ROMANIAN JOURNAL OF INFORMATION SCIENCE AND TECHNOLOGY
Volume **13**, Number 3, 2010, 279–309

On Eco–Foraging Systems

Katalin Anna LÁZÁR[1], Erzsébet CSUHAJ–VARJÚ[2,3], András LŐRINCZ[1]

[1] Department of Software Technology and Methodology, Eötvös Loránd University
Pázmány Péter sétány 1/C, 1117 Budapest, Hungary
E-mail: {lazarka, lorincz}@inf.elte.hu

[2] Computer and Automation Research Institute, Hungarian Academy of Sciences
Kende u. 13–17, 1111 Budapest, Hungary

[3] Department of Algorithms and Applications, Eötvös Loránd University
Pázmány Péter sétány 1/C, 1117 Budapest, Hungary
E-mail: csuhaj@sztaki.hu

Abstract. In this paper we present a formal language theoretic approach to Internet crawlers seeking novel information on the World Wide Web, based on a modified version of eco–grammar systems, called eco–foraging systems. In our model, the grammars correspond to very simple autonomous agents and the generated language to the behaviour of the system. In fact, the agents are represented by regulated rewriting devices, which impose some constraint on the search strategy of the agent. The letters of the generated strings symbolize the web pages. We prove that if we ignore the aging of the web environment in the model, then through the simulation of certain normal form grammars, the eco–foraging systems determine the class of recursively enumerable languages. If the web pages may become obsolete, then the language family generated by unordered scattered context grammars of finite index can be obtained. The ignorance of lifetime implies that the crawlers communicating only through the environment are able to identify any computable set of the environmental states. The lifetime constraint, however, considerably decreases the efficiency of the cooperation of the agents.

Key-words: crawlers, information harvest on the World Wide Web, eco–foraging systems, unordered scattered context grammar of finite index, recursively enumerable languages.

1. Introduction

The World Wide Web is an exponentially growing and a dynamically changing information source. It demonstrates the scale–free small world property [1], [14], [25]. Owing to the scale–free small world nature of the World Wide Web to locate novel information often requires strenuous efforts, hence the need of the elaboration of efficient crawling algorithms. Different approaches exist in the literature that discuss information retrieval on the World Wide Web. Pinkerton [20] applies breadth first, exhaustive crawlers and defines the beginning of the search area by means of anchor text of links as a potential predictor. In [3], Cho et al. employ URL ordering, a principle based on the characteristics of links, when defining the decisions of crawlers. Focused/topic specific/topical crawlers aim at seeking and retrieving only the subset of the World Wide Web that pertains to a specific topic of relevance [8]. Focused crawling was first introduced by Chakrabarti et al. [2]. Conceptual knowledge concerning the topic plays a crucial part in a plethora of approaches to seeking novel information on the World Wide Web. In [15], [16] and [21], the authors propose the use of reinforcement learning methods so that the crawlers are able to extract relevant information while spidering the Web. Menzer et al. [17] study some machine learning issues in the case of topical crawlers and besides the role of exploration versus exploitation, they also examine the role of adaptation (learning and evolutionary algorithms) versus static approaches. In order to ameliorate the performance of the focused crawlers Diligenti et al. [8] as well as Pant and Srinivasan [18], [19] utilize popular classification methods such as Naive Bayes, Support Vector Machine (SVM) and Neural Network classification schemes [9], [13], which are well–established techniques in the areas of text and data mining.

In our paper, we propose a formal language theoretic approach to the behaviour of crawlers seeking novel information on the World Wide Web. The pieces of information serve as food or supplies for the crawlers, for this reason they can also be called foragers [12]. We use the words crawlers and foragers interchangeably throughout the article. We model the behaviour of the crawlers in terms of eco–grammar systems [4], [6]. We modify the mathematical construction proposed in [11]. An eco–grammar system aims at modelling the interplay between the environment and the agents in complex systems such as ecosystems [4], [6]. It intends to capture some aspects of multi–agent systems [5], [10]: the grammars correspond to very simple autonomous agents and the generated language to the behaviour of the system.

We consider a variant of eco–grammar systems, called simple eco–grammar system. Briefly, a simple eco–grammar system consists of some agents and an environment. The agents are represented by a set of context–free rules and the environment by a set of evolution rules, i.e. a complete set of context–free rules applied as in the case of 0L systems. At any moment of time, the behaviour of the system is described by the state of the environment. The environmental state is represented by a string and altered by derivation steps. At each derivation step the agents (as many as able to do it) act on the environment by applying one of their context–free rules in parallel with the environment that replaces the remaining symbols according to its rule set. The evolution of the system can be characterized by the sequences of strings obtainable from the initial string through the sequence of derivation steps.

To characterize the behaviour of the crawlers in quest of novel information, we present the notion of eco–foraging systems. Eco–foraging systems are special eco–grammar systems. The changing environment represents the knowledge space to be discovered by the crawlers, which are agents in the grammar systems theoretic model. The itinerary of the agents, or more precisely, the next piece of information to be discovered is predefined in some way. To this end, we represent the agents as special, very simple programmed grammars [7]. These agents demonstrate a remarkable behaviour, since they are able to describe the language family generated by unordered scattered context grammars of finite index and the language family of recursively enumerable languages under different assumptions, whilst the individual components can generate finite languages only.

The organization of this paper is as follows. In Section 2, we give a brief review of the mathematical definitions and theorems employed throughout our work. In Section 3, we present the formal language theoretic model of Internet crawlers in quest of novel information and demonstrate how powerful our model is. Finally, in Section 4 we summarize our results and propose some future work.

2. Formal Language Prerequisites

For the basic elements of formal language theory we refer to [7], [22], [23], [24]. For an alphabet V, we denote by V^* the set of words over V, by V^+ the set of all nonempty words, i.e. $V^+ = V^* \setminus \{\lambda\}$, where λ is the empty word. $length(x)$ denotes the length of $x \in V^*$. Let $U \subseteq V^*$ and let $|x|_U$ be the number of symbols of the string obtained through erasing the symbols that are not in U from x. For a finite set A, $card(A)$ stands for the number of elements of A. The set of natural numbers is denoted by $\mathbb{N}$ and $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$. The families of context–free, context–sensitive and recursively enumerable languages are denoted by $\mathcal{L}(\mathrm{CF})$, $\mathcal{L}(\mathrm{CS})$ and $\mathcal{L}(\mathrm{RE})$, respectively.

A *programmed grammar* with *appearance checking* is a construction $G = (N, T, S, P)$, where N, T are disjoint alphabets, $S \in N$, and P is a finite set of triplets or rules of the form $(l : A \to x, \sigma(l), \varphi(l))$, where $A \in N$, $x \in (N \cup T)^*$, $l \in Label(P), \sigma(l), \varphi(l) \subseteq Label(P)$, and $Label(P)$ is a set of labels associated with the triplets of P in a one–to–one manner. If only (N, S, P) is indicated, then we speak of a *programmed grammar scheme*.

For $(l : A \to x, \sigma(l), \varphi(l)) \in P$, we define $(\omega, l) \Rightarrow (\omega', h)$, iff either $\omega = \omega_1 A \omega_2, \omega' = \omega_1 x \omega_2, h \in \sigma(l)$, or A does not appear in $\omega, \omega = \omega'$ and $h \in \varphi(l)$, where $\sigma(l)$ is called the success and $\varphi(l)$ the failure field of the rule. The generated language is $L(G) = \{\omega \in T^* \mid (S, l_0) \Rightarrow (\omega_1, l_1) \Rightarrow \ldots \Rightarrow (\omega_m, l_m) = (\omega, l_m), l_i \in Label(P)$, for $0 \le i \le m\}$.

If $\varphi(l) = \emptyset$, then G is a programmed grammar without appearance checking.

We denote by $\mathcal{L}(\mathrm{PR}_{ac})$ and $\mathcal{L}(\mathrm{PR}^{\lambda}_{ac})$ the families of languages generated by programmed grammars in the appearance checking mode with λ–free context–free rules and with arbitrary context–free rules, respectively. If the appearance checking feature is not present, then subscript ac is omitted.

A *matrix grammar* with *appearance checking* is a construction $G = (N, T, S, M, \mathcal{F})$, where N, T are disjoint alphabets, $S \in N$, and $M = \{m_1, m_2, \ldots, m_r\}$ is a finite set of sequences, called matrices, of the form $m_i : (A_{i_1} \to x_{i_1}, \ldots, A_{i_{k_i}} \to x_{i_{k_i}})$, where $A_{i_j} \in N$, $x_{i_j} \in (N \cup T)^*$, $1 \leq i \leq r, 1 \leq j \leq k_i$, and $\mathcal{F}$ is a set of occurrences of rules in the sequences of M.

For $m_i : (A_{i_1} \to x_{i_1}, \ldots, A_{i_{k_i}} \to x_{i_{k_i}}) \in M$, $1 \leq i \leq r, k_i \geq 1$, $\omega, \omega' \in (N \cup T)^*$, we define $\omega \Rightarrow_{m_i} \omega'$, iff there are $\omega_{i_1}, \ldots, \omega_{i_{k_i+1}} \in (N \cup T)^*$ such that $\omega = \omega_{i_1}, \omega' = \omega_{i_{k_i+1}}$ and for each $i, j, 1 \leq i \leq r, 1 \leq j \leq k_i$, either $\omega_{i_j} = \omega'_{i_j} A_{i_j} \omega''_{i_j}$ and $\omega_{i_{j+1}} = \omega'_{i_j} x_{i_j} \omega''_{i_j}$, or A_{i_j} does not occur in $\omega_{i_j}, \omega_{i_j} = \omega_{i_{j+1}}$ and $A_{i_j} \to x_{i_j}$ is an element of $\mathcal{F}$.

The language generated by G is defined by $L(G) = \{\omega \in T^* \mid S \Longrightarrow_{m_{j_1}} y_1 \Longrightarrow_{m_{j_2}} y_2 \Longrightarrow_{m_{j_3}} \ldots \Longrightarrow_{m_{j_s}} \omega, 1 \leq j_i \leq r, 1 \leq i \leq s\}$.

If $\mathcal{F} = \emptyset$, then G is a matrix grammar without appearance checking. In this case the component $\mathcal{F}$ is omitted.

We denote by $\mathcal{L}(\mathrm{MAT}_{ac})$ and by $\mathcal{L}(\mathrm{MAT}_{ac}^{\lambda})$ the families of languages generated by matrix grammars in the appearance checking mode with λ–free context–free rules and with arbitrary context–free rules, respectively. When the appearance checking feature is not present, then subscript *ac* is left out.

It is known from [7] that

$$\mathcal{L}(\mathrm{CF}) \subset \mathcal{L}(\mathrm{PR}_{ac}) = \mathcal{L}(\mathrm{MAT}_{ac}) \subset \mathcal{L}(\mathrm{CS}) \text{ and}$$
$$\mathcal{L}(\mathrm{PR}_{ac}^{\lambda}) = \mathcal{L}(\mathrm{MAT}_{ac}^{\lambda}) = \mathcal{L}(\mathrm{RE}).$$

A context–free matrix grammar $G = (N, T, S, M, \mathcal{F})$ is in the (preliminary) *2–normal form* iff

$$N = \{S\} \cup N^{(1)} \cup N^{(2)} \text{ with } N^{(1)} \cap N^{(2)} = \emptyset, S \notin N^{(1)} \cup N^{(2)},$$

and M contains only matrices of the following forms:

1. $S \to AX, A \in N^{(1)}, X \in N^{(2)}$,
2. $(A \to \beta, X \to Y), A \in N^{(1)}, \beta \in (N^{(1)} \cup T)^*, X, Y \in N^{(2)}$, and
3. $(A \to \beta, X \to \lambda), A \in N^{(1)}, \beta \in (N^{(1)} \cup T)^*, X \in N^{(2)}$.

Moreover, the set $\mathcal{F}$ contains only rules of the form $A \to \beta$ in the matrices of types (2) and (3).

For each context–free matrix grammar G an equivalent context–free matrix grammar G' can be constructed in the (preliminary) 2–normal form [7].

A 0L system (an *interactionless Lindenmayer system*) is a triplet $G = (V, \omega, P)$, where V is an alphabet, $\omega \in V^+$ and P is a finite set of rewriting rules such that for each $a \in V$ there is a rule $a \to x$, $x \in V^*$, in P (we say that P is *complete*). For $z_1, z_2 \in V^*$, we write $z_1 \Longrightarrow z_2$ (with respect to G, if it is necessary, denoted by $\Longrightarrow_G$), if $z_1 = a_1 a_2 \ldots a_r, z_2 = x_1 x_2 \ldots x_r$, for $a_i \to x_i$ in P, $1 \leq i \leq r$. This form of derivation is called a 0L rewriting. The language generated by G is $L(G) = \{z \in V^* \mid \omega \Longrightarrow^* z\}$, where $\Longrightarrow^*$ is the reflexive and transitive closure of $\Longrightarrow$.

A T0L system is a construct $G = (V, \omega, P_1, \ldots, P_n)$, $n \geq 1$, where each $G_i = (V, \omega, P_i)$, $1 \leq i \leq n$, is a 0L system. The language that is generated is $L(G) = \{z \in V^* \mid \omega \Longrightarrow_{G_{i_1}} \omega_1 \Longrightarrow_{G_{i_2}} \ldots \Longrightarrow_{G_{i_m}} \omega_m = z, 1 \leq i_j \leq n, 1 \leq j \leq m\}$.

The families of languages generated by 0L and T0L systems are denoted by $\mathcal{L}(0L)$ and $\mathcal{L}(T0L)$, respectively.

Let G be a grammar of arbitrary type and let N, T and S be its nonterminal alphabet, terminal alphabet and start symbol, respectively. For a derivation $D: S = \omega_1 \Rightarrow \omega_2 \Rightarrow \ldots \Rightarrow \omega_r = \omega \in T^*$ according to G, we set $Ind(D, G) = max\{|\omega_i|_N \mid 1 \leq i \leq r\}$, and for $\omega \in T^*$, we define $Ind(\omega, G) = min\{Ind(D, G) \mid D$ is a derivation for ω in $G\}$. The *index of grammar G* is defined as $Ind(G) = sup\{Ind(\omega, G) \mid \omega \in L(G)\}$. For a language L in the family $\mathcal{L}(X)$ of languages generated by grammars of some type X, we define $Ind_X(L) = inf\{Ind(G) \mid L(G) = L, G$ is of type $X\}$. If no confusion arises, we write $Ind(L)$ instead of $Ind_X(L)$. For a family $\mathcal{L}(X)$, we set $\mathcal{L}_n(X) = \{L \mid L \in \mathcal{L}(X)$ and $Ind_X(L) \leq n\}, n \geq 1$, and $\mathcal{L}_{fin}(X) = \bigcup_{n \geq 1} \mathcal{L}_n(X)$.

