Proceedings of Grammar Systems Week 2004, edited by E. Csuhaj-Varjú and Gy. Vaszil,
MTA SZTAKI, Budapest, pages 240 - 254, 2004.

Monocultures and Homogeneous Environment in Eco-Grammar Systems*

Alica Kelemenová, Michal Tupý
Institute of Computer Science, Silesian University
Opava, Czech Republic
kelemenova@fpf.slu.cz, Michal.Tupy@seznam.cz

1 Introduction

The model of an eco-grammar system was introduced in [6] and presented in detail in [7]. It realizes an attempt to create formal specification for investigation of the interplay between the environment and agents in systems like ecosystems. Eco-grammar systems can be used to model some aspects of the behaviour of any cooperating communities of agents acting in a common dynamic environment. The model is based on the approaches and methods of formal language theory using generative framework of grammar systems [2].

Basic information on the topic can be found in overview papers [5], [11], [12], [13] and [14]. Annotated bibliography *www.sztaki.hu/mms/ecobib.html* provides a good source of information on eco-grammar systems.

The model consists of two interconnected parts. The environment, described by a string developing by L system mode, in totally parallel manner using interactionless rules, and collection of agents (components), each one described by its own string, developing by L system mode using its own set of rules. The agent locally changes the environment using its action rules. Actual state of the environment can influence development of agents and the state of each agent can influence the development of the environment by the choice of the action rules.

The original model is described in a quite general way in order to demonstrate wide possibility of the model to investigate various aspects of the behaviour of ecosystems and to present formal background for such an investigation.

To get insight to the behaviour of eco-grammar systems different special cases, characterized by various restrictions, were studied. We mention simple eco-grammar systems, for example, where agents influence the development of the environment just by their action rules (but not by the states of agents). (See [2], [4], [8], [9], [17].)

In the present paper we study eco-grammar systems, where the behaviour of the environment is really influenced by the state of agents (not only by the existence of agents and their action rules). We discuss in detail the generative power of

*Research supported by the Grant Agency of the Czech Republic grant No. 201/04/0528

monocultures and the generative power of eco-grammar systems with homogeneous environments. The results are related to the number of components of the systems.

In the Section 2 we give formal definition of the eco-grammar system and fix notations used in the paper.

Section 3 contains various examples of eco-grammar systems and some preliminary results concerning the properties of eco-grammar systems generating some typical languages of eco-grammar systems.

In Section 4 we start to study systems with identical components, called monocultures. The restriction to monocultures *with fixed number of components* restricts also the generative power of eco-grammar systems. An important result is that any language of eco-grammar system can be transformed to a language of monoculture simply by adding one word to the original language. Corresponding eco-grammar system for the monoculture have the same number of components as the original one. Still open problem remains the equality of the languages classes of monocultures and the (general) eco-grammar languages when no restriction to the number of components is considered. Number of components is important for the generative power of monocultures. Language classes with different number of components are incomparable.

In Section 5 we consider systems with homogeneous environment (systems with environment described by strings over the unary alphabet). In this case number of components in eco-grammar systems introduces full hierarchy on the corresponding language classes. Eco-grammar systems with less components are more powerful. Languages of unary eco-grammar systems can be generated by eco-grammar systems with single component.

In Section 6 the unary monocultures are treated, i.e. eco-grammar systems with identical components and unary alphabet of the environment. Unary monocultures are as powerful (generative power) as unary eco-grammar systems. This solves an open problem from Section 4 for unary monocultures. The number of components in unary monocultures introduces partial hierarchy on the corresponding language classes, depending on the fact, whether the number of components of the one of the system is a multiple of the number of components of the other system.

In all previous results context rules are used for action of the components in the environment. Consequences of the restriction to context-free action rules are discussed in Section 7. Eco-grammar systems with context-free action rules are less powerful as those with context action rules. The restriction to the context-free case leads to different results for the problems we are dealing with. Language classes of eco-grammar systems with different number of components are incomparable. Monocultures with context-free actions are less powerful as general context-free eco-grammar systems. Last result holds even if no restriction to the number of components are treated. Each context-free eco-language can be extended by one word to a language of context-free monoculture. Incomparability of language classes with different size of components is presented, too.

2 Eco-Grammar Systems: Original Model and its Special Versions

The models of eco-grammar systems are introduced in [6], [7]. An eco-grammar system Σ consists of two mutually influencing parts, an environment characterized by an $0L$ scheme $E = (V_E, P_E)$ and a collection of n agents $A_i = (V_A, P_i, \varphi_i, R_i, \psi_i)$, $1 \leq i \leq n$ characterized by inner developmental rules P_i used to develop i-th agent, action rules R_i influencing the environment locally in the chosen place. Selection functions φ_i, ψ_i determine actually active rules.

Definition 2.1 *An eco-grammar system of the degree n (EG(n) for short) is a structure* $\Sigma = (E, A_1, A_2, \ldots, A_n)$*, where*

- $E = (V_E, P_E)$ *is an 0L schema called an environment of the EG(n)*
- $A_i = (V_i, P_i, R_i, \varphi_i, \psi_i, w_i)$*, for* $1 \leq i \leq n$ *is i-th agent of EG(n), where*
  - V_i *is a finite alphabet,*
  - P_i *is a finite complete set of context-free rules over* V_i*,*
  - R_i *is a finite set of rules over* V_E*,*
  - φ_i *is a function* $V_E^* \to P(P_i)$*,*
  - ψ_i *is a function* $V_i^+ \to P(R_i)$*,*
  - w_i *is an axiom,* $w_i \in V_i^*$*.*

Definition 2.2 A configuration *(or state) of EG(n) system* Σ *is an (n+1)-tuple* $\langle v \rangle = (v_E, v_1, v_2, \ldots, v_n)$*, where* v_E *is a state of the environment and* v_i *for* $1 \leq i \leq n$ *is a state of the i-th agent. The starting configuration is an* $(n+1)$*-tuple* $\langle w \rangle = (w_0, w_1, w_2, \ldots, w_n)$*, where* w_0 *is a starting state (an axiom) of the environment and* $w_1, w_2, \ldots, w_n$ *are axioms of the agents* $A_1, A_2, \ldots, A_n$*, respectively.*

Definition 2.3 A derivation step *in the EG(n) system* $\Sigma = (E, A_1, A_2, \ldots, A_n)$ *is a binary relation* $\Longrightarrow_\Sigma$ *over* $V_E^* \times V_1^* \times V_2^* \times \ldots \times V_n^*$ *such that*
$(v_E, v_1, v_2, \ldots, v_n) \Longrightarrow_\Sigma (v'_E, v'_1, v'_2, \ldots, v'_n)$ *if and only if*

- $v_E = \alpha_0 \beta_{i_1} \alpha_1 \beta_{i_2} \ldots \alpha_{n-1} \beta_{i_n} \alpha_n$,
  $v'_E = \alpha'_0 \beta'_{i_1} \alpha'_1 \beta'_{i_2} \ldots \alpha'_{n-1} \beta'_{i_n} \alpha'_n$,
  where
  $\alpha_k \Longrightarrow_E \alpha'_k, 0 \leq k \leq n$,
  $\beta_{i_k} \to \beta'_{i_k} \in \psi_{i_k}(v_{i_k}), 1 \leq k \leq n, \{i_1, \ldots, i_n\} = \{1, 2, \ldots, n\}$.
- $v_i \Longrightarrow_{\varphi_i(v_E)} v'_i, 1 \leq i \leq n$.

