(i) If $A \to \alpha \in P$, where $A \in N$, $\alpha \in (N \cup T)^*$, then $A \to \alpha \in P'$.

(ii) If $p = A_1A_2\ldots A_n \to B_1B_2\ldots B_m \in P$, where $A_1,A_2,\ldots,A_n,B_1,B_2,\ldots,B_m \in N \cup T$, $m \geq n$, then $A_i \to <(p,A_1)B_1(p,A_2)B_2\ldots(p,A_{i-1})B_{i-1},(p,A_i)B_i, (p,A_{i+1})B_{i+1}(p,A_{i+2})B_{i+2}\ldots(p,A_{n-1})B_{n-1}(p,A_n)B_nB_{n+1}\ldots B_m(p,p)> \in P'$ for $1 \leq i \leq n-1$, and $A_n \to <(p,A_1)B_1(p,A_2)B_2\ldots(p,A_{n-1})B_{n-1}, (p,A_n)B_nB_{n+1}\ldots B_m(p,p),\varepsilon> \in P'$.

(iii) If $p = A_1A_2\ldots A_n \to B_1B_2\ldots B_m \in P$, where $A_1,A_2,\ldots,A_n$, $B_1,B_2,\ldots,B_m \in N \cup T$, $1 \leq m < n$. Then $A_i \to \langle(p,A_1)B_1(p,A_2)B_2\ldots(p,A_{i-1})B_{i-1},(p,A_i)B_i,(p,A_{i+1})B_{i+1}$ $(p,A_{i+2})B_{i+2}\ldots(p,A_m)B_m(p,A_{m+1})(p,A_{m+2})\ldots(p,A_n)\rangle \in P'$ for $1 \leq i \leq m$, and $A_i \to \langle(p,A_1)B_1(p,A_2)B_2\ldots(p,A_m)B_m(p,A_{m+1})$ $(p,A_{m+2})\ldots(p,A_{i-1}),(p,A_i),(p,A_{i+1})(p,A_{i+2})\ldots(p,A_n)\rangle \in P'$ for $m+1 \leq i \leq n$.

(iv) If $p = A_1A_2\ldots A_n \to \varepsilon$, where $A_1,A_2,\ldots,A_n \in N \cup T$, $n > 1$, then $A_i \to \langle(p,A_1)(p,A_2)\ldots(p,A_{i-1}),(p,A_i),(p,A_{i+1})(p,A_{i+2})\ldots(p,A_n)\rangle \in P'$ for $1 \leq i \leq n$.

(v) $(p,A) \to \varepsilon \in P'$, and $(p,p) \to \varepsilon \in P'$ for any $p \in P$, $A \in N \cup T$.

(vi) $A \to A \in P'$ for any $A \in N \cup T$.

It follows from the construction that if $\alpha A_1A_2\ldots A_n\beta \underset{G_1}{\Rightarrow} \alpha B_1B_2\ldots B_m\beta$, where $A_1,A_2,\ldots,A_n,B_1,B_2,\ldots B_n \in N \cup T,\alpha,\beta \in (N \cup T)^*$ using the rule $A_1A_2\ldots A_n \to B_1B_2\ldots B_m, m \geq n$, then $\alpha A_1A_2\ldots A_n\beta \underset{G}{\Rightarrow} \alpha(p,A_1)B_1(p,A_2)B_2\ldots$ $(p,A_n)B_nB_{n+1}\ldots B_m(p,p)\beta \underset{G}{\Rightarrow} \alpha B_1B_2\ldots B_m\beta$, and if $\alpha\gamma\beta \underset{G}{\Rightarrow} \alpha(p,A_1)B_1(p,A_2)B_2\ldots$ $(p,A_n)B_nB_{n+1}\ldots B_n(p,p)$, then $\gamma = A_1A_2\ldots A_n$. The same can be shown if other types of rules of G_1 are used. Therefore $S \underset{G}{\Rightarrow}^* \alpha$, where $\alpha \in (N \cup T)^*$ if and only if $S \underset{G_1}{\Rightarrow}^* \alpha$. Thus $L(G_1) = L(G) \cap T^*$. □

Let the family of context-sensitive languages be denoted by CS.

<u>Theorem 9.</u> $\Pi \underset{\not\subset}{\not\supset}$ CS.

<u>Proof</u>. Suppose that $\Pi \subseteq$ CS. Since context sensitive languages are included in recursive languages and recursive languages are closed under intersection, $L \cap T^*$ is a recursive language for any L in Π and any alphabet T. This is a contradiction to Theorem 8. Therefore, $\Pi \not\subseteq$ CS.

We have shown in Lemma 3 that the language $L = \{a^{3^i2^j} : i \geq 0, j \geq 0\}$ is not in Π. However, L is clearly a context sensitive language. Therefore, CS $\not\subset \Pi$. □

Now, we would like to compare the family of context sensitive languages to the family of regular global context L-languages. It is clear from the previous theorem and from Theorem 3 that the family of regular global context L-languages is not included in the family of context sensitive languages. To prove that the family of regular global context L-languages does not contain all context-sensitive languages we introduce the concept of exponentially dense languages.

Definition 9. Language L is called exponentially dense if there exist constants c_1 and c_2 having the following property: For any $n \geq 0$ there exists a string α in L such that $c_1 e^{(n-1)c_2} \leq |\alpha| < c_1 e^{nc_2}$.

Lemma 5. Any regular global context L-language which is infinite is exponentially dense.

Proof. Let L be an infinite, regular context L-language. Let $G = (\Sigma,\Gamma,P,C,\sigma)$ be a regular global context L-system generating L. Let $c_1 = |\sigma|$, $d_2 = \max\{|\gamma| : (A,a,\gamma) \in P$ for some $A \subset \Gamma$, $a \in \Sigma$ and $\gamma \in \Sigma^*\}$. Let $c_2 = \log d_2$. Since L is infinite, $d_2 > 1$. If $n = 0$ then,clearly, $c_1 \leq |\sigma| < c_1 e^{c_2}$. Let n be an arbitrary fixed integer, $n > 0$. Since L is infinite, there exists $\alpha \in L$ such that $|\alpha| \geq c_1 e^{nc_2}$. As $\alpha \in L$ and $|\alpha| > |\sigma|$ there exist $k > 1$ and $\beta_1,\beta_2,\ldots,\beta_k \in L$ so that $\beta_i \Rightarrow \beta_{i+1}$ for $1 \leq i \leq k-1$, $\beta_1 = \sigma$ and $\beta_k = \alpha$. Let j be an integer, $1 \leq j < k$ such that $|\beta_j| < c_1 e^{nc_2}$ and $|\beta_{j+1}| \geq c_1 e^{nc_2}$. Clearly, such integer j exists. Now we have $|\beta_j| \geq |\beta_{j+1}|/d_2 \geq c_1 e^{nc_2}/d_2 = c_1 e^{(n-1)c_2}$. □

Lemma 6. The language $\{a^{2^{2^n}} : n \geq 0\}$ is not a regular global context L-language.

Proof. The language $\{a^{2^{2^n}} : n \geq 0\}$ is not exponentially dense and therefore by Lemma 6 is not a regular global context L-language. □

Theorem 10. $CS \not\subseteq \Psi$.

Proof. By Theorems 3 and 9, Ψ is not included in CS. The language $L = \{a^{2^{2^n}} : n \geq 0\}$ is a context sensitive language, however, L is not in Ψ by Lemma 6. □