An *unordered scattered context grammar* is a construct $G = (N, T, S, P)$, where N, T are disjoint alphabets, $S \in N$, and $P = \{p_1, p_2, \ldots, p_r\}$ is a finite set of sequences of the form $p_i : (A_{i_1} \rightarrow x_{i_1}, \ldots, A_{i_{k_i}} \rightarrow x_{i_{k_i}})$, where $A_{i_j} \in N$, $x_{i_j} \in (N \cup T)^*$, $1 \leq i \leq r, 1 \leq j \leq k_i$. We say that ω directly derives ω', written as $\omega \Rightarrow \omega'$, iff for some i, $1 \leq i \leq r$, and for some permutation π of $\{1, \ldots, k_i\}$, $\omega = \omega_1 A_{i,\pi(j_1)} \omega_2 A_{i,\pi(j_2)} \ldots \omega_n A_{i,\pi(j_m)} \omega_{m+1}$, $\omega_j \in (N \cup T)^*$, for $1 \leq j \leq m+1$, and $\omega' = \omega_1 x_{i,\pi(j_1)} \omega_2 x_{i,\pi(j_2)} \ldots \omega_n x_{i,\pi(j_m)} \omega_{m+1}$. The generated language is $L(G) = \{x \in T^* \mid S \Longrightarrow^* x\}$.

We denote by $\mathcal{L}(USC)$ and $\mathcal{L}_{fin}(USC)$ the language families generated by unordered scattered context grammars and by unordered scattered context grammars of finite index, respectively.

3. Internet Crawlers: Formal Definitions

In this section we introduce the notion of eco–foraging systems (FEG systems) to model the behaviour of Internet crawlers in quest of novel information. Whilst harvesting information on the web these crawlers compete as well as cooperate/collaborate with each other. Eco–foraging systems have two main components: the web environment and the agents. We focus on eco–foraging systems in which the components are represented as programmed grammars, or more precisely, as programmed grammar schemes.

3.1. Eco–Foraging Systems

First, we deal with eco–foraging systems that model the case when no lifetime is associated with the web pages, i.e. we ignore that during web crawling some pages may become obsolete.

Now we define the web environment (the environment in eco–grammar systems). The web environment represents the continuously changing World Wide Web domain.

Definition 1. *The web environment with n foragers*, $n \geq 1$, is a construction

$$E = (V_E, T'_E, \mathcal{P}_E)$$

such that

- V_E is a finite alphabet, $V_E = V_M \cup T'_E \cup V_N \cup \bar{V}_N$, with $V_N = \bigcup_{i=1}^{n} N_i$ and $\bar{V}_N = \bigcup_{i=1}^{n} N_i^{(i)}$, where
  - V_M is a finite set,
  - $N_i = \{X_{i,1}, \ldots, X_{i,s_i}\}, N_i^{(i)} = \{X_{i,1}^{(i)}, \ldots, X_{i,s_i}^{(i)}\}$, $1 \leq s_i$, $1 \leq i \leq n$, are finite alphabets,
  - $T_E = \bigcup_{j=1}^{k} N_{i_j}$, and for some k, $1 \leq k \leq n$, $\{j_1, \ldots, j_k\} \subseteq \{1, \ldots, n\}$,
  - $T'_E = \{Z' \mid Z \in T_E\}$,
  - V_M, T'_E, V_N, and $\bar{V}_N$, are pairwise disjoint sets,
- $\mathcal{P}_E = \{P_{E_1}, \ldots, P_{E_r}\}$, where P_{E_q}, $1 \leq q \leq r$, is a finite set of rules of the following forms:
  - $Y \to \alpha$, where $Y \in V_N, \alpha \in V_N^*$,
  - $Z^{(i)} \to \beta$, where $Z^{(i)} \in N_i^{(i)}$, $1 \leq i \leq n$, and $\beta \in V_N^* \cup V_N^* Z^{(i)} V_N^*$,
  - $Z^{(j)} \to Z'$, $Z' \to Z'$, where $Z^{(j)} \in N_j^{(j)}$, $1 \leq j \leq n$, $N_j \subseteq T_E$, $Z' \in T'_E$ and $Z \in T_E$,
  - $U \to \gamma$, where $U \in V_M$ and $\gamma \in V_N^* V_M^* V_N^*$.

  Moreover, any rule set in $\mathcal{P}_E$ is complete, i.e. for any $c \in V_E$, there is at least one rule in any P_{E_q}, $1 \leq q \leq r$.

In Definition 1, V_E is the alphabet of the web environment, i.e. the web pages that can be altered through the joint action of the foragers and the web environment. V_E consists of the union of all alphabets N_i and $N_i^{(i)}$, $1 \leq i \leq n$, T'_E and V_M. The elements of N_i correspond to web pages that can be identified, while those of $N_i^{(i)}$ to web pages that were actually visited by the i–th forager. T_E represents the web pages that should be visited by the foragers, T'_E describes that those web pages that should be visited were really recognized by the foragers and reinforced later by the environment. The symbols from V_M characterize how the web environment works, i.e. they cannot be rewritten by any of the agents. $\mathcal{P}_E$ is the set of all rule sets P_{E_q}, $1 \leq q \leq r$, where each P_{E_q} is a set of rules (the so–called evolution rules): it describes the update of a non–visited web page, of a visited one and some other kinds of rewritings, respectively. In particular, rules of the form $Y \to \alpha$, where $Y \in V_N$, $\alpha \in V_N^*$, correspond to the update (insertion of new web page(s) into the environment, the deletion or the substitution of some part of the environmental state) of a non–visited web page, rules of the form $Z^{(i)} \to \beta$, where $Z^{(i)} \in N_i^{(i)}$, $1 \leq i \leq n$, and $\beta \in V_N^* \cup V_N^* Z^{(i)} V_N^*$, express that the actually visited web page has been deleted or left unaltered and at the same time some new web pages may have been inserted, rules of the form $Z^{(i)} \to Z'$, $Z' \to Z'$, where $Z^{(i)} \in N_i^{(i)}$, $1 \leq i \leq n$, $N_i \subseteq T_E$, $Z' \in T'_E$ and $Z \in T_E$, represent that the web pages visited by the foragers are reinforced by the environment, rules of the form $U \to \gamma$, where $U \in V_M$ and $\gamma \in V_N^* V_M^* V_N^*$, describe that symbols from the finite set V_M have been rewritten and/or some new web pages have been inserted.

We impose some constraint on the rules of the agents of the eco–grammar systems to describe the search strategy of these agents.

Definition 2. *A programmed eco–foraging system with appearance checking (an $FEG_{PR_{ac}}$ system) of degree n, $n \geq 1$,* is a construction

$$\Gamma = (E, A_1, \ldots, A_n, c_{init})$$

such that

- $E = (V_E, T'_E, \mathcal{P}_E)$ is the web environment (see Definition 1),
- $A_i = (N_i \cup N_i^{(i)}, S_i, R_i)$, $1 \leq i \leq n$, is the i–th forager, a programmed grammar scheme with appearance checking, where
  - $N_i \cup N_i^{(i)}$ is the nonterminal alphabet of the i–th forager (see Definition 1),
  - $S_i \in N_i$ is the start symbol of the i–th forager,
  - R_i is a finite set of triplets of the following forms:
    - $(l_{i,1} : S_i \to S_i^{(i)}, \sigma_i(l_{i,1}), \psi_i(l_{i,1}))$, $\sigma_i(l_{i,1}) \subseteq \{l_{i,1}, \ldots, l_{i,s_i}\}$, $\psi_i(l_{i,1}) = \{l_{i,1}\}$, is called the initial rule of the i–th forager,
    - $(l_{i,k} : X_{i,k} \to X_{i,k}^{(i)}, \sigma_i(l_{i,k}), \psi_i(l_{i,k}))$, $X_{i,k} \in N_i \setminus \{S_i\}, X_{i,k}^{(i)} \in N_i^{(i)} \setminus \{S_i^{(i)}\}$, $2 \leq k \leq s_i$, with $\sigma_i(l_{i,k}) \subseteq \{l_{i,1}, \ldots, l_{i,s_i}\}, \psi_i(l_{i,k}) \subseteq \{h_{i,2}, \ldots, h_{i,s_i}\}$, or
    - $(h_{i,k} : X_{i,k}^{(i)} \to X_{i,k}^{(i)}, \sigma_i(h_{i,k}), \psi_i(h_{i,k}))$, $X_{i,k}^{(i)} \in N_i^{(i)} \setminus \{S_i^{(i)}\}$, $2 \leq k \leq s_i$, with $\sigma_i(h_{i,k}) \subseteq \{l_{i,1}, \ldots, l_{i,s_i}\}, \psi_i(h_{i,k}) \subseteq \{h_{i,2}, \ldots, h_{i,s_i}\}$, where
      - $Label(R_i) = \{l_{i,1}, \ldots, l_{i,s_i}, h_{i,2}, \ldots, h_{i,s_i}\}$ is the set of labels of the rules in R_i.
- $c_{init} = (l_{1,1}, \ldots, l_{n,1}; \omega_{init})$, where $l_{i,1}$ is the label of the initial rule of the i–th forager, $1 \leq i \leq n$, and $\omega_{init} = z_0 S_{j_1} z_1 \ldots z_{k-1} S_{j_k} z_k$, $S_{j_h} \in N_{j_h}, z_l \in V_E^*, 1 \leq h \leq k, 0 \leq l \leq k, \{j_1, \ldots, j_k\} \subseteq \{1, \ldots, n\}$, is called the initial configuration of Γ. The string ω_{init} is called the initial state of the web environment of Γ or the initial environmental state.

In Definition 2, the agents or foragers are special programmed grammar schemes with appearance checking. $S_i \in N_i$ is the first web page that the i–th agent has to visit. The agents have two types of rules except for the initial step. $S_i \to S_i^{(i)}$ is the initial rule of the i–th agent. Not until the forager has visited the first web page, will it be able to jump to any of its subsequent rules. At subsequent steps, the rules of the i–th agent have the forms $X_{i,k} \to X^{(i)}_{i,k}$, $X_{i,k} \in N_i \setminus \{S_i\}, X^{(i)}_{i,k} \in N^{(i)}_i \setminus \{S^{(i)}_i\}$, or $X^{(i)}_{i,k} \to X^{(i)}_{i,k}$, $X^{(i)}_{i,k} \in N^{(i)}_i \setminus \{S^{(i)}_i\}$, $1 \leq i \leq n$, $2 \leq k \leq s_i$. Rules $X_{i,k} \to X^{(i)}_{i,k}$, $X_{i,k} \in N_i \setminus \{S_i\}, X^{(i)}_{i,k} \in N^{(i)}_i \setminus \{S^{(i)}_i\}$, $1 \leq i \leq n$, $2 \leq k \leq s_i$, describe that the i–th agent tries to visit a not yet discovered web page. Rules $X^{(i)}_{i,k} \to X^{(i)}_{i,k}$, $X^{(i)}_{i,k} \in N^{(i)}_i \setminus \{S^{(i)}_i\}$, $1 \leq i \leq n$, $2 \leq k \leq s_i$, on the other hand, express that the i–th agent goes to a web page that it has discovered previously. As the initial state of the web environment, ω_{init} indicates, initially, we do not suppose that every agent is able to commence its work. When the agents start their work, they have to apply their initial rules.

In the sequel, we define the way in which eco–foraging systems work.

Definition 3. Let $\Gamma = (E, A_1, \ldots, A_n, c_{init})$ be an $\mathrm{FEG}_{PR_{ac}}$ system of degree n, $n \geq 1$. An $(n+1)$–tuple $c = (k_1, \ldots, k_n; \omega_E)$, where $k_i \in \mathit{Label}(R_i)$, $1 \leq i \leq n$, $\omega_E \in V^*_E$, is called *a configuration of* Γ. ω_E is the state of the web environment of Γ in configuration c or the environmental state in configuration c.

Definition 4. Let $\Gamma = (E, A_1, \ldots, A_n, c_{init})$ be an $\mathrm{FEG}_{PR_{ac}}$ system of degree n, $n \geq 1$ (see Definition 1), and let $c_1 = (k_1, \ldots, k_n; \omega_E)$ and $c_2 = (k'_1, \ldots, k'_n; \omega'_E)$ be two configurations of Γ. We say that c_1 *directly derives* c_2 *in* Γ, written as $c_1 \Longrightarrow_\Gamma c_2$, if the following conditions hold:

1. $\omega_E = u_1\alpha_{i_1}u_2 \ldots u_r\alpha_{i_r}u_{r+1}$ and $\omega''_E = u_1\beta_{i_1}u_2 \ldots u_r\beta_{i_r}u_{r+1}$,

   where $\{i_1, \ldots, i_r\} \subseteq \{1, \ldots, n\}$, $\alpha_{i_j} \in N_j \cup N^{(j)}_j, \beta_{i_j} \in N^{(j)}_j$, $1 \leq j \leq r$, $u_h \in V^*_E$, $1 \leq h \leq r+1$,
2. $(k_{i_j} : \alpha_{i_j} \to \beta_{i_j}, \sigma(k_{i_j}), \psi(k_{i_j})) \in R_{i_j}$ and $k'_{i_j} \in \sigma(k_{i_j})$, $1 \leq j \leq r$,
3. there is no $m \in \{1, \ldots, n\} \setminus \{i_1, \ldots, i_r\}$ such that $(k_m : \alpha_{i_m} \to \beta_{i_m}, \sigma(k_m), \psi(k_m)) \in R_m$ can be applied to $u_1u_2 \ldots u_{r+1}$,
4. $k'_m \in \psi(k_m)$ for $m \in \{1, \ldots, n\} \setminus \{i_1, \ldots, i_r\}$,
5. $\omega'_E = v_1\beta_{i_1}v_2 \ldots v_r\beta_{i_r}v_{r+1}$, where $u_1 \ldots u_{r+1} \Longrightarrow v_1 \ldots v_{r+1}$ is a 0L rewriting according to some P_{E_q}, $1 \leq q \leq r$, $P_{E_q} \in \mathcal{P}_E$.

In Definition 4, the programmed grammar schemes determine the next rule to be applied on the basis of the previous one(s). If the forager has managed to identify a web page, then it will try to search for a novel one. If the attempt of the forager is not successful and it has not yet commenced its work, then it will try to visit its first web page again. If the forager fails to discover a web page different from the initial one, then it will go to a web page that it has discovered previously. If the crawler has managed to identify the previously discovered web page, then it will go to a not yet visited page, otherwise to a visited one.

The next state of the web environment is determined both by the action rules of the foragers and the set of rules of the web environment. As the reader may observe, the evolution rules of the environment are applied in the same manner as a 0L system.

If the environment has more than one set of evolution rules, then it behaves like a T0L system. The actions of the foragers have priority over the evolution of the web environment. The foragers have to perform their actions simultaneously.