We assume that all agents are active in any derivation step (expressed by $\{i_1, \ldots, i_n\} = \{1, 2, \ldots, n\}$).

In the case when it is clear which $_\Sigma$ is considered the symbol $_\Sigma$ can be omitted. A transitive closure of the relation $\Longrightarrow$ is denoted by $\Longrightarrow^+$ and a transitive and reflexive closure of the relation $\Longrightarrow$ is denote by $\Longrightarrow^*$.

A derivation is in *a canonic form*, if moreover

$$\begin{aligned} v_E &= \beta_1\beta_2\ldots\beta_n\alpha_n, \\ v'_E &= \beta'_1\beta'_2\ldots\beta'_n\alpha'_n. \end{aligned}$$

In the case when the choice of an action rule is independent on the actual state of agent A, i.e. $\psi_A(v) = R_A$ for all $v \in V_A^+$ the environment is independent on actual states of the agents and agents influence the development of the environment by the set of its action rules only. The selection function for activity of agents is universal (any action rule can be active). In this case we specify agents simply by $A = \{R_A\}$. Such an agent is called *a simple agent.* An eco-grammar system with simple agents only is called *simple eco-grammar system.*

A configuration of the simple eco-grammar system consists only of the state of the environment. A derivation step $\Longrightarrow_\Sigma$ over V_E^* of the simple eco-grammar system $\Sigma = (E, A_1, A_2, \ldots, A_n)$ has the form $v \Longrightarrow_\Sigma v'$ if and only if
$v = \alpha_0\beta_{i_1}\alpha_1\beta_{i_2}\ldots\alpha_{n-1}\beta_{i_n}\alpha_n$, and $v' = \alpha'_0\beta'_{i_1}\alpha'_1\beta'_{i_2}\ldots\alpha'_{n-1}\beta'_{i_n}\alpha'_n$, for
$\alpha_k \Longrightarrow_E \alpha'_k, 0 \le k \le n$, and $\beta_{i_k} \to \beta'_{i_k} \in R_{i_k}, 1 \le k \le n, \{i_1, \ldots, i_n\} = \{1, 2, \ldots n\}$.

Definition 2.4 A language *of an EG(n) system* $\Sigma = (E, A_1, A_2, \ldots, A_n)$ *and the initial state of the environment* w_0 *is a language of all states of the environment which can be derived from the initial configuration, i.e.*
$L(\Sigma, w_0) = \{v : (w_0, w_1, w_2, \ldots w_n) \Longrightarrow^* (v, v_1, v_2, \ldots v_n)\}$.

Definition 2.5 *An EG(n) system* Σ *is* monoculture *(MEG(n) for short) if* $A_i = A$ *for all* $1 \le i \le n$. *We shall use the denotation* $\Sigma = (E, A^n)$ *in this case.*

An EG(1) system is evidently monoculture.

Definition 2.6 *An EG(n) system* Σ *is* unary *(UEG(n)) if* $|V_E| = 1$. *Unary monoculture with n agents is denoted by MHEG(n).*

Definition 2.7 *The language class of eco-grammar systems (monoculture, unary, unary monoculture) of the degree n is denoted by* $\mathcal{L}(EG(n))$ *(*$\mathcal{L}(MEG(n))$, $\mathcal{L}(UEG(n))$, $\mathcal{L}(MUEG(n))$*).*

3 Systems with Locally Restricted Exponential Behavior

The language $L_{l,m} = \{a^{l^i+m} : i \in N\}$ is frequently used in the further parts of the paper. Its typical property is that each underlining EG system for $L_{l,m}$ has exactly one rule in the environment.

Lemma 3.1 *Let* $L_{l,m} = L(\Sigma, w_0)$ *for an EG(n) system* $\Sigma = (E, A_1, \ldots, A_n)$. *Then* $P_E = \{a \to a^{l^j}\}$ *for some* $j \ge 1$.

Proof: Assume contrary, that two different rules $a \to a^r, a \to a^s$ are in P_E and $r > s$. There are infinitely many words in $L(\Sigma, w_0)$ with at least one symbol rewritten

by the rules of the environment. Therefore infinitely many pairs of words u, v in $L(\Sigma, w_0)$ have $|u| - |v| = r - s$. But $L_{l,m}$ does not have that property.
Let $P_E = \{a \rightarrow a^s\}$ for some fixed number s. We have $s \neq 0, s \neq 1$ since $L(\Sigma, w_0)$ is finite for $P_E = \{a \rightarrow \lambda\}$ and there are infinitely many pairs of words $u, v \in L(\Sigma, w_0)$ such that $u \Longrightarrow v$, with constant $|u| - |v|$ for $P_E = \{a \rightarrow a\}$.
Assume $s = l^j + d$ for some fixed j and d, $0 \leq d < l^{j+1} - l^j$. We prove that $d = 0$.
Let $(a^k, w_1, w_2, \ldots, w_n) \Longrightarrow (a^{k'}, w'_1, w'_2, \ldots, w'_n)$ for $k = l^i + m$ and $k' = l^{i'} + m$.
Let $r \geq n$ letters are rewritten to t letters by action rules and all other letters are rewritten by the rule of the environment in that derivation step. Then

$$\begin{aligned} k' &= ks - rs + t, \\ l^{i'} + m &= (l^i + m) * s - rs + t, \end{aligned}$$

Now we assign $c' = sm - rs + t - m$. All numbers are constant (s, m) or limited (r, t) so c' is limited too.

$$\begin{aligned} l^i * s + c' &= l^{i'}, \\ l^i(l^j + d) + c' &= l^{i'}, \\ l^{i+j} + l^i * d + c' &= l^{i'}. \end{aligned}$$