The transitive (and reflexive) closure of $\Longrightarrow_\Gamma$ is denoted by $\Longrightarrow_\Gamma^+$ ($\Longrightarrow_\Gamma^*$).

Definition 5. *The language generated by an* $FEG_{PR_{ac}}$ *system* $\Gamma = (E, A_1, \ldots, A_n, c_{init})$ is defined by $L(\Gamma) = \{u \mid c_{init} = (l_{1,1}, \ldots, l_{n,1}; \omega) \Longrightarrow_\Gamma^* (k_1, \ldots, k_n; u), u \in {T'}_E^*\}$.

If no confusion arises, then subscript Γ can be omitted.

The language family generated by $\mathrm{FEG}_{PR_{ac}}$ systems is denoted by $\mathcal{L}(\mathrm{FEG}_{PR_{ac}})$.

Let us illustrate how programmed eco–foraging systems work through an example. This example demonstrates that although the agents working separately are able to recognize finite languages only, their cooperation leads to complex behaviour.

Example 1. Let $L_1 = \{{A'}^n {B'}^n {C'}^n \mid n \geq 1\}$. The programmed eco–foraging system Γ with appearance checking that generates L, i.e. $L = L(\Gamma)$, is as follows:

$$\Gamma = (E, A_1, A_2, A_3, c_{init})$$

such that

- $E = (V_E, T'_E, \mathcal{P}_E)$ is the web environment, where
  - $V_E = \{A, B, C\} \cup \{A^{(1)}, B^{(2)}, C^{(3)}\} \cup \{A', B', C'\}$,
  - $T'_E = \{A', B', C'\}$,
  - $\mathcal{P}_E = \{P_{E_1}, P_{E_2}\}$, where
    - $P_{E_1} = \{A \to A, B \to B, C \to C, A^{(1)} \to A', B^{(2)} \to B', C^{(3)} \to C', A' \to A', B' \to B', C' \to C'\}$,
    - $P_{E_2} = \{A \to A, B \to B, C \to C, A^{(1)} \to AA^{(1)}, B^{(2)} \to BB^{(2)}, C^{(3)} \to CC^{(3)}, A' \to A', B' \to B', C' \to C'\}$,
- $A_i = (N_i \cup N_i^{(i)}, S_i, R_i)$, $i = 1, 2, 3$, is the i–th forager, where
  - $N_1 = \{A\}, N_1^{(1)} = \{A^{(1)}\}, N_2 = \{B\}, N_2^{(2)} = \{B^{(2)}\}, N_3 = \{C\}, N_3^{(3)} = \{C^{(3)}\}$,
  - $S_1 = A, S_2 = B, S_3 = C$,
  - the triplets of R_i, $1 \leq i \leq 3$, are of the following forms:
    - $(l_i : X_i \to X_i^{(i)}, \sigma_i(l_i), \psi_i(l_i))$, with $\sigma_i(l_i) = \{l_i\}, \psi_i(l_i) = \{l_i\}$, $X_i \in N_i, X_i^{(i)} \in N_i^{(i)}$,
      - · $Label(R_i) = \{l_i\}$ and $\sigma_i, \psi_i : Label(R_i) \to 2^{Label(R_i)}, i = 1, 2, 3$,
- $c_{init} = (l_1, l_2, l_3; \omega_{init})$, where $\omega_{init} = ABC$.

The derivation is as follows: $ABC \Longrightarrow_\Gamma A^{(1)}B^{(2)}C^{(3)}$, where we have two possibilities to continue. If we apply environmental table P_{E_1}, then we will obtain: $A^{(1)}B^{(2)}C^{(3)} \Longrightarrow_\Gamma A'B'C'$, which is a terminal string. If we use P_{E_2}, then we will receive: $A^{(1)}B^{(2)}C^{(3)} \Longrightarrow_\Gamma AA^{(1)}BB^{(2)}CC^{(3)}$. Continuing the derivation with $AA^{(1)}BB^{(2)}CC^{(3)}$, we can employ either environmental table P_{E_1} or table P_{E_2}. Let us assume that we will use environmental table P_{E_1} (the continuation of the derivation for table P_{E_2} may be done analogously, thus it is left to the reader). As a result of the utilization of P_{E_1}, we will attain: $AA^{(1)}BB^{(2)}CC^{(3)} \Longrightarrow_\Gamma A^{(1)}A'B^{(2)}B'C^{(3)}C' \Longrightarrow_\Gamma A'A'B'B'C'C'$, which is a terminal string. We applied again environmental table P_{E_1} at the last step. The derivation could have been continued in a different way, if we had employed environmental table P_{E_2} at the last step.

Observe that L_1 is a context–sensitive language, but it is not context–free.

3.2. The Power of Eco–Foraging Systems

In the sequel, we demonstrate that the class of recursively enumerable languages is exactly the same as the class of languages generated by programmed eco–foraging systems with appearance checking. It signifies that the foragers communicating only through the environment are able to identify any computable set of the environmental states. The following theorem holds:

Theorem 1. $\mathcal{L}(\mathrm{RE}) = \mathcal{L}(\mathrm{FEG}_{PR_{ac}})$.

Proof

We only prove that $\mathcal{L}(\mathrm{RE}) \subseteq \mathcal{L}(\mathrm{FEG}_{PR_{ac}})$, the reverse inclusion can be shown by using standard techniques. Let us assume that $L \subseteq T'^*$, $L \in \mathcal{L}(\mathrm{RE})$. Let us suppose that L is generated by $G = (N, T', S, M, \mathcal{F})$, with $M = (m_1, \dots, m_n)$, where G is a matrix grammar in the (preliminary) 2–normal form. Let $T = \{b_j \mid 1 \leq j \leq s\}$, $T' = \{b'_j \mid b_j \in T, 1 \leq j \leq s\}$. We define the following homomorphism: $h : T' \cup N \to T \cup N$, where $h(b') = b$, for $b' \in T'$, $b \in T$ and $h(B) = B$, for $B \in N$. To prove the statement, we construct a programmed eco–foraging system Γ with appearance checking for G such that $L(G) = L(\Gamma)$ holds. The idea is that we simulate the derivations in G by derivations in Γ.

The programmed eco–foraging system with appearance checking, able to simulate the matrix grammar is as follows:

$$\Gamma = (E, A_1, \dots, A_n, A_{n+1}, \dots, A_{n+s}, c_{init}),$$

where

- E is the web environment,
- $A_1, \dots, A_n$, are the foragers that simulate the matrices of the matrix grammar G,
- $A_{n+1}, \dots, A_{n+s}$ are special foragers that check whether the generated string is a terminal one or not according to G (in the first case, the derivation is correct, whereas in the second case, the string produced is not in the language generated by the matrix grammar), and

– c_{init} is the initial configuration.

To help the reader in following the simulation, we will denote the nonterminal letters of agents $A_{n+1}, \ldots, A_{n+s}$ by small letters. This change does not impose any restriction on the corresponding definitions.

Now we define the components of Γ. Let

$$\begin{aligned} V_E = & N \cup T \cup \{B^{(p)} \mid B \in N, 1 \leq p \leq n\} \cup \{b_j^{(n+j)}, b_j' \mid b_j \in T, 1 \leq j \leq s\} \cup \\ & \{Z_k, Z_k^{(k)} \mid 1 \leq k \leq n\} \cup \{Z_0, Z_{n+1}, Z_{n+2}\} \cup \{Z_0^{(k)}, Z_{n+1}^{(k)}, Z_{n+2}^{(k)} \mid 1 \leq k \leq n\} \cup \\ & \{C, Z_{fin}, Z_{fin}^{(fin)}, F\} \cup \{C^{i_j} \mid 1 \leq i \leq n, 1 \leq j \leq 6\}. \end{aligned}$$

The alphabet of the web environment contains all letters of the alphabet of G and other symbols that assist the simulation. These are the marker symbols $Z_0, Z_1, \ldots, Z_n, Z_{n+1}, Z_{n+2}, Z_{fin}$ and C, their indexed versions, and the trap symbol F. The trap symbol cannot be removed form the sentential form.

The idea behind the simulation is as follows: we suppose that ω' is a sentential form in G and that the corresponding word generated by Γ has the form $CZ_kZ_0Z_{n+1}Z_{n+2}\omega, 1 \leq k \leq n$, where $h(\omega') = \omega$, or its indexed version, which indicates the fact whether a given forager is active or not. The axiom is of the form $CZ_0Z_{n+1}Z_{n+2}S$, where S is the start symbol of the matrix grammar to be simulated. Both foragers $A_i, 1 \leq i \leq n$, and the web environment can perform a rule on $Z_0, Z_1, \ldots, Z_n, Z_{n+1}, Z_{n+2}$, but only the web environment is allowed to rewrite C and Z_{fin}. Forager A_i may perform a rule only on $Z_i, 1 \leq i \leq n$, Z_0, Z_{n+1} and Z_{n+2}. Symbols $Z_0, Z_1, \ldots, Z_n, Z_{n+1}, Z_{n+2}, Z_{fin}$ and C make it possible that only the forager that simulates a matrix of G or those foragers that check whether a symbol corresponds to a letter from T', and the web environment can change the environmental word at the same time at any step of the derivation.

In the following, we present the definition of the foragers and the environmental tables and detail their roles in the simulation.

When designing the rules of A_i, we have to distinguish two cases, depending on the fact whether m_i is with or without appearance checking, if we want to simulate the matrices of the forms $(A \to \alpha', X \to Y)$ or $(A \to \alpha', X \to \lambda)$. First, we will deal with matrices of the form $(A \to \alpha', X \to Y)$ (the second case can be treated in a similar manner, if we substitute Y with λ). Without any loss of generality, we may suppose that forager A_i simulates the work of matrix $m_i, 1 \leq i \leq n$.

Let us now consider the simulation of matrix m_i being of the form $(A \to \alpha', X \to Y)$.

First, let us suppose that m_i is without appearance checking, which means that $A \to \alpha' \notin \mathcal{F}$ and if A is not in the string, then the derivation fails. Let $A_i = (N_i \cup N_i^{(i)}, S_i, R_i)$, where $N_i = \{A, X, Z_0, Z_i, Z_{n+1}, Z_{n+2}\}$, $N_i^{(i)} = \{A^{(i)}, X^{(i)}, Z_0^{(i)}, Z_i^{(i)}, Z_{n+1}^{(i)}, Z_{n+2}^{(i)}\}$, and $S_i = Z_i$. The rule set of A_i, i.e. R_i, is as follows:

– $(l_{i_1} : Z_i \to Z_i^{(i)}, \{l_{i_2}\}, \{l_{i_1}\})$,

– $(l_{i_2} : Z_{n+2} \to Z^{(i)}_{n+2}, \{l_{i_4}\}, \{h_{i_2}\})$,

– $(h_{i_2} : Z^{(i)}_{n+2} \to Z^{(i)}_{n+2}, \{l_{i_6}\}, \{h_{i_2}\})$,

– $(l_{i_3} : A \to A^{(i)}, \{l_{i_5}\}, \{h_{i_2}\})$,

– $(h_{i_3} : A^{(i)} \to A^{(i)}, \{l_{i_5}\}, \{h_{i_2}\})$,

– $(l_{i_4} : X \to X^{(i)}, \{l_{i_3}\}, \{h_{i_2}\})$,

– $(h_{i_4} : X^{(i)} \to X^{(i)}, \{l_{i_3}\}, \{h_{i_2}\})$,

– $(l_{i_5} : Z_0 \to Z^{(i)}_0, \{l_{i_1}\}, \{h_{i_2}\})$,

– $(h_{i_5} : Z^{(i)}_0 \to Z^{(i)}_0, \{l_{i_5}\}, \{h_{i_2}\})$,

– $(l_{i_6} : Z_{n+1} \to Z^{(i)}_{n+1}, \{l_{i_6}\}, \{h_{i_2}\})$,

– $(h_{i_6} : Z^{(i)}_{n+1} \to Z^{(i)}_{n+1}, \{l_{i_6}\}, \{h_{i_2}\})$.

We explain how the work of a matrix m_i can be simulated by the interplay of A_i and the environment. Let us assume that we have a word of the form $CZ_iZ_0Z_{n+1}Z_{n+2}\omega$ in Γ, where ω' is the corresponding sentential form in G, $h(\omega') = \omega$. The application of environmental tables $P_{E_{i_1}}, P_{E_{i_2}}, P_{E_{i_3}}, P_{E_{i_4}}, P_{E_{i_5}}, P_{E_{i_6}}$ and $P_{E_{i_7}}$ follows in succession owing to their construction. In the meantime, no other table can be employed without introducing a trap symbol. The trap symbol may also indicate the lack of the appearance checking feature.

For forager A_i that has been designated for matrix m_i, we will construct the environmental tables. For technical reasons, we introduce the following rule sets:

$$\begin{aligned}
P_E^{\{b_m,D,F\}} =& \{b_m^{(n+m)} \to F, b'_m \to F, b_m \to b_m \mid b_m \in T, 1 \le m \le s\} \cup \\
& \{D \to D \mid D \in N\} \cup \{F \to F\},
\end{aligned}$$

$$\begin{aligned}
P_{E_{i,1}}^{\{Z_0,Z_p\}} =& \{Z_0 \to Z_0, Z_0^{(i)} \to F, Z_p \to F, Z_p^{(p)} \to F \mid 1 \le p \le n, p \ne i\},\\
P_{E_{i,2}}^{\{Z_0,Z_p\}} =& \{Z_0 \to Z_0, Z_0^{(i)} \to Z_0^{(i)}, Z_p \to F, Z_p^{(p)} \to F \mid 1 \le p \le n, p \ne i\},\\
P_{E_{i,3}}^{\{Z_0,Z_p\}} =& \{Z_0 \to F, Z_0^{(i)} \to Z_0, Z_p \to F, Z_p^{(p)} \to F \mid 1 \le p \le n, p \ne i\},\\
P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} =& \{Z_i \to F, Z_i^{(i)} \to Z_i^{(i)}, Z_{fin} \to F, Z_{fin}^{(fin)} \to F\},\\
P_{E_{i,2}}^{\{Z_i,Z_{fin}\}} =& \{Z_i \to F, Z_i^{(i)} \to \lambda, Z_{fin} \to F, Z_{fin}^{(fin)} \to F\},\\
P_{E_{i,1}}^{\{Z_{n+1}\}} =& \{Z_{n+1} \to Z_{n+1}, Z_{n+1}^{(i)} \to Z_{n+1}^{(i)}, Z_{n+1}^{(p)} \to F \mid 1 \le p \le n, p \ne i\},\\
P_{E_{i,2}}^{\{Z_{n+1}\}} =& \{Z_{n+1} \to Z_{n+1}, Z_{n+1}^{(p)} \to F \mid 1 \le p \le n\},\\
P_{E_{i,1}}^{\{Z_{n+2}\}} =& \{Z_{n+2} \to Z_{n+2}, Z_{n+2}^{(i)} \to Z_{n+2}^{(i)}, Z_{n+2}^{(p)} \to F \mid 1 \le p \le n, p \ne i\},
\end{aligned}$$

$$P_{E_{i,2}}^{\{Z_{n+2}\}} = \{Z_{n+2} \to Z_{n+2}, Z_{n+2}^{(i)} \to Z_{n+2}, Z_{n+2}^{(p)} \to F \mid 1 \le p \le n, p \ne i\}.$$

The first environmental table has the following form:

$$\begin{aligned} P_{E_{i_1}} = &\{C \to C^{i_1}, C^{i_j} \to F, C^{k_l} \to F \mid 1 \le j \le 6, 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{D^{(q)} \to F \mid D \in N, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

After we have performed the first step of the simulation, the word will be of the form $C^{i_1} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2} \omega$, $\omega \in (N \cup T)^*$, obtained by the joint action of forager A_i ($Z_i \to Z_i^{(i)}$) and the environmental table, $P_{E_{i_1}}$. If we had employed table $P_{E_{j_1}}$ for some $1 \le j \le n$, $j \ne i$, then at the next step the trap symbol would be introduced. The trap symbol would also appear, if we had applied environmental tables $P_{E_{i_2}}, P_{E_{i_3}}, P_{E_{i_4}}, P_{E_{i_5}}, P_{E_{i_6}}$ or $P_{E_{i_7}}$.

The second environmental table is as follows:

$$\begin{aligned} P_{E_{i_2}} = &\{C^{i_1} \to C^{i_2}, C^{i_j} \to F, C \to F, C^{k_l} \to F \mid 2 \le j \le 6, 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{D^{(q)} \to F \mid D \in N, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

The environmental word, which is received through the joint action of forager A_i ($Z_{n+2} \to Z_{n+2}^{(i)}$) and table $P_{E_{i_2}}$, has the form $C^{i_2} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2}^{(i)} \omega$, $\omega \in (N \cup T)^*$ after the second step of the simulation.

The third environmental table is of the form below:

$$\begin{aligned} P_{E_{i_3}} = &\{C^{i_2} \to C^{i_3}, C^{i_j} \to F \mid j = 1,3,4,5,6\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{D^{(q)} \to F \mid D \in N, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

By the action of A_i ($X \to X^{(i)}$) and $P_{E_{i_3}}$, we obtain $C^{i_3} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2}^{(i)} \gamma X^{(i)} \beta$ or $C^{i_3} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2}^{(i)} \omega$ depending on whether X is present in or absent from the environmental string. In the former case $\omega = \gamma X \beta, \beta, \gamma \in (N \cup T)^*$, in the latter ω remains unaltered.

If X has had an occurrence in the string, we have to check whether A is in the string. To this end, forager A_i employs rule $A \to A^{(i)}$. If the environmental word has not contained X, then forager A_i will apply an identical rewriting to $Z_{n+2}^{(i)}$.

The fourth environmental table has the following form:

$$\begin{aligned} P_{E_{i_4}} = &\{C^{i_3} \to C^{i_4}, C^{i_j} \to F \mid j = 1,2,4,5,6\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{X^{(i)} \to X^{(i)}, X^{(r)} \to F \mid 1 \le r \le n, r \ne i\} \cup \\ &\{B^{(q)} \to F \mid B \in N \setminus \{X\}, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

After the fourth step of the simulation, by the joint action of forager A_i and the environmental table, $P_{E_{i_4}}$, we obtain a word of one of the following forms: $C^{i_4}Z_i^{(i)}Z_0Z_{n+1}Z_{n+2}^{(i)}\bar{\gamma}X^{(i)}\delta A^{(i)}\bar{\beta}$, if both X and A have appeared in the string, $C^{i_4}Z_i^{(i)}Z_0Z_{n+1}Z_{n+2}^{(i)}\gamma X^{(i)}\beta$, if X has been present, but A is absent from the string, or $C^{i_4}Z_i^{(i)}Z_0Z_{n+1}Z_{n+2}^{(i)}\omega$, if X has not occurred in it. In the first case, $\omega = \bar{\gamma}X\delta A\bar{\beta}, \bar{\beta}, \bar{\gamma}, \delta \in (N \cup T)^*$, in the second case, $\omega = \gamma X\beta$, $\beta, \gamma \in (N \cup T)^*$, in the third case, ω does not change.

In the fifth step of the simulation, if both $X^{(i)}$ and $A^{(i)}$ appear in the string, forager A_i performs rule $Z_0 \to Z_0^{(i)}$, if $X^{(i)}$ is present in, but $A^{(i)}$ is absent from the string, A_i employs rule $Z_{n+2}^{(i)} \to Z_{n+2}^{(i)}$, if $X^{(i)}$ does not appear in the string, A_i substitutes Z_{n+1} for $Z_{n+1}^{(i)}$. In effect, the replacement of Z_{n+1} with $Z_{n+1}^{(i)}$ indicates that the derivation will not be successful. The web environment changes marker C^{i_4} and performs some identical rewritings in a parallel manner. Consequently, the fifth environmental table is of the form:

$$\begin{aligned}P_{E_{i_5}} = &\{C^{i_4} \to C^{i_5}, C^{i_j} \to F \mid j = 1,2,3,5,6\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{A^{(i)} \to A^{(i)}, X^{(i)} \to X^{(i)}\} \cup \{A^{(r)} \to F, X^{(r)} \to F \mid 1 \le r \le n, r \ne i\} \cup \\ &\{B^{(q)} \to F \mid B \in N \setminus \{A, X\}, 1 \le q \le n\} \cup \\ &P_E^{\{b_m, D, F\}} \cup P_{E_{i,2}}^{\{Z_0, Z_p\}} \cup P_{E_{i,1}}^{\{Z_i, Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}.\end{aligned}$$

As a result of the joint action of forager A_i and the environment, we attain one of the environmental words below: $C^{i_5}Z_i^{(i)}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\bar{\gamma}X^{(i)}\delta A^{(i)}\bar{\beta}$, $C^{i_5}Z_i^{(i)}Z_0Z_{n+1}Z_{n+2}^{(i)}\gamma X^{(i)}\beta$, or $C^{i_5}Z_i^{(i)}Z_0Z_{n+1}^{(i)}Z_{n+2}^{(i)}\omega$.

In the sixth step of the simulation, if both $X^{(i)}$ and $A^{(i)}$ occur in the string, forager A_i rewrites again Z_i to $Z_i^{(i)}$, if $X^{(i)}$ is present, but $A^{(i)}$ is absent from the string, or if $X^{(i)}$ does not appear in the string, A_i replaces Z_{n+1} with $Z_{n+1}^{(i)}$. The sixth environmental table has the form:

$$\begin{aligned}P_{E_{i_6}} = &\{C^{i_5} \to C^{i_6}, C^{i_j} \to F \mid j = 1,2,3,4,6\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{A^{(i)} \to A^{(i)}, X^{(i)} \to X^{(i)}\} \cup \{A^{(r)} \to F, X^{(r)} \to F \mid 1 \le r \le n, r \ne i\} \cup \\ &\{B^{(q)} \to F \mid B \in N \setminus \{A, X\}, 1 \le q \le n\} \cup \\ &P_E^{\{b_m, D, F\}} \cup P_{E_{i,2}}^{\{Z_0, Z_p\}} \cup P_{E_{i,2}}^{\{Z_i, Z_{fin}\}} \cup P_{E_{i,2}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}.\end{aligned}$$

As a consequence, we receive one of the following environmental words: $C^{i_6}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\bar{\gamma}X^{(i)}\delta A^{(i)}\bar{\beta}$, $C^{i_6}Z_0Z_{n+1}^{(i)}Z_{n+2}^{(i)}\gamma X^{(i)}\beta$, or the derivation will not lead to a terminal string.

In the last step, forager A_i applies rule $Z_i \to Z_i^{(i)}$. In the meantime, the web environment rewrites $X^{(i)}$ to Y and $A^{(i)}$ to α, $Z_{n+1}^{(i)}$ to F and performs some other kinds of rewritings in order to make it possible for another or for the same forager to continue the work. Taking everything into consideration, the seventh environmental table has the following form:

$$\begin{aligned}P_{E_{i_7}} =&\{C^{i_6} \to CZ_r \mid 1 \le r \le n\} \cup \{C^{i_6} \to CZ_{fin}, C^{i_j} \to F \mid 1 \le j \le 5\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \neq i, 1 \le l \le 6\} \cup \\ &\{A^{(i)} \to \alpha, X^{(i)} \to Y\} \cup \{A^{(r)} \to F, X^{(r)} \to F \mid 1 \le r \le n, r \neq i\} \cup \\ &\{B^{(q)} \to F \mid B \in N \setminus \{A, X\}, 1 \le q \le n\} \cup \\ &P_E^{\{b_m, D, F\}} \cup P_{E_{i,3}}^{\{Z_0, Z_p\}} \cup P_{E_{i,2}}^{\{Z_i, Z_{fin}\}} \cup P_{E_{i,2}}^{\{Z_{n+1}\}} \cup P_{E_{i,2}}^{\{Z_{n+2}\}}.\end{aligned}$$

The environmental word obtained by the joint action of forager A_i and table $P_{E_{i_7}}$ has the form $CZ_rZ_0Z_{n+1}Z_{n+2}\bar{\gamma}Y\delta\alpha\bar{\beta}$, or $CZ_{fin}Z_0Z_{n+1}Z_{n+2}\bar{\gamma}Y\delta\alpha\bar{\beta}$, if both $X^{(i)}$ and $A^{(i)}$ appear in the string, or the derivation will not be successful.

By the construction and explanations above, the reader can easily verify that the joint work of forager A_i and the environmental tables simulates the application of m_i and only that.

Secondly, let us suppose that m_i is with appearance checking, which means that $A \to \alpha' \in \mathcal{F}$ and it can be passed over if it cannot be applied. Let $A_i = (N_i \cup N_i^{(i)}, S_i, R_i)$, where $N_i = \{A, X, Z_0, Z_i, Z_{n+1}, Z_{n+2}\}$, $N_i^{(i)} = \{A^{(i)}, X^{(i)}, Z_0^{(i)}, Z_i^{(i)}, Z_{n+1}^{(i)}, Z_{n+2}^{(i)}\}$, and $S_i = Z_i$. The rule set R_i of A_i can be defined as follows:

- $(l_{i_1} : Z_i \to Z_i^{(i)}, \{l_{i_2}\}, \{l_{i_1}\})$,
- $(l_{i_2} : Z_{n+2} \to Z_{n+2}^{(i)}, \{l_{i_4}\}, \{h_{i_2}\})$,
- $(h_{i_2} : Z_{n+2}^{(i)} \to Z_{n+2}^{(i)}, \{l_{i_6}\}, \{h_{i_2}\})$,
- $(l_{i_3} : A \to A^{(i)}, \{l_{i_1}\}, \{h_{i_5}\})$,
- $(h_{i_3} : A^{(i)} \to A^{(i)}, \{l_{i_2}\}, \{h_{i_6}\})$,
- $(l_{i_4} : X \to X^{(i)}, \{l_{i_5}\}, \{h_{i_2}\})$,
- $(h_{i_4} : X^{(i)} \to X^{(i)}, \{l_{i_5}\}, \{h_{i_2}\})$,
- $(l_{i_5} : Z_0 \to Z_0^{(i)}, \{l_{i_3}\}, \{h_{i_2}\})$,
- $(h_{i_5} : Z_0^{(i)} \to Z_0^{(i)}, \{l_{i_1}\}, \{h_{i_2}\})$,
- $(l_{i_6} : Z_{n+1} \to Z_{n+1}^{(i)}, \{l_{i_6}\}, \{h_{i_2}\})$,
- $(h_{i_6} : Z_{n+1}^{(i)} \to Z_{n+1}^{(i)}, \{l_{i_6}\}, \{h_{i_2}\})$.

We present the environmental tables, which together with forager A_i, $1 \le i \le n$, simulate the application of m_i.

The first environmental table $P_{E_{i_1}}$ is of the following form:

$$\begin{aligned} P_{E_{i_1}} =&\{C \to C^{i_1}, C^{i_j} \to F, C^{k_l} \to F \mid 1 \le j \le 6, 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{D^{(q)} \to F \mid D \in N, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

After we have performed the first step of the simulation, the word will be of the form $C^{i_1} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2} \omega$, $\omega \in (N \cup T)^*$, obtained by the joint action of forager A_i ($Z_i \to Z_i^{(i)}$) and the environmental table, $P_{E_{i_1}}$. If we had employed table $P_{E_{j_1}}$ for some $1 \le j \le n$, $j \ne i$, then at the next step the trap symbol would be introduced. The same is true for environmental tables $P_{E_{i_2}}, P_{E_{i_3}}, P_{E_{i_4}}, P_{E_{i_5}}$, $P_{E_{i_6}}$ and $P_{E_{i_7}}$.

The second environmental table is presented below:

$$\begin{aligned} P_{E_{i_2}} =&\{C^{i_1} \to C^{i_2}, C^{i_j} \to F, C \to F, C^{k_l} \to F \mid 2 \le j \le 6, 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{D^{(q)} \to F \mid D \in N, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

The environmental word, which is received through the joint action of forager A_i ($Z_{n+2} \to Z_{n+2}^{(i)}$) and table $P_{E_{i_2}}$, has the form $C^{i_2} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2}^{(i)} \omega$, $\omega \in (N \cup T)^*$ after the second step of the simulation.

The third environmental table is of the form below:

$$\begin{aligned} P_{E_{i_3}} =&\{C^{i_2} \to C^{i_3}, C^{i_j} \to F \mid j = 1, 3, 4, 5, 6\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{D^{(q)} \to F \mid D \in N, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

By the action of A_i ($X \to X^{(i)}$) and $P_{E_{i_3}}$, we obtain $C^{i_3} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2}^{(i)} \gamma X^{(i)} \beta$, or $C^{i_3} Z_i^{(i)} Z_0 Z_{n+1} Z_{n+2}^{(i)} \omega$, depending on whether X is present in or absent from the environmental string. In the former case $\omega = \gamma X \beta, \beta, \gamma \in (N \cup T)^*$, in the latter ω remains unaltered.