We can choice i such that $l^i * d + c' > 0, l^i > max(|c'|)$ so

$$l^{i+j} + l^i * d + c' > l^{i+j}$$

and so for $l^j - l^{j-1} - 1 \geq d$ we obtain

$$l^{i+j} + l^i * d + c' \leq l^{i+j+1} - l^i + c' < l^{i+j+1}.$$

No i' exists so $d = 0$ and $P_E = \{a \rightarrow a^{l^j}\}$ for some $j \geq 1$. □

Example: Various representations of the language $L_{l,m} = \{a^{l^i+m} : i \in N\}$ by EG(m) systems are possible. They differ in the rule of the environment.

$$\begin{aligned} \Sigma_j &= (E, A_1, A^{m-1}), \\ E &= (\{a\}, \{a \rightarrow a^{l^j}\}), \\ A_1 &= (V, P, R, \varphi, \psi, Y), \\ A &= (V, P, R, \varphi, \psi, X), \\ V &= \{X, Y\}, \\ P &= \{X \rightarrow X, Y \rightarrow X\}, \\ R &= \{a \rightarrow a\} \cup \{a^{l+1} \rightarrow a^{l^i+1} : 1 \leq i \leq j\}, \\ \varphi(w) &= P, \text{ for } w \in a^* \\ \psi(Y) &= \{a^{l+1} \rightarrow a^{l^i+1} : 1 \leq i \leq j\}, \\ \psi(X) &= \{a \rightarrow a\}. \end{aligned}$$

Derivations in the system are of the form
$(a^{l+m}, Y, X, \ldots, X) \Longrightarrow (a^{l^i+m}, X, X, \ldots, X) \Longrightarrow (a^{l^i l^j+m}, X, X, \ldots, X) \Longrightarrow$ $(a^{l^i l^{2j}+m}, X, X, \ldots, X) \Longrightarrow \ldots$

In the first step, the first agent replaces $l + 1$ letters by one of the words $a^{l^1+1}, \ldots, a^{l^j+1}$ and each of the remaining $m - 1$ agents leave symbol a unchanged.

After the first step one of the words $a^{l+m}, \ldots, a^{l^j+m}$ is generated. In next steps agents leave m symbols unchanged and a^{l^i} symbols are rewritten by the rule of the environment. So $L(\Sigma, a^{l+m}) = L_{l,m}$.

Lemma 3.2 *Let $L = \{w_1, \ldots, w_n\}, n \geq 2$ and $|w_i| \leq k-1$ for all $1 \leq i \leq n$. Then L is not an EG(k) language.*

Proof: Each agent of an EG(k) system rewrites at least one symbol of the environment. Any language with at least two words produced by an EG(k) system has to have the word of the system of the length at least k, which is chosen to be the axiom. Otherwise no word can be generated from the axiom. □

4 Monocultures

Monocultures are eco-grammar systems with identical agents including their axioms. Obviously, each system with one agent is a monoculture. Monocultures with at least two agents are less powerful than eco-grammar systems with the same number of agents.

Theorem 4.1 *$\mathcal{L}(MEG(n)) \subset \mathcal{L}(EG(n))$ for $n > 1$.*

Proof: We have $\mathcal{L}(MEG(n)) \subseteq \mathcal{L}(EG(n))$ by the definition.
For $n > 1$ $L = \{a^{n-1}, a^n\} \in \mathcal{L}(EG(n)) - \mathcal{L}(MEG(n))$ and simple EG(n) system $\Sigma = ((\{a\}, \{a \to a\}), \{a \to \lambda\}, \{a \to a\}^{n-1})$ with axiom a^n generates $L(\Sigma, a^n) = L$. We prove $L \notin \mathcal{L}(MEG(n))$ by contradiction. Suppose that there is an MEG(n) system $\Sigma' = (E', A'^n), A' = (V', P', R', \varphi', \psi', w'_0)$ producing L.
The axiom of the environment is a^n since each agent rewrites in one step at least one letter. To obtain a^{n-1} one of the agents acts with the rule $a \to \lambda$. But, alternatively, the same rule $a \to \lambda$ can be used by all agents, too, and the system Σ' produces λ contradictory with the form of L. □

The situation may be different in a case, where no limitation to the number of agents is considered.

Open problem: (In)equality of classes $\mathcal{L}(EG)$ and $\mathcal{L}(MEG)$.

It is surprising that there is not too big difference between generative power of the eco-grammar systems and monocultures. Indeed, an arbitrary language of eco-grammar system can be transformed to language of monoculture by adding one special word to it.

Theorem 4.2 *Let $L \in \mathcal{L}(EG(n)) - \mathcal{L}(MEG(n))$. Then there is a word u such that $L \cup \{u\} \in \mathcal{L}(MEG(n))$.*

Proof: Let $L \in \mathcal{L}(EG(n)) - \mathcal{L}(MEG(n))$ and $L = L(\Sigma, w)$ for axiom w and for $\Sigma = (E, A_1, A_2, \ldots, A_n)$, $E = (V_E, P_E)$, $A_i = (V_i, P_i, R_i, \varphi_i, \psi_i, w_{i,0})$.

Assume $V_i \cap V_j = \emptyset$ for $i \neq j$ and $|V_E| > 1$. Let us choose any $u = u_1 u_2 \ldots u_n$ such that $u_i \in V_E^+$ and $|u|_{u_i} = 1$ for $i = 1, \ldots, n$. (In the case $|V_E| = 1$ we extend the alphabet to the binary one, in order to get an axiom u with the above property.)

We construct an MEG system $\Sigma' = (E, A^n)$, $A = (V, P, R, \varphi, \psi, a)$ such that $L' = L \cup \{u\} = L(\Sigma', u)$:

$$
\begin{aligned}
V &= V_1 \cup V_2 \cup \ldots \cup V_n \cup \{a\} \cup \{a_i : 1 \leq i \leq n\} \text{ for} \\
&\quad (V_1 \cup V_2 \cup \ldots \cup V_n) \cap (\{a\} \cup \{a_i : 1 \leq i \leq n\}) = \emptyset, \\
P &= P_1 \cup P_2 \cup \ldots \cup P_n \cup \{a \to a_i : 1 \leq i \leq n\} \cup \\
&\quad \cup \{a_i \to w_{i,0} : 1 \leq i \leq n\}, \\
R &= R_1 \cup R_2 \cup \ldots \cup R_n \cup \{u_i \to u_i : 1 \leq i \leq n\} \cup \\
&\quad \cup \{u_1 \to w\} \cup \{u_i \to \lambda : 2 \leq i \leq n\}, \\
\varphi : &\quad \varphi(u) = \{a \to a_i, a_i \to w_{i,0} : 1 \leq i \leq n\}, \\
&\quad \varphi(v) = \varphi_1(v) \cup \varphi_2(v) \cup \ldots \cup \varphi_n(v) \text{ for } v \in L, \\
\psi : &\quad \psi(a) = \{u_i \to u_i : 1 \leq i \leq n\}, \\
&\quad \psi(a_1) = \{u_1 \to w\}, \\
&\quad \psi(a_i) = \{u_i \to \lambda : 2 \leq i \leq n\}, \\
&\quad \psi(x) = \psi_i(x) \text{ for all } x \in V_i^*.
\end{aligned}
$$