In the fourth step, depending on the occurrence of $X^{(i)}$ in the environmental word, forager A_i either replaces Z_0 with $Z_0^{(i)}$, or applies an identical rewriting to $Z_{n+2}^{(i)}$. The fourth environmental table has to be constructed as follows:

$$\begin{aligned} P_{E_{i_4}} =&\{C^{i_3} \to C^{i_4}, C^{i_j} \to F \mid j = 1, 2, 4, 5, 6\} \cup \\ &\{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ &\{X^{(i)} \to X^{(i)}, X^{(r)} \to F \mid 1 \le r \le n, r \ne i\} \cup \\ &\{B^{(q)} \to F \mid B \in N \setminus \{X\}, 1 \le q \le n\} \cup \\ &P_E^{\{b_m,D,F\}} \cup P_{E_{i,1}}^{\{Z_0,Z_p\}} \cup P_{E_{i,1}}^{\{Z_i,Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

After the fourth step of the simulation we obtain by the joint action of forager A_i and the environmental table, $P_{E_{i_4}}$, a word of one of the following forms: $C^{i_4}Z_i^{(i)}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\gamma X^{(i)}\beta$, if $X^{(i)}$ occurs in the string, or $C^{i_4}Z_i^{(i)}Z_0Z_{n+1}Z_{n+2}^{(i)}\omega$ otherwise. In the first case, $\omega = \gamma X\beta$, $\beta, \gamma \in (N \cup T)^*$, in the second case, ω does not change.

In the fifth step of the simulation, we control the presence of A in the string or introduce $Z_{n+1}^{(i)}$ into the string. In the first case, agent A_i employs rule $A \to A^{(i)}$, in the second case, it substitutes Z_{n+1} for $Z_{n+1}^{(i)}$. The fifth environmental table is of the form:

$$\begin{aligned} P_{E_{i_5}} =& \{C^{i_4} \to C^{i_5}, C^{i_j} \to F \mid j = 1, 2, 3, 5, 6\} \cup \\ & \{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ & \{A^{(i)} \to A^{(i)}, X^{(i)} \to X^{(i)}\} \cup \{A^{(r)} \to F, X^{(r)} \to F \mid 1 \le r \le n, r \ne i\} \cup \\ & \{B^{(q)} \to F \mid B \in N \setminus \{A, X\}, 1 \le q \le n\} \cup \\ & P_E^{\{b_m, D, F\}} \cup P_{E_{i,2}}^{\{Z_0, Z_p\}} \cup P_{E_{i,1}}^{\{Z_i, Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

As a result of the joint action of forager A_i and the environment, we attain one of the environmental words $C^{i_5}Z_i^{(i)}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\gamma' X^{(i)}\delta A^{(i)}\beta'$, $C^{i_5}Z_i^{(i)}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\gamma X^{(i)}\beta$, or $C^{i_5}Z_i^{(i)}Z_0Z_{n+1}^{(i)}Z_{n+2}^{(i)}\omega$. In the first case, $\omega = \bar{\gamma}X\delta A\bar{\beta}, \bar{\beta}, \bar{\gamma}, \delta \in (N \cup T)^*$, in the second case, $\omega = \gamma X\beta$, $\beta, \gamma \in (N \cup T)^*$, in the third case, ω does not change.

In the sixth step, if $X^{(i)}$ and $A^{(i)}$ are in the string, then forager A_i attempts to employ $Z_i \to Z_i^{(i)}$. If only $X^{(i)}$ occurs in the environmental word, then forager A_i rewrites $Z_0^{(i)}$ identically. Should $X^{(i)}$ be absent from the string, then A_i tries to substitute Z_{n+1} for $Z_{n+1}^{(i)}$. The sixth environmental table can be constructed as follows:

$$\begin{aligned} P_{E_{i_6}} =& \{C^{i_5} \to C^{i_6}, C^{i_j} \to F \mid j = 1, 2, 3, 4, 6\} \cup \\ & \{C \to F, C^{k_l} \to F \mid 1 \le k \le n, k \ne i, 1 \le l \le 6\} \cup \\ & \{A^{(i)} \to A^{(i)}, X^{(i)} \to X^{(i)}\} \cup \{A^{(r)} \to F, X^{(r)} \to F \mid 1 \le r \le n, r \ne i\} \cup \\ & \{B^{(q)} \to F \mid B \in N \setminus \{A, X\}, 1 \le q \le n\} \cup \\ & P_E^{\{b_m, D, F\}} \cup P_{E_{i,2}}^{\{Z_0, Z_p\}} \cup P_{E_{i,1}}^{\{Z_i, Z_{fin}\}} \cup P_{E_{i,1}}^{\{Z_{n+1}\}} \cup P_{E_{i,1}}^{\{Z_{n+2}\}}. \end{aligned}$$

By the joint action of forager A_i and the environment, we receive $C^{i_6}Z_i^{(i)}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\bar{\gamma}X^{(i)}\delta A^{(i)}\bar{\beta}$, $C^{i_6}Z_i^{(i)}Z_0^{(i)}Z_{n+1}Z_{n+2}^{(i)}\gamma X^{(i)}\beta$, or $C^{i_6}Z_i^{(i)}Z_0Z_{n+1}^{(i)}Z_{n+2}^{(i)}\omega$. In the first case, $\omega = \bar{\gamma}X\delta A\bar{\beta}, \bar{\beta}, \bar{\gamma}, \delta \in (N \cup T)^*$, in the second case, $\omega = \gamma X\beta$, $\beta, \gamma \in (N \cup T)^*$, in the third case, ω does not change.

In the last step, forager A_i applies rule $Z_i \to Z_i^{(i)}$, if $X^{(i)}$ is in the string. If $X^{(i)}$ does not have an occurrence in the string, agent A_i tries to rewrite Z_{n+2} to $Z_{n+2}^{(i)}$. In the meantime, the web environment replaces $X^{(i)}$ with Y, $A^{(i)}$ with α (if $A^{(i)}$ is not present in the string, then the rule $A^{(i)} \to \alpha$ will not be employed), and $Z^{(i)}_{n+1}$ with F and performs some other kinds of rewritings in order to make it possible for another or for the same forager to continue the work. Taking everything into consideration, the seventh environmental table must have the following form:

$$\begin{aligned} P_{E_{i_7}} =& \{C^{i_6} \to CZ_r \mid 1 \leq r \leq n\} \cup \{C^{i_6} \to CZ_{fin}, C^{i_j} \to F \mid 1 \leq j \leq 5\} \cup \\ & \{C \to F, C^{k_l} \to F \mid 1 \leq k \leq n, k \neq i, 1 \leq l \leq 6\} \cup \\ & \{A^{(i)} \to \alpha, X^{(i)} \to Y\} \cup \{A^{(r)} \to F, X^{(r)} \to F \mid 1 \leq r \leq n, r \neq i\} \cup \\ & \{B^{(q)} \to F \mid B \in N \setminus \{A, X\}, 1 \leq q \leq n\} \cup \\ & P_E^{\{b_m, D, F\}} \cup P_{E_{i,3}}^{\{Z_0, Z_p\}} \cup P_{E_{i,2}}^{\{Z_i, Z_{fin}\}} \cup P_{E_{i,2}}^{\{Z_{n+1}\}} \cup P_{E_{i,2}}^{\{Z_{n+2}\}}. \end{aligned}$$

The environmental word obtained by the joint action of forager A_i and table $P_{E_{i_7}}$ has the form $CZ_rZ_0Z_{n+1}Z_{n+2}\bar{\gamma}Y\delta\alpha\bar{\beta}$ $(CZ_{fin}Z_0Z_{n+1}Z_{n+2}\bar{\gamma}Y\delta\alpha\bar{\beta})$, $CZ_rZ_0Z_{n+1}Z_{n+2}\gamma Y\beta$ $(CZ_{fin}Z_0Z_{n+1}Z_{n+2}\gamma Y\beta)$, or the derivation will not lead to a terminal word.

As in the previous case, the reader can verify that the joint work of forager A_i and the environmental tables simulates the application of m_i and only that.

At some stage of the derivation, we guess whether ω is a terminal word or not (the environmental word is of the form $CZ_{fin}Z_0Z_{n+1}Z_{n+2}\omega$). We check this conjecture by the joint action of foragers A_{n+j}, $1 \leq j \leq s$, and environmental table $P_{E_{fin}}$.

Let $A_{n+j} = (N_{n+j} \cup N^{(n+j)}_{n+j}, S_{n+j}, R_{n+j})$, where $N_{n+j} = \{b_j\}$, $N^{(n+j)}_{n+j} = \{b^{(n+j)}_j\}$, and $S_{n+j} = b_j$, $1 \leq j \leq s$. The rule set of A_{n+j}, i.e. R_{n+j}, $1 \leq j \leq s$, is defined as follows: $(b_j \in T = \{b_j \mid 1 \leq j \leq s\})$:

– $(l_j : b_j \to b^{(n+j)}_j, \{l_j\}, \{l_j\})$.

Observe that $(l_j : b_j \to b^{(n+j)}_j, \{l_j\}, \{l_j\})$ is the only rule of forager A_{n+j}, thus its initial rule, as well.

Let

$$\begin{aligned} P_{E_{fin}} =& \{C^{i_p} \to F \mid 1 \leq i \leq n, 1 \leq p \leq 6\} \cup \{C \to \lambda\} \cup \\ & \{D \to F, D^{(q)} \to F \mid D \in N, 1 \leq q \leq n\} \cup \\ & \{b^{(n+j)}_j \to b'_j, b'_j \to b'_j, b_j \to b_j \mid b_j \in T, 1 \leq j \leq s\} \cup \{F \to F\} \cup \\ & \{Z_0 \to \lambda, Z^{(k)}_0 \to F, Z_{fin} \to \lambda, Z^{(fin)}_{fin} \to F \mid 1 \leq k \leq n\} \cup \\ & \{Z_k \to \lambda, Z^{(k)}_k \to F \mid 1 \leq k \leq n\} \cup \{Z_{n+1} \to \lambda, Z^{(k)}_{n+1} \to F \mid 1 \leq k \leq n\} \cup \\ & \{Z_{n+2} \to \lambda, Z^{(k)}_{n+2} \to F \mid 1 \leq k \leq n\}. \end{aligned}$$

Foragers A_{n+j}, $1 \leq j \leq s$, rewrite terminal letters $b_j, 1 \leq j \leq s$, to their indexed versions $b^{(n+j)}_j$, and in the meantime environmental table $P_{E_{fin}}$ deletes the marker symbols, introduces F for the letters from N and rewrites letters $b^{(n+j)}_j$, $1 \leq j \leq s$, to b'_j. The work of foragers $A_{n+j}, 1 \leq j \leq s$, cannot be interfered with the work of the other foragers. The joint work of foragers A_{n+j}, $1 \leq j \leq s$, and environmental table $P_{E_{fin}}$ is iterated as many times as it is necessary. At the end of the procedure, if the word obtained is a string over T', then a word that can be generated by G is received.

It only remains to be shown how the simulation begins. The initial state of the web environment is $\omega = CZ_0Z_{n+1}Z_{n+2}S$. We have to simulate matrix $(S \to AX)$. Let us assume that the web environment has a table $P_{E_{start}}$ that performs this simulation, where

$$\begin{aligned}
P_{E_{start}} = &\{C \to CZ_k, Z_k \to F, Z_k^{(k)} \to F \mid 1 \leq k \leq n\} \cup \\
&\{C^{p_q} \to F \mid 1 \leq p \leq n, 1 \leq q \leq 6\} \cup \\
&\{S \to AX\} \cup \{D \to F \mid D \in N \setminus \{S\}\} \cup \{B^{(m)} \to F \mid B \in N, 1 \leq m \leq n\} \cup \\
&\{b_j \to F, b'_j \to F, b_j^{(n+j)} \to F \mid b_j \in T, 1 \leq j \leq s\} \cup \\
&\{Z_0 \to Z_0, Z_0^{(k)} \to F, Z_{fin} \to F, Z_{fin}^{(fin)} \to F \mid 1 \leq k \leq n\} \cup \\
&\{Z_{n+1} \to Z_{n+1}, Z_{n+1}^{(k)} \to F \mid 1 \leq k \leq n\} \cup \\
&\{Z_{n+2} \to Z_{n+2}, Z_{n+2}^{(k)} \to F \mid 1 \leq k \leq n\} \cup \{F \to F\}.
\end{aligned}$$

Initially, all the foragers are inactive, since there is not a $Z_k, 1 \leq k \leq n$, in the sentential form. The derivation is as follows: $CZ_0Z_{n+1}Z_{n+2}S \Longrightarrow_\Gamma CZ_kZ_0Z_{n+1}Z_{n+2}AX$ for some $k, 1 \leq k \leq n$. Since S does not occur in any other matrix, this is the only way how the simulation can begin.

Owing to the form of the tables, it can be seen that all but no more words than the elements of $L(G)$ can be derived. Hence the theorem is verified. □

3.3. Eco–Foraging Systems with Time

Secondly, we move on to eco–foraging systems where certain web pages may have lifetime. We modify the alphabet of agents to keep track of the aging of the web environment [12]. If the lifetime of a web page is 0, it signifies that the web page is no longer recognizable by any of the foragers.

Definition 6. *The web environment with n foragers with time*, $n \geq 1$, is a construction

$$E = (V_E, T_E, \mathcal{P}_E)$$

such that

- V_E is a finite alphabet, where $V_E = V_M \cup T_E \cup V_N \cup \bar{V}_N$ with $V_N = \bigcup_{i=1}^n N_i$ and $\bar{V}_N = \bigcup_{i=1}^n N_i^{(i)}$, $n \geq 1$, where
  - V_M is a finite set,
  - $N_i = \bigcup_{j=1}^{s_i} N_{i,j}$, $N_{i,j} = \bigcup_{k=0}^{t_{i,j}} N_{i,j}(k) = \bigcup_{k=0}^{t_{i,j}} \{X_{i,j}(k)\}$,
  - $N_i^{(i)} = \bigcup_{j=1}^{s_i} N_{i,j}^{(i)}$, $N_{i,j}^{(i)} = \bigcup_{k=0}^{t_{i,j}} N_{i,j}^{(i)}(k) = \bigcup_{k=0}^{t_{i,j}} \{X_{i,j}^{(i)}(k)\}$,

- T_E is a finite alphabet, and
- V_M, T_E, V_N and $\bar{V}_N$ are pairwise disjoint sets,

- $\mathcal{P}_E = \{P_{E_1}, \ldots, P_{E_m}\}$, where P_{E_q}, $1 \leq q \leq m$, is a finite set of rules of the following forms, $V_{N_{max}} = \bigcup_{p=1}^{n} \bigcup_{r=1}^{s_p} N_{p,r}(t_{p,r})$, $t_{p,r} \geq 1$, $1 \leq i \leq n, 1 \leq j \leq s_i$, $1 \leq k \leq t_{i,j}$:
  - $X_{i,j}(k) \to X_{i,j}(k-1)$, where $X_{i,j}(k-1), X_{i,j}(k) \in N_{i,j}$,
  - $Y \to \alpha$, where $Y \in \bigcup_{i=1}^{n} N_i$, and $\alpha \in (T_E \cup V_{N_{max}})^*$,
  - $X_{i,j}^{(i)}(k) \to \beta$, where $X_{i,j}^{(i)}(k) \in N_{i,j}^{(i)}$, and $\beta \in (T_E \cup V_{N_{max}})^* \cup (T_E \cup V_{N_{max}})^* X_{i,j}^{(i)}(k-1)(T_E \cup V_{N_{max}})^*$, $X_{i,j}^{(i)}(k-1) \in N_{i,j}^{(i)}$,
  - $X_{i,j}(0) \to X_{i,j}(0)$, $X_{i,j}(0) \in N_{i,j}$,
  - $X_{i,j}^{(i)}(0) \to X_{i,j}^{(i)}(0)$, $X_{i,j}^{(i)}(0) \in N_{i,j}^{(i)}$, or
  - $U \to \gamma$, where $U \in V_M$ and $\gamma \in (T_E \cup V_{N_{max}})^* \cup (T_E \cup V_{N_{max}})^* V_M (T_E \cup V_{N_{max}})^*$.

  Moreover, any rule set in $\mathcal{P}_E$ is complete, i.e. for any $c \in V_E$, there is at least one rule in any P_{E_q}, $1 \leq q \leq m$.

If the lifetime of the web pages is included, then the interpretation of the various components of the web environment is analogous to the one presented for Definition 1. Therefore herein we emphasize only the differences. The alphabet of an agent also contains the information about the lifetime of the web pages that the agent is able to recognize. We assign a maximal lifetime to each web page. If the environment rewrites a web page and the web page will still be present in the environmental string, then the lifetime of the web page will be reduced by one regardless of whether any agents have managed to identify the web page or not. The lifetime of the newly introduced web pages will be maximal (in $V_{N_{max}}$). We do not assign lifetime to the elements of V_M and T_E. Furthermore, T_E is disjoint from V_M, V_N and $\bar{V}_N$. While in Definition 1 the elements of T_E can be rewritten by the agents, in this definition they can be changed by the environment only.

Definition 7. *A programmed eco–foraging system with appearance checking with time (an $FEG_{PR_{ac}}^{time}$ system) of degree n, $n \geq 1$,* is a construction

$$\Gamma = (E, A_1, \ldots, A_n, c_{init})$$

such that

- $E = (V_E, T_E, \mathcal{P}_E)$ is the web environment (see Definition 6),
- $A_i = (N_i \cup N_i^{(i)}, S_i, R_i)$, $1 \leq i \leq n$, is the i–th forager, a programmed grammar scheme with appearance checking, where
  - $N_i \cup N_i^{(i)}$ is the nonterminal alphabet of the i–th forager (see Definition 6),

- $S_i \in N_i$ is the start symbol of the i–th forager, $S_i = \{X_{i,1}\}$,
- R_i is a finite set of rules of the following forms:
  - $(l_{i,1}(k) : S_i(k) \to S_i^{(i)}(k-1), \sigma_i(l_{i,1}(k)), \psi_i(l_{i,1}(k)))$, $\sigma_i(l_{i,1}(k)) \subseteq l_{i,1} \cup \ldots \cup l_{i,s_i}$, $\psi_i(l_{i,1}(k)) \subseteq l_{i,1}$, $1 \leq k \leq t_{i,1}$, is called the initial rule of the i–th forager, where $l_{i,j} = \{l_{i,j}(z) \mid 1 \leq z \leq t_{i,j}\}$, $1 \leq j \leq s_i$,
  - $(l_{i,j}(k) : X_{i,j}(k) \to X_{i,j}^{(i)}(k-1), \sigma_i(l_{i,j}(k)), \psi_i(l_{i,j}(k)))$, $X_{i,j}(k) \in N_i \setminus \{S_i\}, X_{i,j}^{(i)}(k-1) \in N_i^{(i)} \setminus \{S_i^{(i)}\}$, $2 \leq j \leq s_i$, with $\sigma_i(l_{i,j}(k)) \subseteq l_{i,1} \cup \ldots \cup l_{i,s_i}, \psi_i(l_{i,j}(k)) \subseteq h_{i,2} \cup \ldots \cup h_{i,s_i}$, where $l_{i,j} = \{l_{i,j}(z) \mid 1 \leq z \leq t_{i,j}\}$, $1 \leq j \leq s_i$, $h_{i,j'} = \{h_{i,j'}(z) \mid 1 \leq z \leq t_{i,j'}\}$, $2 \leq j' \leq s_i$, or
  - $(h_{i,j}(k) : X_{i,j}^{(i)}(k) \to X_{i,j}^{(i)}(k-1), \sigma_i(h_{i,j}(k)), \psi_i(h_{i,j}(k)))$, $X_{i,j}^{(i)}(k)$, $X_{i,j}^{(i)}(k-1) \in N_i^{(i)} \setminus \{S_i^{(i)}\}$, $2 \leq j \leq s_i$, with $\sigma_i(h_{i,j}(k)) \subseteq l_{i,1} \cup \ldots \cup l_{i,s_i}, \psi_i(h_{i,j}(k)) \subseteq h_{i,2} \cup \ldots \cup h_{i,s_i}$, where $l_{i,j} = \{l_{i,j}(z) \mid 1 \leq z \leq t_{i,j}\}$, $1 \leq j \leq s_i$, $h_{i,j'} = \{h_{i,j'}(z) \mid 1 \leq z \leq t_{i,j'}\}$, $2 \leq j' \leq s_i$, and
    - $\cdot$ $Label(R_i) = l_{i,1} \cup \ldots \cup l_{i,s_i} \cup h_{i,2} \cup \ldots \cup h_{i,s_i}$ is the set of labels of R_i.

- $c_{init} = (l_{1,1}(t_{1,1}), \ldots, l_{n,1}(t_{n,1}); \omega_{init})$, where $l_{i,1}(t_{i,1})$, $t_{i,1} \geq 1$, is the label of the initial rule of the i–th forager with the corresponding maximal time, $1 \leq i \leq n$, and $\omega_{init} = z_0 X_{j_1}(t_{j_1}) z_1 \ldots z_{k-1} X_{j_k}(t_{j_k}) z_k$, $X_{j_h}(t_{j_h}) \in N_{j_h}$, $t_{j_h} \geq 1$, $z_0 \in T_E^* \cup T_E^* V_M T_E^*$, $z_l \in T_E^*, 1 \leq h \leq k, 1 \leq l \leq k, \{j_1, \ldots, j_k\} \subseteq \{1, \ldots, n\}$, is called the initial configuration of Γ. The string ω_{init} is called the initial state of the web environment of Γ or the initial environmental state.

In Definition 7 when the agent tries to visit a not yet discovered web page employing rules of the form $X_{i,j}(k) \to X_{i,j}^{(i)}(k-1)$, $X_{i,j}(k) \in \bar{N}_i, X_{i,j}^{(i)}(k-1) \in N_i^{(i)}$, $1 \leq i \leq n, 1 \leq j \leq s_i$, $1 \leq k \leq t_{i,j}$, then the lifetime of the web page will be reduced by one, if the application of the rule has been successful. Should the agent go to a web page that it has discovered previously using rules of the form $X_{i,j}^{(i)}(k) \to X_{i,j}^{(i)}(k-1)$, $X_{i,j}^{(i)}(k-1), X_{i,j}^{(i)}(k) \in N_i^{(i)}$, $1 \leq i \leq n, 2 \leq j \leq s_i$, $1 \leq k \leq t_{i,j}$, then the lifetime of the corresponding web page will be decreased by one again. In the axiom, the lifetime of the nonterminal letters of those agents that are able to commence their work is maximal. Notice that ω_{init} contains at most one symbol from V_M.

In the sequel, we define the way in which eco–foraging systems with time work.

Definition 8. Let $\Gamma = (E, A_1, \ldots, A_n, c_{init})$, be an $\mathrm{FEG}_{PR_{ac}}^{time}$ system of degree n, $n \geq 1$ (see Definition 7). *An $(n+1)$–tuple $c = (q_{1,j_1}(k_{1,j_1}), \ldots, q_{n,j_n}(k_{n,j_n}); \omega_E)$, where $q_{i,j_i}(k_{i,j_i}) \in Label(R_i)$, $1 \leq k_{i,j_i} \leq t_{i,j_i}$, $1 \leq i \leq n$, $1 \leq j_i \leq s_i$, and $\omega_E \in V_E^*$*, is called a configuration of Γ. ω_E is the state of the web environment of Γ in configuration c or the environmental state in configuration c.

Definition 9. Let $\Gamma = (E, A_1, \ldots, A_n, c_{init})$, $1 \leq t_{i,1}$, $1 \leq i \leq n$, be an $\mathrm{FEG}_{PR_{ac}}^{time}$ system of degree n (see Definition 7). Let $c_1 = (q_{1,j_1}(k_{1,j_1}), \ldots, q_{n,j_n}(k_{n,j_n}); \omega_E)$, $c_2 = (q'_{1,j_1}(k'_{1,j_1}), \ldots, q'_{n,j_n}(k'_{n,j_n}); \omega'_E)$ be two configurations of Γ, $1 \leq k_{i,j_i}, k'_{i,j_i} \leq$

t_{i,j_i}, $1 \leq i \leq n$, $1 \leq j_i \leq s_i$, and $\omega_E, \omega'_E \in V_E^*$. We say that c_1 *directly derives* c_2, written as $c_1 \Longrightarrow_\Gamma c_2$, if the following conditions hold:

1. $\omega_E = u_1\alpha_{i_1}(k_{i_1})u_2 \ldots u_r\alpha_{i_r}(k_{i_r})u_{r+1}$ and $\omega''_E = u_1\beta_{i_1}(k_{i_1}-1)u_2 \ldots u_r\beta_{i_r}(k_{i_r}-1)u_{r+1}$, where $\{i_1, \ldots, i_r\} \subseteq \{1, \ldots, n\}$, $\alpha_{i_j}(k_{i_j}) \in N_j \cup N_j^{(j)}, \beta_{i_j}(k_{i_j}-1) \in N_j^{(j)}$, $1 \leq k_{i_j} \leq t_{i_j}$, $1 \leq j \leq r$, $u_h \in V_E^*$, $1 \leq h \leq r+1$,

2. $(q_{i_j}(k_{i_j}) : \alpha_{i_j}(k_{i_j}) \to \beta_{i_j}(k_{i_j}-1), \sigma(q_{i_j}(k_{i_j})), \psi(q_{i_j}(k_{i_j}))) \in R_{i_j}$ and $q'_{i_j}(k'_{i_j}) \in \sigma(q_{i_j}(k_{i_j}))$, $1 \leq j \leq r$,

3. there is no $m \in \{1, \ldots, n\} \setminus \{i_1, \ldots, i_r\}$ such that $(q_m(k_m) : \alpha_{i_m}(k_m) \to \beta_{i_m}(k_m-1), \sigma(q_m(k_m)), \psi(q_m(k_m))) \in R_m$ can be applied to $u_1u_2 \ldots u_{r+1}$, $1 \leq k_m \leq t_m$,

4. $q'_m(k'_m) \in \psi(q_m(k_m))$ for $m \in \{1, \ldots, n\} \setminus \{i_1, \ldots, i_r\}$, $1 \leq k'_m \leq t'_m$,

5. $\omega'_E = v_1\beta_{i_1}(k_{i_1}-1)v_2 \ldots v_r\beta_{i_r}(k_{i_r}-1)v_{r+1}$, where $u_1 \ldots u_{r+1} \Longrightarrow v_1 \ldots v_{r+1}$ is a 0L rewriting according to some P_{E_q}, $1 \leq q \leq m$, where $P_{E_q} \in \mathcal{P}_E$.

The function of a programmed eco–foraging system with time is analogous to the function of a programmed eco–foraging system without time (see Definition 4), therefore herein we present only the differences. In Definition 9, the agents that participate in the derivation will reduce the lifetime of the web pages they visit. The lifetime of the web pages remaining still present in the environment or being introduced at the given step, will be modified by the environment as it is described in the remark following Definition 6.

The transitive (and reflexive) closure of $\Longrightarrow_\Gamma$ is denoted by $\Longrightarrow_\Gamma^+$ ($\Longrightarrow_\Gamma^*$).

Definition 10. *The language generated by an* $FEG_{PR_{ac}}^{time}$ *system* $\Gamma = (E, A_1, \ldots, A_n, c_{init})$ is defined by $L(\Gamma) = \{y \mid c_{init} = (l_{1,1}(t_{1,1}), \ldots, l_{n,1}(t_{n,1}); \omega_{init}) \Longrightarrow_\Gamma^* (q_{1,j_1}(k_{1,j_1}), \ldots, q_{n,j_n}(k_{n,j_n}); y), y \in T_E^*\}$.

If no confusion arises, then subscript Γ can be omitted.

The family of languages generated by $\mathrm{FEG}_{PR_{ac}}^{time}$ systems is denoted by $\mathcal{L}(\mathrm{FEG}_{PR_{ac}}^{time})$.

Let us now illustrate how programmed eco–foraging systems with time work through an example.

Example 2. Let $L_2 = \{A^nB^nC^n \mid n \geq 1\}$. The programmed eco–foraging system Γ with time with appearance checking that generates L_2, i.e. $L_2 = L(\Gamma)$, is as follows:

$$\Gamma = (E, A_1, A_2, A_3, c_{init})$$

such that

– $E = (V_E, T_E, \mathcal{P}_E)$ is the web environment, where

- $V_E = \{A(t), B(t), C(t) \mid 0 \le t \le 3\} \cup \{A^{(1)}(t), B^{(2)}(t), C^{(3)}(t) \mid 0 \le t \le 3\} \cup \{A, B, C\}$,
- $T_E = \{A, B, C\}$,
- $\mathcal{P}_E = \{P_{E_1}, P_{E_2}\}$, where
  - $P_{E_1} = \{A(0) \to A(0), A(t) \to A(t-1), B(0) \to B(0), B(t) \to B(t-1), C(0) \to C(0), C(t) \to C(t-1), A^{(1)}(0) \to A^{(1)}(0), A^{(1)}(t) \to A, B^{(2)}(0) \to B^{(2)}(0), B^{(2)}(t) \to B, C^{(3)}(0) \to C^{(3)}(0), C^{(3)}(t) \to C, A \to A, B \to B, C \to C \mid 1 \le t \le 3\}$,
  - $P_{E_2} = \{A(0) \to A(0), A(t) \to A(t-1), B(0) \to B(0), B(t) \to B(t-1), C(0) \to C(0), C(t) \to C(t-1), A^{(1)}(0) \to A^{(1)}(0), A^{(1)}(t) \to A(3)A^{(1)}(t-1), B^{(2)}(0) \to B^{(2)}(0), B^{(2)}(t) \to B(3)B^{(1)}(t-1), C^{(3)}(0) \to C^{(3)}(0), C^{(3)}(t) \to C(3)C^{(1)}(t-1), A \to A, B \to B, C \to C \mid 1 \le t \le 3\}$,

- $A_i = (N_i \cup N_i^{(i)}, S_i, R_i)$, $i = 1, 2, 3$, is the i–th forager, where
  - $N_1 = \{A(t) \mid 0 \le t \le 3\}, N_1^{(1)} = \{A^{(1)}(t) \mid 0 \le t \le 3\}, N_2 = \{B(t) \mid 0 \le t \le 3\}, N_2^{(2)} = \{B^{(2)}(t) \mid 0 \le t \le 3\}, N_3 = \{C(t) \mid 0 \le t \le 3\}, N_3^{(3)} = \{C^{(3)}(t) \mid 0 \le t \le 3\}$,
  - the triplets of R_i are of the following forms:
    - $(l_i(t) : X_i(t) \to X_i^{(i)}(t-1), \sigma_i(l_i(t)), \psi_i(l_i(t)))$, with $\sigma_i(l_i(t)) \subseteq l_i, \psi_i(l_i(t)) \subseteq l_i$, $l_i = \{l_i(t) \mid 1 \le t \le 3\}$, $X_i(t) \in N_i$, $X_i^{(i)}(t-1) \in N_i^{(i)}$, $1 \le t \le 3$, where
      - · $Label(R_i) = \{l_i(t) \mid 1 \le t \le 3\}$ and $\sigma_i, \psi_i : Label(R_i) \to 2^{Label(R_i)}, i = 1, 2, 3$,

- $c_{init} = (l_1(3), l_2(3), l_3(3); \omega_{init})$, where $\omega_{init} = A(3)B(3)C(3)$.