The derivation in Σ' proceeds as follows:

$$(u, a, a, \ldots, a) \Longrightarrow_{\Sigma'} (u, a_1, a_2, \ldots, a_n) \Longrightarrow_{\Sigma'} (w, w_{1,0}, w_{2,0}, \ldots, w_{n,0})$$

Σ' is in starting configuration of system Σ generating language $L(\Sigma, w)$ and next steps follow derivation in Σ, therefore $L(\Sigma', u) = L(\Sigma, w) \cup \{u\}$. □

Theorem 4.3 *$\mathcal{L}(MEG(n))$ and $\mathcal{L}(MEG(m))$ are incomparable for $n \neq m$.*

Proof: Let $n > m$.

a) $L_{n,m} = \{a^{n^i+m} : i \in N\} \in \mathcal{L}(MEG(m)) - \mathcal{L}(MEG(n))$.

For MSEG(m) system $\Sigma = ((\{a\}, \{a \to a^n\}), \{a \to a\}^m)$ we have $L(\Sigma, a^{1+m}) = L_{n,m}$.

Suppose there is an MEG system Σ with $n, n > m$ agents for $L_{n,m}$. In the first derivation step n agents can use the same rule $x \to x'$ and all letters in the remaining part of the environment are rewritten to the power of n letters by Lemma 3.1. Therefore the axiom $w_0 = x^n y$ is rewritten to $w' = (x'^n y'^n)$, $|x'| * n + |y'| * n = kn$, i.e. $n^i + m = kn$ for some k and this is not valid for $n > m > 0$.

b) $L_n = \{(a^i b^i)^n : i \geq 1\} \in \mathcal{L}(MEG(n)) - \mathcal{L}(MEG(m))$. The MEG(n) system $\Sigma = (E, A^n)$ for L_n has $E = (\{a, b\}, \{a \to a, b \to b\})$, $A = \{ab \to aabb\}$ and the axiom $(ab)^n$.

To prove $L_n \notin \mathcal{L}(MEG(m))$ for $n > m$ assume contrary that there is an MEG(m) system $\Sigma' = (E', A'^m)$ and $L(\Sigma', w) = L_n$ for some axiom w. We prove that $P_{E'} = \{a \to a^i, b \to b^i\}$ for some $i \in N$.

The environment does not contain a rule with both symbols a, b on the right side. Otherwise some derived word contains more than n occurrences of subword ab or ba.

The environment contains neither rules $a \to b^i$ nor $b \to a^j$. Otherwise some derived word starts with b or ends with a.

The environment contains neither a pair of rules $a \to a^i, a \to a^j$ nor pair of rules $b \to b^i, b \to b^j$ for $i \neq j$. Otherwise both $a^i w', a^j w' \in L(\Sigma', w))$ or both $w'b^i, w'b^j \in L(\Sigma', w)$ and L_n does not have such a property.

The environment does not contain rules $a \to a^i, b \to b^j$ for $i \neq j$, otherwise the language $L(\Sigma', w)$ contains words w with $|w|_a \neq |w|_b$.This gives $E' = (\{a, b\}, \{a \to a^i, b \to b^i\})$ for some $i \geq 1$.

Let r be maximum number of symbols rewritten by agents and t be maximum number of symbols generated by agents in one derivation step. Let us consider a derivation step $(w, w_1, \ldots, w_m) \Longrightarrow (w', w'_1, \ldots, w'_m)$ for $w = (a^h b^h)^n, h > r, h > t$. The value of h guarantees that action rules $u \to v$ used to rewrite $(a^h b^h)^n$ have $u, v \in a^*b^* \cup b^*a^*$. Each agent can change some symbols in one neighboring occurrences of subwords a^h, b^h. At least one occurrence of a^h or b^h is rewritten by the rules of the environment. Assume that rules of agents are applied in the prefix $(a^h b^h)^m a^h$ of w in the discussed derivation step $((a^h b^h)^n, w_1, \ldots, w_m) \Longrightarrow (w', w'_1, \ldots, w'_m)$. Then the last occurrence of the string b^h in w is rewritten by the rules of the environment and we have $w' = (a^{h'} b^{h'})^n = v'ab^{hi}$. For $i = 1,\ h' = h,\ w' = w$ and the language $L(\Sigma', w))$ is finite. For $i > 1,\ h' = 2h$ and $(a^{h+1}b^{h+1})^n \notin L(\Sigma', w)$. Therefore $L \notin \mathcal{L}(MEG(m))$.

c) a^*a^n and $\{a^{m-1}\}$ are examples of languages both in $\mathcal{L}(MEG(m))$ and also in $\mathcal{L}(MEG(n))$ for $n > m > 0$. □

5 Unary Eco-Grammar Systems

In this section systems with homogeneous environment represented by unary alphabet are studied. We compare generative power of unary eco-grammar systems with different number of components. These language classes are ordered by inclusion relation. The power of systems increases by decreasing the number of agents.

Theorem 5.1 $\mathcal{L}(UEG(n+1)) \subset \mathcal{L}(UEG(n))$ *for* $n \geq 1$.

Proof: To prove $\mathcal{L}(UEG(n+1)) \subseteq \mathcal{L}(UEG(n))$ assume a UEG(n+1) system $\Sigma = (E, A_1, A_2, \ldots A_n, A_{n+1})$, with $A_i = (V_i, P_i, R_i, \varphi_i, \psi_i, w_{i,0})$ for $1 \leq i \leq n+1$. We construct an equivalent UEG(n) system Σ', where the last component of Σ' simulates both the behaviour of n-th and $n+1$-st component of Σ.

$\Sigma' = (E, A_1, \ldots, A_{n-1}, B)$, where $B = (V_B, P_B, R_B, \varphi_B, \psi_B, w_{B,0})$ and

$$\begin{aligned}
V_B &= V_n \cup V'_{n+1}, \\
P_B &= P_n \cup P'_{n+1}, \\
R_B &= \{\alpha_n \alpha'_{n+1} \to \beta_n \beta'_{n+1} : \alpha_n \to \beta_n \in R_n, \alpha_{n+1} \to \beta_{n+1} \in R_{n+1}\}, \\
\varphi_B &= \varphi_n \cup \varphi'_{n+1}, \\
\psi_B(w_n w'_{n+1}) &= \{a^k \to a^l : k = k_n + k_{n+1}, l = l_n + l_{n+1}, a^{k_n} \to a^{l_n} \in \psi_n(w_n), \\
&\quad a^{k_{n+1}} \to a^{l_{n+1}} \in \psi_{n+1}(w_{n+1})\}, \\
w_{B0} &= w_{n,0} w'_{n+1,0}.
\end{aligned}$$

By $V'_{n+1}, P'_{n+1}, \alpha'_{n+1}, \beta'_{n+1}, w'_{n+1,0}$ we mean primed copy of the original objects.