At the first step all foragers are active. They rewrite different symbols from the environmental string to their indexed versions and the lifetime of the web pages will be reduced. Forager A_1 changes $A(3)$ to $A^{(1)}(2)$, forager A_2 $B(3)$ to $B^{(2)}(2)$ and forager A_3 $C(3)$ to $C^{(3)}(2)$. The web environment remains inactive. From $A^{(1)}(2)B^{(2)}(2)C^{(3)}(2)$ we have two possibilities to continue. If we apply environmental table P_{E_1}, then we will obtain: $A^{(1)}(2)B^{(2)}(2)C^{(3)}(2) \Longrightarrow_\Gamma ABC$. If we use P_{E_2}, then we will receive: $A^{(1)}(2)B^{(2)}(2)C^{(3)}(2) \Longrightarrow_\Gamma A(3)A^{(1)}(1)B(3)B^{(2)}(1)C(3)C^{(3)}(1)$. ABC is a terminal string. Continuing the derivation with $A(3)A^{(1)}(1)B(3)B^{(2)}(1)C(3)C^{(3)}(1)$, we can employ either environmental table P_{E_1} or table P_{E_2}. Let us assume that we will use environmental table P_{E_1} (the continuation of the derivation for table P_{E_2} may be done analogously, thus it is left to the reader). As a result of the utilization of P_{E_1}, we will attain: $A(3)A^{(1)}(1)B(3)B^{(2)}(1)C(3)C^{(3)}(1) \Longrightarrow_\Gamma A^{(1)}(2)AB^{(2)}(2)BC^{(3)}(2)C \Longrightarrow_\Gamma AABBCC$. We applied again environmental table P_{E_1} at the last step and received a terminal string. The derivation could have been continued in a different way, if we had employed environmental table P_{E_2} at the last step. The verification is left to the reader.

Observe that L_2 is a context–sensitive language, but it is not context–free.

3.4. The Power of Eco–Foraging Systems with Time

We will show that the language family determined by unordered scattered context grammars of finite index is equal to the language family generated by programmed eco–foraging systems with time with appearance checking.

Theorem 2. $\mathcal{L}_{fin}(\mathrm{USC}) = \mathcal{L}(\mathrm{FEG}^{time}_{PR_{ac}})$.

Proof. First, we will prove that $\mathcal{L}_{fin}(\mathrm{USC}) \subseteq \mathcal{L}(\mathrm{FEG}^{time}_{PR_{ac}})$.

Let L be a language generated by an unordered scattered context grammar $G = (N, T, S, P)$ of finite index, where $ind(L) = ind(L(G))$. To verify the statement, we will construct an $\mathrm{FEG}^{time}_{PR_{ac}}$ system Γ such that $L(\Gamma) = L(G)$.

Let $P = \{p_1, p_2, \ldots, p_s\}$, where $p_l : (X_{l,1} \to x_{l,1}, \ldots, X_{l,k_l} \to x_{l,k_l})$, $1 \le l \le s, k_l \ge 1$. Let us assume that $ind(L(G)) = r$ and $r \ge k_l$. Then for all $\omega \in L(G)$, there exists a derivation $S \Longrightarrow \omega_1 \Longrightarrow \omega_2 \ldots \Longrightarrow \omega_z = \omega, z \ge 1$, such that there are at most r nonterminal symbols in $\omega_q, 1 \le q \le z$. These are the derivations that we will simulate.

Before we give the details of the simulation, we make some observations about the derivations of finite index in G.

Let $\omega = u_1 D_1 u_2 \ldots u_j D_j u_{j+1}$ be a sentential form in G, where $j \le r$, $D_k \in N$, $u_h \in T^*$, $1 \le k \le j$, $1 \le h \le j+1$. Let us call $[D_1 \ldots D_j]$ the nonterminal cut of ω, and let us denote the nonterminal cut by $c^{(nt)}(\omega)$. We regard nonterminal cuts $[D_1 \ldots D_j]$ as equivalent with respect to all permutations of letters $D_1, D_2, \ldots, D_j$.

Let us denote by $\mathcal{C}$ the set of all nonterminal cuts of the sentential forms in G. Let $\mathcal{C}_r$ be the set of elements of $\mathcal{C}$ of length at most r. By the length of the nonterminal cut of $\omega \in (N \cup T)^*$, we mean the number of nonterminals in $c^{(nt)}(\omega)$.

We say that nonterminal cut $c'_1 = [D_1 \ldots D_j]$ yields nonterminal cut $c'_2 = [B_1 \ldots B_l]$, $1 \le j \le r$, $0 \le l \le r$, through the use of rule $p \in P$, denoted by $c'_1 \mapsto_p c'_2$, if there are two sentential forms ω_1 and ω_2 in G such that $c^{(nt)}(\omega_1) = c'_1$, $c^{(nt)}(\omega_2) = c'_2$ and if we apply rule p to ω_1, then $\omega_1 \Longrightarrow_G \omega_2$ holds. We can determine the set of rules $P'_{c'_1,c'_2} \subseteq P$ for arbitrary two nonterminal cuts $c'_1 = [D_1 \ldots D_j]$ and $c'_2 = [B_1 \ldots B_l]$, where $1 \le j \le r$, $0 \le l \le r$, such that for any $p \in P'_{c'_1,c'_2}$, $c'_1 \mapsto_p c'_2$ holds. (Notice that this rule set can be empty.)

Let $\mathcal{D} = \{(c'_1, c'_2, p) \mid c'_1 \mapsto_p c'_2, p \in P, c'_1, c'_2 \in \mathcal{C}_r\}$. Since the number of nonterminals in the nonterminal cuts is bounded by r and the number of productions in G is a finite set, $\mathcal{D}$ is a finite set, as well.

Observe that if $S \Longrightarrow_{p_{i_1}} \omega_1 \Longrightarrow_{p_{i_2}} \omega_2 \Longrightarrow_{p_{i_3}} \ldots \Longrightarrow_{p_{i_z}} \omega_z = \omega, z \ge 1$, is a derivation in G such that there are at most r nonterminal symbols in $\omega_q, 1 \le q \le z$, then $c^{(nt)}(S) \mapsto_{p_{i_1}} c^{(nt)}(\omega_1) \mapsto_{p_{i_2}} c^{(nt)}(\omega_2) \mapsto_{p_{i_3}} \ldots \mapsto_{p_{i_z}} c^{(nt)}(\omega_z) = c^{(nt)}(\omega)$ holds. Furthermore, $c^{(nt)}(S) \mapsto_{p_{i_1}} c^{(nt)}(\omega_1) \mapsto_{p_{i_2}} c^{(nt)}(\omega_2) \mapsto_{p_{i_3}} \ldots \mapsto_{p_{i_z}} c^{(nt)}(\omega_z) = c^{(nt)}(\omega)$ may belong to several derivations in G. It signifies that starting from S and applying the rules $p_{i_1}, \ldots, p_{i_z}$ in this order to the corresponding sentential forms, we obtain all derivations $S \Longrightarrow_{p_{i_1}} u_1 \Longrightarrow_{p_{i_2}} u_2 \Longrightarrow_{p_{i_3}} \ldots \Longrightarrow_{p_{i_z}} u_z = u$ in G, where

$c^{(nt)}(u_j) = c^{(nt)}(\omega_j)$, $1 \le j \le z$, holds.

In order to prove that $\mathcal{L}_{fin}(\text{USC}) \subseteq \mathcal{L}(\text{FEG}^{time}_{PR_{ac}})$, we construct $\Gamma \in \text{FEG}^{time}_{PR_{ac}}$ such that for each triplet (c_1', c_2', p) in $\mathcal{D}$ the foragers of Γ indicate how the the nonterminals will be replaced, if $c_1' \mapsto_p c_2'$, and only that. These foragers may identify not only the nonterminals rewritten by p in the sentential form, but also (depending on p) those nonterminals that remain unaltered. To complete the simulation of the application of p, the environment substitutes (some of) the nonterminals and may perform other kinds of rewritings. The idea is that every derivation $S \Longrightarrow_{p_{i_1}} \omega_1 \Longrightarrow_{p_{i_2}} \omega_2 \Longrightarrow_{p_{i_3}} \dots \Longrightarrow_{p_{i_z}} \omega_z = \omega$ in G corresponds to a computation $U_{i_0} M_{i_0}(t_{i_0}) S(t_0) \Longrightarrow^* U_{i_1} M_{i_1}(t_{i_1}) \omega_1(t_1) \Longrightarrow^* U_{i_2} M_{i_2}(t_{i_2}) \omega_2(t_2) \Longrightarrow^* \dots \Longrightarrow^* U_{i_{z-1}} M_{i_{z-1}}(t_{i_{z-1}}) \omega_{i_{z-1}}(t_{z-1}) \Longrightarrow^* U_{i_z} \omega_z(t_z) = U_{i_z} \omega(t_z)$ in Γ and vice versa, where $M_{i_h}(t_{i_h})$, $t_{i_h} \in \mathbb{N}$, $0 \le h \le z-1$, is the starting nonterminal of the forager commencing the simulation of the application of production p_{i_h} to $w_{z_{i_{h-1}}}$ and $t = r+1$ (r is the index of G).

Now we define the components of Γ. As in the proof of Theorem 1, we present the foragers and the environmental tables with which these foragers cooperate during the simulation.

Let $card(\mathcal{D}) = m$ and let $M_k = (c_1', c_2', p) \in \mathcal{D}$, $1 \le k \le m$, where nonterminal cut $c_1' = [D_1 \dots D_j]$ yields nonterminal cut $c_2' = [B_1 \dots B_l]$, $1 \le j \le r$, $0 \le l \le r$, if we use rule $p \in P$.

For M_k, $k \in \{i_2, \dots, i_z\}$, we construct foragers $A_{k,i}$, $1 \le i \le j$.

Let $A_{k,i} = (N_{k,i} \cup N^{(k,i)}_{k,i}, S_{k,i}, R_{k,i})$, where $N_{k,i} = \{M_{k,i,0}(t), D_i(t') \mid 1 \le t \le t_{k,i,0}, 1 \le t' \le t_{k,i,1}\}$, $N^{(k,i)}_{k,i} = \{M^{(k,i)}_{k,i,0}(t), D^{(k,i)}_i(t') \mid 1 \le t \le t_{k,i,0}, 1 \le t' \le t_{k,i,1}\}$, and $S_{k,i} = M_{k,i,0}(t_{k,i,0})$, $t_{k,i,0} \ge 1$. The rule set of forager $A_{k,i}$, i.e. $R_{k,i}$, $1 \le i \le j$, is as follows:

- $(l_{k,i,0}(t) : M_{k,i,0}(t) \to M^{(k,i)}_{k,i,0}(t-1), l_{k,i,1}, l_{k,i,0})$, $1 \le t \le t_{k,i,0}$,
- $(l_{k,i,1}(t) : D_i(t) \to D^{(k,i)}_i(t-1), l_{k,i,0}, h_{k,i,1})$, $1 \le t \le t_{k,i,1}$,
- $(h_{k,i,1}(t) : D^{(k,i)}_i(t) \to D^{(k,i)}_i(t-1), l_{k,i,1}, h_{k,i,1})$, $1 \le t \le t_{k,i,1}$, where
  - $l_{k,i,0} = \{l_{k,i,0}(t) \mid 1 \le t \le t_{k,i,0}\}$, $l_{k,i,1} = \{l_{k,i,1}(t) \mid 1 \le t \le t_{k,i,1}\}$, $h_{k,i,1} = \{h_{k,i,1}(t) \mid 1 \le t \le t_{k,i,1}\}$.

For forager $A_{k,i}$, $1 \le i \le j-1$, simulating derivation $U_{k,i,0} M_{k,i,0}(t_{M_{k,i,0}}) \omega(t_\omega) \Longrightarrow^* U_{k,i+1,0} M_{k,i+1,0}(t_{M_{k,i+1,0}}) \omega'(t_{\omega'})$, we will construct the environmental tables. For technical reasons, let $P_E^{\{a,F\}} = \{a \to a \mid a \in T_E\} \cup \{F \to F\}$.

The first environmental table $P_{k,i,0}$, $1 \le i \le j-1$, is of the form below:

$$\begin{aligned} P_{k,i,0} = &\{U_{k,i,0} \to U_{k,i,1}, U' \to F \mid U' \in V_M \setminus \{U_{k,i,0}\}\} \cup \\ &\{C(t) \to C(t-1), C(0) \to F \mid C \in V_E \setminus (V_M \cup T_E), t \in \mathbb{N}\} \cup P_E^{\{a,F\}}. \end{aligned}$$

By the joint action of forager $A_{k,i}$, $1 \leq i \leq j-1$, $(M_{k,i,0}(t) \to M^{(k,i)}_{k,i,0}(t-1)$, $t \in \mathbb{N})$ and the environmental table, $P_{k,i,0}$, we obtain $U_{k,i,0}M_{k,i,0}(t_{M_{k,i,0}})\omega(t_\omega) \Longrightarrow U_{k,i,1}M^{(k,i)}_{k,i,0}(t_{M_{k,i,0}}-1)\omega(t_\omega-1)$.