$L(\Sigma, w_0) = L(\Sigma', w_0)$ follows from the property:
$(w_i, w_{1,i}, \ldots, w_{n,i}, w_{n+1,i}) \Longrightarrow_\Sigma (w_{i+1}, w_{1,i+1}, \ldots, w_{n,i+1}, w_{n+1,i+1})$
if and only if
$(w_i, w_{1,i}, \ldots, w_{n,i} w'_{n+1,i}) \Longrightarrow_{\Sigma'} (w_{i+1}, w_{1,i+1}, \ldots, w_{n,i+1} w'_{n+1,i+1})$.

Σ and Σ' are unary, therefore place where agents act do not influence the resulting word. Assume that agents rewrite the prefix of the actual environmental word and that developmental rules of environment are used to rewrite its suffix. This gives identical environmental words derived by systems Σ and Σ'.

$\{a^n, a^{n-1}\} \in \mathcal{L}(UEG(n)) - \mathcal{L}(UEG(n+1))$ implies proper inclusion. □

Important consequence of the previous theorem is that unary eco-grammar systems with one component can generate all UEG languages.

Theorem 5.2 $\mathcal{L}(UEG(1)) = \mathcal{L}(UEG)$

Proof: $\mathcal{L}(UEG(n)) \subset \mathcal{L}(UEG(1))$ for each $n \geq 2$ directly follows from the Theorem 5.1 and

$\mathcal{L}(UEG) = \bigcup_{i=1}^{\infty} \mathcal{L}(UEG(n)) \subseteq \mathcal{L}(UEG(1)) \subseteq \mathcal{L}(UEG)$.
So all $\subseteq$ has to be equality, which gives the Theorem. □

6 Unary Monocultures

According the previous Sections eco-grammar systems with homogeneous environments posses hierarchy on the language classes defined by systems with different number of components, while monocultures with different number of components are incomparable with respect to the generative power. In the present Section we show that homogeneous monocultures introduce partial ordering on the language classes of eco-grammar systems with different number of components.

Theorem 6.1 *Let m, n be natural numbers $n > m > 0$.*
$\mathcal{L}(UMEG(n)) \subset \mathcal{L}(UMEG(m))$ *for m dividing n,*
$\mathcal{L}(UMEG(n))$ *and* $\mathcal{L}(UMEG(m))$ *are incomparable, otherwise.*

Proof: Let $n = cm$ for $c > 1$. We prove $\mathcal{L}(UMEG(cm)) \subseteq \mathcal{L}(UMEG(m))$. Let UMEG(cm) system $\Sigma = (E, A^{cm}), A = (V, P, R, \varphi, \psi, w_0)$ produces $L = L(\Sigma, w)$. We construct an UMEG(m) system $\Sigma' = (E, B^m),\ B = (V', P', R', \varphi', \psi', w'_o)$ such that $L(\Sigma', w) = L(\Sigma, w) = L$:

$$
\begin{aligned}
V' &= V \cup \{@\}, \\
P' &= P \cup \{@ \to @\}, \\
R' &= \{\alpha_1 \ldots \alpha_c \to \alpha'_1 \ldots \alpha'_c : \alpha_i \to \alpha'_i \in R, 1 \leq i \leq c\}, \\
\varphi'(w_E) &= \varphi(w_E) \cup \{@ \to @\}, \\
\psi'(w_1 @ w_2 @ \ldots @ w_c) &= \{\alpha_1 \alpha_2 \ldots \alpha_c \to \alpha'_1 \alpha'_2 \ldots \alpha'_c : \alpha_i \to \alpha'_i \in \psi(w_i), 1 \leq i \leq c\}, \\
w'_0 &= (w_0 @)^{c-1} w_0.
\end{aligned}
$$

Agent B simulates the behaviour of c agents A with respect to the environment, so $L(\Sigma, w) = L(\Sigma', w)$.

The above inclusion is proper. To prove it we verify $L_{n,m} = \{a^{n^i+m} : i \in N\} \in \mathcal{L}(UMEG(m)) - \mathcal{L}(UMEG(n))$.

There is a UMEG(m) system $\Sigma = ((\{a\}, \{a \to a^n\}), \{a \to a\}^m)$ such that $L(\Sigma, a^{n+m}) = L_{n,m}$.

Suppose there is an UMEG(n) system Σ for $L_{n,m}$. In the first derivation step n agents can use the same rule and all letters in the remaining part of the environment are rewritten to the power of n as in Lemma 3.1. Therefore $w_0 = x^n y$ is rewritten to $w' = (x'^n y'^n)$, $|x'| * n + |y'| * n = kn$, i.e. $n^i + m = kn$ and this is not valid for $n > m > 0$.

$$L_{m,n} = \{a^{m^i+n} : i \in N\} \in \mathcal{L}(UMEG(n)) - \mathcal{L}(UMEG(m)) \text{ for } m \text{ not dividing } n.$$

To prove $L_{m,n}$ is not in $\mathcal{L}(UMEG(m))$ analogously to the discussion above we have $m^i + n = km$. This can be satisfied just for m dividing n.

Languages a^*a^n and $\{a^{m+1}\}$ are both in $\mathcal{L}(UMEG(m))$ and $\mathcal{L}(UMEG(n))$ for $n > m > 0$. (For $L = a^*a^n$ n or less agents can guarantee n letters in the derived word, other occurrences of a can be generated by $P_E = \{a \to \lambda, a \to a, a \to aa\}$. Axiom of the system always belongs to the generated language. So eco-grammar system for the singleton contains the only word as axiom and all rules acting in the environment of the eco-grammar system are of the form $a \to a$.) □

Next theorem solves the open problem from Section 3 for unary systems.

Theorem 6.2 $\mathcal{L}(UMEG) = \mathcal{L}(UEG)$.

Proof: $\mathcal{L}(UEG) = \bigcup_{n=1}^{\infty} \mathcal{L}(UEG(n)) = \mathcal{L}(UEG(1)) = \mathcal{L}(UMEG(1)) = \bigcup_{n=1}^{\infty} \mathcal{L}(UMEG(n)) = \mathcal{L}(UMEG)$. □

7 Context-free action rules

In this section the eco-grammar systems with context-free action rules are studied. Such a restriction decreases the generative power of the corresponding eco-grammar systems.

Theorem 7.1 *$\mathcal{L}(0EG(n)) \subset \mathcal{L}(EG(n))$ for $n \geq 1$,*
$\mathcal{L}(X0EG(n)) \subset \mathcal{L}(XEG(n))$ for $n \geq 1$ and $X \in \{M, U, UM\}$.

Proof: The relation $\subseteq$ follows from the definition in all cases. To prove proper inclusions we consider the language$L_{3,2n} = \{a^{3^i+2n} : i \in N\}$.

$L_{3,2n} \in \mathcal{L}(UMEG(n))$ and $L_{3,2n} = L(\Sigma, a^{3+2n})$ for the simple eco-grammar system $\Sigma = ((\{a\}, \{a \to a^3\}), \{aa \to aa\}^n)$. To prove $L_{3,2n} \notin \mathcal{L}(0EG(n))$ assume contrary that there is an 0EG(n) system $\Sigma' = ((\{a\}, P'_E), A'_1, A'_2, \ldots, A'_n)$ and w' such that $L(\Sigma', w') = L_{3,2n}$.

By Lemma 3.1 we have $P'_E = \{a \to a^{3^k}\}$ for some k. Let n symbols are replaced by agents to at most t symbols in one derivation step. Let us discuss a derivation step $(a^{3^h+2n}, w_1, \ldots, w_n) \Longrightarrow (w', w'_1, \ldots, w'_n)$.

$$|w'| = 3^{h+k} + 3^k n + c$$

where $0 \leq c \leq t-n$ symbols are added by agents. We show that such a system is not able to generate the word $a^{3^{h+1}+2n}$. Assume that $|w'| = 3^{h+k} + 3^k n + c = 3^{h+1} + 2n$ Then we get contradiction $c = 3^h(3 - 3^k) + n * (2 - 3^k) < -n < 0 \leq c$. Therefore we conclude with $L_{3,2n} \notin \mathcal{L}(0EG(n))$. □

We compare the power of context-free monocultures with that of context-free eco-grammar systems. We obtain proper inclusion even if no restriction to the number of components is considered. (Compare with open problem in general case.)

Theorem 7.2 *$\mathcal{L}(M0EG) \subset \mathcal{L}(0EG)$ and $\mathcal{L}(UM0EG) \subset \mathcal{L}(U0EG)$.*

Proof: The relation $\subseteq$ follows from the definition in both cases. We prove $L \in \mathcal{L}(0EG) - \mathcal{L}(M0EG)$ for unary language $L = \{a^{4^{2i}} : i \geq 2\} \cup \{a^{4^{4i+1}-4} : i \geq 1\} \cup \{a^{4^{4i+3}-5} : i \geq 1\}$.

We use 0EG system $\Sigma = \{E, A_1, A_2\}$ and the axiom a^{4^4} to generate L.

$$\begin{aligned}
E &= (\{a\}, \{a \to a^4\}),\\
A_1 &= (V, P, R_1, \varphi, \psi_1, A),\\
A_2 &= (V, P, R_2, \varphi, \psi_2, A),\\
V &= \{A, B, C, D\},\\
P &= \{A \to B, B \to C, C \to D, D \to A\},\\
R_1 &= \{a \to \lambda, a \to a^{20}\},\\
R_2 &= \{a \to a^4, a \to a^3, a \to a^8\},\\
\varphi : \varphi(w) &= P,
\end{aligned}$$

$$\begin{aligned}
\psi_1 : \psi_1(A) &= \{a \to \lambda\}, & \psi_2 : \psi_2(A) &= \{a \to a^4\},\\
\psi_1(B) &= \{a \to a^{20}\}, & \psi_2(B) &= \{a \to a^4\},\\
\psi_1(C) &= \{a \to \lambda\}, & \psi_2(C) &= \{a \to a^3\},\\
\psi_1(D) &= \{a \to a^{20}\}, & \psi_2(D) &= \{a \to a^8\}.
\end{aligned}$$

Derivations in Σ for the axiom a^{4^4} are of the form

$(a^{4^4}, A, A) \Longrightarrow (a^{4^5-4}, B, B) \Longrightarrow (a^{4^6}, C, C) \Longrightarrow (a^{4^7-5}, D, D) \Longrightarrow (a^{4^8}, A, A) \Longrightarrow$
$\ldots$
$(a^{4^{4i}}, A, A) \Longrightarrow (a^{4^{4i+1}-4}, B, B) \Longrightarrow (a^{4^{4i+2}}, C, C) \Longrightarrow (a^{4^{4i+3}-5}, D, D) \Longrightarrow$
$(a^{4^{4(i+1)}}, A, A) \Longrightarrow \ldots$

Therefore $L = L(\Sigma, (a^{4^4}, A, A))$.

To prove $L \notin \mathcal{L}(M0EG)$ we proceed by contradiction. Let monoculture $\Sigma' = (E, A^n)$ generates language L. We prove following points i), ..., v) for Σ':

i) $P_E = \{a \to a^{4^l}\}$ for some $l \in N$.

To obtain an "almost" exponential growth of the language we can use only one environmental rule, i.e. $P_E = \{a \to a^{4^l}\}$ for some $l \in N$. (See Lemma 3.1)

ii) All agents use identical action rules in one derivation step.

Let i-th step be the first step where different action rules of agents are used

$(w, w_A^n) \Longrightarrow^{i-1} (w_{i-1}, w_{1,i-1}, w_{2,i-1} \ldots w_{n,i-1}) \Longrightarrow (w_i, w_{1,i}, w_{2,i} \ldots w_{n,i})$.

Suppose that the first and the second agents use different action rules. Denote w_i the word of the environment and let k be the difference between lengths of right sides of action rules used by the first and the second agent. Because all action rules

used in $i-1$ previous steps were the same and all right sides of action rules are the same, similar effect to the environmental word can be obtained after $i-1$ steps by derivations when both agents work like the first or like the second agent, respectively. Follow the $i-$th derivation step in both cases:
$(w, w_A^n) \Longrightarrow^{i-1} (w_{i-1}, w_{1,i-1}, w_{1,i-1}, w_{3,i-1} \dots w_{n,i-1}) \Longrightarrow (w_i', w_{1,i}, w_{1,i}, w_{3,i} \dots w_{n,i})$,
$(w, w_A^n) \Longrightarrow^{i-1} (w_{i-1}, w_{2,i-1}, w_{2,i-1}, w_{3,i-1} \dots w_{n,i-1}) \Longrightarrow (w_i'', w_{2,i}, w_{2,i}, w_{3,i} \dots w_{n,i})$.
We have $|w_i| - |w_i'| = k$ and $|w_i''| - |w_i| = k$, but no three-tuple of words from L fulfills that condition.

Points i) and ii) result to the deterministic behaviour of the system Σ' with respect to the development of the environment.

iii) $w_0 = a^{4^4}$.

Assume that a^{4^4} is not the axiom and follow the derivation step producing word a^{4^4}
$(a^r, w_1, w_2, w_3, \dots, w_n) \Longrightarrow (a^{4^4}, u_1, u_2, \dots u_n)$. Then $r \geq 4^5 - 4$. $P_E = \{a \to a^{4^l}\}$ so action rules of agents eliminate symbols of the environment. Action rules are context-free, i.e. $a \to \lambda$ each component eliminates at most one symbol. To produce a^{4^4} from some $a^r \in L$ we need at least $3 * 4^4 - 4$ components. This blocks the derivation of the word a^{4^4} contrary with the fact that L is infinite. It gives $w_0 = a^{4^4}$.

iv) $l = 1$ and $P_E = \{a \to a^4\}$.

Due to points i)-iii) there is only one possible derivation for the environment in Σ', namely

$$a^{4^4} \Longrightarrow^+ a^{4^5-4} \Longrightarrow^+ a^{4^6} \Longrightarrow^+ a^{4^7-5} \Longrightarrow \dots$$

In this derivation the only rule, $a \to a^{4^l}$, of the environment is used combined by action rules of at most 4^4 agents.
The second part of the above derivation gives

$$4^6 = (4^5 - 4 - n) * 4^l + n * i \geq (4^5 - 4 - n) * 4^l \geq (4^5 - 4 - 4^4) * 4^l \geq (3 * 4^4 - 4) * 4^l$$

This gives $l < 2$.

v) We determine the number of agents n of Σ'. For derivation steps producing a^{4^5-4} and a^{4^7-5} we have

$$\begin{array}{rr} (4^4 - n) * 4 + i * n = 4^5 - 4, & (4^6 - n) * 4 + i * n = 4^7 - 5, \\ n = 4/(4-i), & n = 5/(4-i), \\ 0 \leq i < 4,\ 1 \leq n \leq 4 & i = 3,\ n = 5. \end{array}$$

There is no n satisfying derived conditions. Therefore no M0EG system exists for language L. □

Similarly as in general case we can add to a language of eco-grammar system one special word to obtain a language of context-free monoculture. The construction of the eco-grammar system in the proof of Theorem 4.1 does not save the context-freeness of the action rules. We can add to the alphabet of the environment V_E new letters $u_1, u_2, \dots, u_n$ and use construction analogous to that for general case. In the proof of the next theorem we give another construction which does not increase the size of at least binary environmental alphabet. There are more than two initialization steps needed in this case.

Theorem 7.3 *Let $L \in \mathcal{L}(0EG(n))$ and $|alph(L)| \geq 2$. Then there is a word $u \in (alph(L))^*$ such that $L \cup \{u\} \in \mathcal{L}(M0EG(n))$.*

Proof: The case $L \in \mathcal{L}(M0EG(n))$ is trivial. Let $L \in \mathcal{L}(0EG(n)) - \mathcal{L}(M0EG(n))$ and $L = L(\Sigma, w)$ for axiom w and for $\Sigma = (E, A_1, A_2, \ldots, A_n)$, where $E = (V_E, P_E)$ and $A_i = (V_i, P_i, R_i, \varphi_i, \psi_i, w_{i,0})$, $0 \leq i \leq n$. We choose $u = ba^{n-1}$ for a, b in V_E and we present M0EG(n) system $\Sigma' = (E, A^n)$ for $L \cup \{ba^{n-1}\}$, where axiom is ba^{n-1} and each agent $A = (V, P, R, \varphi, \psi, a)$ is determined by:

$$
\begin{aligned}
V &= V_1 \cup V_2 \cup \ldots \cup V_n \cup \{a\} \cup \{a_{i,j} : 1 \leq i, j \leq n\} \text{ for} \\
&\quad (V_1 \cup V_2 \cup \ldots \cup V_n) \cap (\{a\} \cup \{a_{i,j} : 1 \leq i, j \leq n\}) = \emptyset, \\
P &= P_1 \cup P_2 \cup \ldots \cup P_n \cup \{a \to a_{i,1} : 1 \leq i \leq n\} \cup \\
&\quad \{a_{i,j} \to a_{i,j+1} : 1 \leq i \leq n, 1 \leq j \leq n-1\} \cup \{a_{i,n} \to w_{i,0} : 1 \leq i \leq n\}, \\
R &= R_1 \cup R_2 \cup \ldots \cup R_n \cup \{b \to b, a \to a, b \to \lambda, a \to w_0\}, \\
\varphi : \varphi(u) &= \{a \to a_{i,1} : 1 \leq i \leq n\} \cup \\
&\quad \{a_{i,j} \to a_{i,j+1} : 1 \leq i \leq n, 1 \leq j \leq n-1\} \cup \{a_{i,n} \to w_{i,0} : 1 \leq i \leq n\}, \\
\varphi(v) &= \varphi_1(v) \cup \varphi_2(v) \cup \ldots \cup \varphi_n(v) \text{ for } v \in L, \\
\psi : \psi(a) &= \{a \to a, b \to b\}, \\
\psi(a_{i,i}) &= \{b \to b\} \text{ for } 1 \leq i \leq n-1, \\
\psi(a_{i,j}) &= \{a \to a\} \text{ for } i \neq j, 1 \leq i \leq n, 1 \leq j \leq n-1, \\
\psi(a_{n,n}) &= \{a \to w_0\}, \\
\psi(a_{i,n}) &= \{b \to \lambda\} \text{ for } 1 \leq i \leq n-1, \\
\psi(x) &= \psi_i(x) \text{ for all } x \in V_i^*.
\end{aligned}
$$

We describe the behaviour of the agents in Σ'. All agents start with a. In the first step each agent gets its number as the index i in $a_{i,1}$. Symbol $a_{i,j}$ is the state of i-th agent after j steps of derivation. In the environment an agent in state $a_{i,j}$ rewrites symbol b for $i = j$, otherwise it rewrites symbol a. Two agents can have same number k, but after k steps both of the agents have to rewrite symbol b and there is only one symbol b in the environment, derivation stops. Note that the state of the environment is identical with the axiom during these steps. In a successful derivation we have one agent of each type after n steps. The last agent rewrites the symbol b to w_0 and the other agents eliminate a-s. In the same time the states of all agents are rewritten to their original starting state in Σ. Formally

$$
\begin{aligned}
&(ba^{n-1}, a^n) \Longrightarrow (ba^{n-1}, a_{1,1}, a_{2,1}, \ldots, a_{n,1}) \Longrightarrow (ba^{n-1}, a_{1,2}, a_{2,2}, \ldots, a_{n,2}) \Longrightarrow \ldots \Longrightarrow \\
&(ba^{n-1}, a_{1,n-1}, a_{2,n-1}, \ldots, a_{n,n-1}) \Longrightarrow (ba^{n-1}, a_{1,n}, a_{2,n}, \ldots, a_{n,n}) \Longrightarrow \\
&(w_0, w_{1,0}, w_{2,0}, \ldots, w_{n,0})
\end{aligned}
$$

Σ' is in starting configuration of system Σ and next steps follow derivation in Σ, therefore $L(\Sigma', ba^{n-1}) = L(\Sigma, w) \cup \{ba^{n-1}\}$. □

We continue with comparison of the generative power of monocultures with different number of components. The restriction to context-free action rules destroys the (partial) hierarchy on language classes introduced by the number of components of systems.

Theorem 7.4 *$\mathcal{L}(X0EG(n))$ and $\mathcal{L}(X0EG(m))$ are incomparable but not disjoint for $X \in \{M, U, UM\}$ and $n, m \in N, n \neq m$.*

Proof: Let $n > m$. $\{a^m, \lambda\} \in \mathcal{L}(UM0EG(m)) - \mathcal{L}(0EG(n))$ by Lemma 3.2.

It is enough to prove $L_{n+1,n+1} \in \mathcal{L}(UM0EG(n)) - \mathcal{L}(0EG(m))$ where $L_{n+1,n+1} = \{a^{(n+1)^i+n+1} : i \in N\}$. We have $L_{n+1,n+1} = L(\Sigma, a^{2n+2})$ for the simple unary M0EG(n) system $\Sigma = ((\{a\}, \{a \to a^{n+1}\}), \{a \to \lambda\}^n)$.

To verify $L_{n+1,n+1} \notin \mathcal{L}(0EG(m))$ for $n > m$ assume contrary that some 0EG(m) system $\Sigma' = ((\{a\}, P_E), A'_1, A'_2, \ldots, A'_m)$ generates $L_{n+1,n+1}$. By Lemma 3.1 we have $P'_E = \{a \to a^{(n+1)^k}\}$ for some k.

Consider a derivation step $(a^{(n+1)^j+n+1}, w_1, w_2, \ldots, w_n) \Longrightarrow (w', w'_1, w'_2, \ldots, w'_n)$ For $n > m$ at least $(n+1)^j + 2$ symbols of the environment are rewritten by the rule of the environment $a \to a^{(n+1)^k}$ to $(n+1)^{j+k} + 2(n+1)^k$ symbols. Agents can add some symbols $c \geq 0$ up to agents maximum c^A.

$(n+1)^{j+k} + 2(n+1)^k + c \leq (n+1)^{j+r} + (n+1)$ for $r \geq k$,

$c \leq (n+1)^{j+r} - (n+1)^{j+k} - 2(n+1)^k + (n+1) < 0$.

System Σ' does not generate $L_{n+1,n+1}$ and there is no U0EG(m) system for $L_{n+1,n+1}$. □

References

[1] M. H. ter Beek. Simple eco-grammar systems with prescribed teams. In: *Grammatical models of multi-agent systems*, G. Păun, A. Salomaa (eds.), Gordon and Breach, Yverdon, 1999, 113–135

[2] J. Csima. On extended simple eco-grammar systems. *Acta Cybernetica 13* (1998) 359–373

[3] J. Csima. On hybrid eco-rewriting systems. *Grammars 1* (1999) 239–253

[4] J. Csima. Two remarks on variants of simple eco-grammar systems. *Acta Cybernetica 14* (2000) 569–582

[5] E. Csuhaj-Varjú. Eco-grammar systems: Recent results and perspectives. In: *Artificial Life: Grammatical models*, ed. G. Păun, Black Sea University Press, 1995, 79–103

[6] E. Csuhaj-Varjú, J. Kelemen, A. Kelemenová, G. Păun. Eco(grammar) systems. A preview. In: *Proc. 12th European Meeting on Cybernetics and System Research, Vienna, 1994,* ed. R. Trappl, World Scientific, Singapore, 1994, 941–948

[7] E. Csuhaj-Varjú, J. Kelemen, A. Kelemenová, G. Păun. Eco-grammar systems. A grammatical framework for studying lifelike interactions. *Artificial Life 3(1)* (1997) 1–28

[8] E. Csuhaj-Varjú, A. Kelemenová. Team behavior in eco-grammar systems. *Theoretical computer science 209* (1998) 213–224

[9] E. Csuhaj-Varjú, V. Mitrana. Dynamical teams in eco-grammar systems. *Fundamenta Informaticae 44* (2000) 83–94

[10] E. Csuhaj-Varjú, G. Păun, A. Salomaa. Conditional tabled eco-grammar systems versus (E)TOL systems. *Journal of Universal Computer Science 5(1)* (1995) 252–269

[11] A. Kelemenová. Descriptional complexity of eco-grammar systems. In: *Proc. Workshop DCFS 2003, Descriptional complexity of formal systems,* E. Csuhaj-Varjú, C. Kintala, D. Wotschke, Gy. Vaszil (eds.), MTA SZTAKI Budapest, 2003, 42–52

[12] A. Kelemenová. Eco-grammar systems. In: *Formal languages and applications,* Physica Verlag, Heidelberg, 11 pp. (in print)

[13] A. Kelemenová, J. Kelemen. From colonies to eco(grammar) systems. An overview. In: *Proc. Important Results and Trends in Theoretical Computer Science, Graz, 1994.* J. Karhumäki, H. A. Maurer, G. Rozenberg (eds.), Lecture Notes in Computer Science 812, Springer Verlag, Berlin, 1994, 211–231

[14] G. Păun. Eco-grammar systems: An early survey. In: *Decentralized intelligent multi-agent systems symposium.* Cracow, 1995, 355–362

[15] P. Sosík. Eco-grammar systems, decidability and the tiling problem. In: *Proc. MFCS 98 satellite workshop Grammar systems*, ed, A. Kelemenová, Silesian University, Opava, 1998, 195–213

[16] P. Sosík. On the decidability problems of eco-grammar systems. *Journal of Automata, Languages and Combinatorics* (2000) 1 45–58

[17] D. Watjen. Remarks on simple eco-grammar systems with prescribed teams. *Journal of Automata, Languages and Combinatorics 8* (2003) 1 85–108