At the second step, forager $A_{k,i}$, $1 \leq i \leq j-1$, has to check whether D_i is in the string. The second environmental table has the form:

$$\begin{aligned}P_{k,i,1} =&\{U_{k,i,1} \to U_{k,i+1,0}, U' \to F \mid U' \in V_M \setminus \{U_{k,i,1}\}\}\cup\\ &\{C(t) \to C(t-1), C(0) \to F \mid C \in V_E \setminus (V_M \cup T_E \cup \{M^{(k,i)}_{k,i,0}(t'-1)\}), t, t' \in \mathbb{N}\}\\ &\cup \{M^{(k,i)}_{k,i,0}(t) \to M_{k,i+1,0}(t'), M^{(k,i)}_{k,i,0}(0) \to F \mid t, t' \in \mathbb{N}\} \cup P_E^{\{a,F\}}.\end{aligned}$$

At the second step, forager $A_{k,i}$, $1 \leq i \leq j-1$, performs rule $D_i(t) \to D_i^{(k,i)}(t-1)$, $t \in \mathbb{N}$. As a consequence of the parallel action of the forager and the environment, we receive the word $U_{k,i,1}M^{(k,i)}_{k,i,0}(t_{M_{k,i,0}}-1)\omega(t_\omega-1) \Longrightarrow U_{k,i+1,0}M_{k,i+1,0}(t_{M_{k,i+1,0}})\omega'(t_\omega-2)D_i^{(k,i)}(t_\omega-2)\omega''(t_\omega-2)$.

If $i = j$, then the first step of the simulation is analogous to the case when $A_{k,i}$, $1 \leq i \leq j-1$, acts in parallel with environmental table $P_{k,i,0}$. The second environmental table, $P_{k,j,1}$, should be modified as follows:

$$\begin{aligned}P_{k,j,1} =&\{U_{k,j,1} \to \hat{U}_{k',1,0}, U' \to F \mid U' \in V_M \setminus \{U_{k,j,1}\}, k' \in \{i_2, \ldots, i_z\}, k' \neq k\}\cup\\ &\{C(t) \to C(t-1), C(0) \to F \mid C \in V_E \setminus (V_M \cup T_E \cup \{M^{(k,j)}_{k,j,0}(t'-1)\}), t, t' \in \mathbb{N}\}\\ &\cup \{M^{(k,j)}_{k,j,0}(t) \to \lambda, M^{(k,j)}_{k,j,0}(0) \to F \mid t \in \mathbb{N}\} \cup P_E^{\{a,F\}}.\end{aligned}$$

After the second step through the parallel action of forager $A_{k,j}$, $(D_j(t) \to D_j^{(k,j)}(t-1)$, $t \in \mathbb{N})$ and environmental table, $P_{k,j,1}$, we obtain the word $U_{k,i,1}M^{(k,j)}_{k,j,0}(t_{M_{k,j,0}}-1)\omega(t_\omega-1) \Longrightarrow \hat{U}_{k',1,0}\omega'(t_\omega-2)D_j^{(k,j)}(t_\omega-2)\omega''(t_\omega-2)$.

After forager $A_{k,j}$ has finished the substitution of $D_j(t)$ for $D_j^{(k,j)}(t-1)$, $t \in \mathbb{N}$, and the corresponding environmental has replaced all the other symbols, then only the environment is allowed to continue the derivation due to the lack of symbols M_k. At this step, the environment replaces the nonterminals of the nonterminal cut present at the environment word and identified by the foragers and it either starts the simulation of the next valid configuration transmission or finishes the derivation. The environmental table can be defined as follows:

$$\begin{aligned}P_{k'} =&\{\hat{U}_{k',1,0} \to U_{k',1,0}M_{k',1,0}(t), \hat{U}_{k',1,0} \to \lambda \mid t \in \mathbb{N}\}\cup\\ &\{U' \to F \mid U' \in V_M \setminus \{\hat{U}_{k',1,0}\}\}\cup\\ &\{C(t) \to C(t-1) \mid C \in V_E \setminus (V_M \cup T_E \cup \{D_1^{(k,1)}(t_1), \ldots, D_j^{(k,j)}(t_j)\}),\\ &t, t_1, \ldots, t_j \in \mathbb{N}\} \cup \{C(0) \to F \mid C \in V_E \setminus (V_M \cup T_E \cup \{D_1^{(k,1)}(t_1), \ldots,\\ &D_j^{(k,j)}(t_j)\}), t, t_1, \ldots, t_j \in \mathbb{N}\} \cup \{D_1^{(k,1)}(t_1) \to \alpha_1, \ldots, D_j^{(k,j)}(t_j) \to\\ &\alpha_j \mid \alpha_1, \ldots, \alpha_j \in (V_E \setminus V_M)^*, t_1, \ldots, t_j \in \mathbb{N}\} \cup P_E^{\{a,F\}}.\end{aligned}$$

As a result of the employment of $P_{k'}$, the environmental word may have either the form $\hat{U}_{k',1,0}\omega'(t_{\omega'})D_1^{(k,1)}(t_1)\dots D_j^{(k,j)}(t_j)\omega''(t_{\omega''}) \Longrightarrow U_{k',1,0}M_{k',1,0}(t)\omega'(t_{\omega'}-1)\alpha_1\dots\alpha_j\omega''(t_{\omega''}-1)$ or the form $\hat{U}_{k',1,0}\omega'(t_{\omega'})D_1^{(k,1)}(t_1)\dots D_j^{(k,j)}(t_j)\omega''(t_{\omega''}) \Longrightarrow \omega'(t_{\omega'}-1)\alpha_1\dots\alpha_j\omega''(t_{\omega''}-1)$.

It only remains to be shown how the simulation begins. The initial state of the web environment is $U_{i_0}M_{i_0}(t_{i_0})S(t_0)$. We have to simulate rule $S \to \omega_1 \in P$. Let us assume that the web environment has a table $P_{E_{start}}$ that performs this simulation, where

$$\begin{aligned} P_{E_{start}} = &\{S(t_0) \to \omega_1(t_1) \mid t_0, t_1 \in \mathbb{N}\} \cup \{U_{i_0} \to U_{i_1}, U' \to F \mid U' \in V_M \setminus \{U_{i_0}\}\} \cup \\ &\{M_{i_0}(t_{i_0}) \to M_{i_1}(t_{i_1}) \mid t_{i_0}, t_{i_1} \in \mathbb{N}\} \cup \\ &\{C(t) \to F \mid C \in V_E \setminus (V_M \cup T_E \cup \{M_{i_0}(t_{i_0})\}), t, t_{i_0} \in \mathbb{N}\} \cup P_E^{\{a,F\}}. \end{aligned}$$

Initially, all the foragers are inactive, since there is not a $M_{i_h}, 1 \le h \le z$, in the sentential form. The derivation is as follows: $U_{i_0}M_{i_0}(t_{i_0})S(t_0) \Longrightarrow U_{i_1}M_{i_1}(t_{i_1})\omega_1(t_1)$. Since S does not occur in any other rule of G, this is the only way how the simulation can begin.

During the simulation the sequences $(c_1', c_2', p) \in \mathcal{D}$ determine the valid configuration transmissions. Since the agents belonging to a nonterminal cut rewrite all of it nonterminals and only those, therefore it can be concluded that we can simulate all of the appropriate derivations in G. If the environmental word contains only terminals, then it is the element of the generated language. Hence the inclusion $\mathcal{L}_{fin}(\mathrm{USC}) \subseteq \mathcal{L}(\mathrm{FEG}_{PR_{ac}}^{time})$ is verified.

Secondly, we will prove the other inclusion, i.e. $\mathcal{L}_{fin}(\mathrm{USC}) \supseteq \mathcal{L}(\mathrm{FEG}_{PR_{ac}}^{time})$.

Let

$$\Gamma = (E, A_1, \dots, A_n, c_{init})$$

be an $\mathrm{FEG}_{PR_{ac}}^{time}$ system, defined as in Definition 7. To prove the statement, we will construct an unordered scattered context grammar $G = (N, T_E, S, P)$ of finite index such that $L(\Gamma) = L(G)$ holds. The proof is based on simulating the derivations in Γ with derivations of finite index in G.

In the sequel, we make some remarks on how Γ works. Let $c = (k_1, \dots, k_n; u_1D_{h_1}(t_1)u_2D_{h_2}(t_2)\dots u_jD_{h_j}(t_j)u_{j+1})$ be a configuration of Γ, where $t_j \in \mathbb{N}_0$, $j \ge 1$, $D_{h_m} \in V_N \cup \bar{V}_N$, $1 \le m \le j$, $u_l \in T_E^*, 1 \le l \le j+1$, and $k_1, \dots, k_n$ are the labels of the rules to be applied by the agents. For the sake of legibility, we also refer to the elements of $V_N \cup \bar{V}_N$ as the nonterminals of Γ. Observe that there must be a number s, $s \in \mathbb{N}$, such that if there are more than s nonterminals in the environmental state $\omega = u_1D_{h_1}(t_1)u_2D_{h_2}(t_2)\dots u_jD_{h_j}(t_j)u_{j+1}$, i.e. $j > s$, then after a certain number of derivation steps, some of the t_js will become 0. This statement can be explained by the fact that if there are n agents in the eco–foraging system and t is the maximum of the lifetimes associated with the nonterminals of Γ, and there are $n \cdot t + 1$ such nonterminals in the environmental state v of some configuration, then after performing $t+1$ derivation steps on environmental state v, we will obtain an environmental state containing at least one nonterminal whose lifetime is equal to 0.

Owing to the fact that it is not possible to remove the nonterminal with lifetime 0 from the strings, every derivation in Γ that results in a word over T_E cannot contain a configuration in which the environmental state has a nonterminal with lifetime 0. Consequently, when we simulate the derivations of Γ with derivations of G, it is enough to consider derivations with configurations of the forms $c = (k_1, \ldots, k_n; u_1 D_{h_1}(t_1) u_2 D_{h_2}(t_2) \ldots u_j D_{h_j}(t_j) u_{j+1})$ only, where D_{h_m} is a nonterminal, $1 \leq m \leq j$,, $u_r \in T_E^*$, $t_j \in \mathbb{N}$, $1 \leq r \leq j+1$, and $j \leq s$. Let us call $c^{(nt)}(c) = [D_{h_1}(t_1) D_{h_2}(t_2) \ldots D_{h_j}(t_j)]$ the nonterminal cut of c.

Since $s \in \mathbb{N}$ and the set of nonterminals as well as the set of rules of the agents are finite sets, we can determine the set of all valid configuration transmissions. The configuration transmission from c_1' to c_2' is valid, provided that the lengths of their nonterminal cut are less than s and there is no nonterminal in neither of the nonterminal cuts with lifetime 0.

Let us label elements of $\mathcal{P}_E$ by elements of $Label(\mathcal{P}_E)$ and let us suppose that $Label(\mathcal{P}_E)$ and $Label(R_i)$, $1 \leq i \leq n$, are pairwise disjoint sets.

We say that nonterminal cut $c_1' = [D_1 \ldots D_j]$ yields nonterminal cut $c_2' = [B_1 \ldots B_l]$, $1 \leq j \leq s$, $0 \leq l \leq s$, through the use of the sequence of rules labelled by $\bar{k}_1, \ldots, \bar{k}_j$, where $\bar{k}_1, \ldots, \bar{k}_j \in Label(\mathcal{P}_E) \cup \bigcup_{i=1}^n Label(R_i)$ denoted by $c_1' \mapsto_{(\bar{k}_1, \ldots, \bar{k}_j)} c_2'$, if there are two configurations c_1 and c_2 in Γ such that $c^{(nt)}(c_1) = c_1'$, $c^{(nt)}(c_2) = c_2'$, and if we apply the rule sequence labelled by $\bar{k}_1, \ldots, \bar{k}_j$ to c_1, then $c_1 \Longrightarrow_\Gamma c_2$ holds. We can determine the labels of the rules of all such possible rule sequences.

Analogously to the first part of the proof of the theorem, we can construct the finite set $\mathcal{D} = \{[(k_1, \ldots, k_n); (\bar{k}_1, \ldots, \bar{k}_j); (c_1', c_2'); (k_1', \ldots, k_n')] \mid c_1' = c^{(nt)}(c_1), c_2' = c^{(nt)}(c_2), c_1' \mapsto_{(\bar{k}_1, \ldots, \bar{k}_j)} c_2', \bar{k}_h \in Label(\mathcal{P}_E) \cup \bigcup_{i=1}^n Label(R_i), 1 \leq h \leq j, k_v, k_v' \in Label(R_v)$, where k_v is the current label of R_v, and k_v' is the next label of $R_v, 1 \leq v \leq n\}$. Note we can employ this construction only because the terminal symbols are not altered and the number of nonterminals in each environmental state we consider is limited by a constant. Furthermore, since we know the current labels of the rules the agents apply, we can calculate the new labels of the rules of the agents, as well.

Based on the above observations, G have the rules of the following form:

- $(S \to M_{init} \omega_{init})$, where ω_{init} is the axiom of Γ, $M_{init} = [(\emptyset, \ldots, \emptyset); (\emptyset, \ldots, \emptyset); (\emptyset, c_{init}); (k_1, \ldots, k_n)]$, the eco–grammar system has not started to work in the beginning, we denote this fact by the dummy symbol $(\emptyset)$, which refers to the empty label of the agents and the environment as well as the empty configuration, c_{init} is the nonterminal cut of the initial configuration, $k_1, \ldots, k_n$ are the initial labels of the agents,

- $(M \to M', D_{h_1}(t_1) \to \alpha_1, \ldots, D_{h_j}(t_j) \to \alpha_j)$, where $M, M' \in \mathcal{D}$,
  $M = [(k_1, \ldots, k_n); (\bar{k}_1, \ldots, \bar{k}_j); (c_1', c_2'); (k_1', \ldots, k_n')]$,
  $M = [(k_1', \ldots, k_n'); (\bar{k}'_1, \ldots, \bar{k}'_l); (c_2', c_3'); (k_1'', \ldots, k_n'')]$,
  and k_m is the label of $D_{h_m}(t_1) \to \alpha_m$, $1 \leq m \leq j$, $1 \leq j \leq s$, $0 \leq l \leq s$,

- $(M \to M_{fin}, a \to a)$, where $a \in T_E$,

- $(M_{fin} \to \lambda)$.

In this way, we can simulate the configuration transmissions in Γ. All the valid configurations can be received and no other configurations can be obtained. Due to the fact that the rules above follow the order of the configuration transmissions, G generates the same words as Γ does. Hence the theorem is proved. □

4. Discussion

In this paper we presented an approach to the behaviour of Internet crawlers seeking novel information on the World Wide Web. We employed programmed grammars, or more precisely, programmed grammar schemes, for the description of the behaviour of the crawlers. We proved that if we ignore the aging of the web environment in the model, then through the simulation of certain normal form grammars, we can obtain the class of recursively enumerable languages. It means that the crawlers communicating only through the environment are able to identify any computable set of the environmental states. If we assume, however, that the web pages may become obsolete, then we can produce the language family generated by unordered scattered context grammars of finite index.

Our aim was to illustrate the great diversity of employing regulated rewriting devices in the field of web crawling techniques. Besides programmed grammars, there are other regulated rewriting devices that prescribe the sequences of productions or determine the dependence of the rule on the history of the derivation [7], [23]. The idea of prescribing the sequences of productions or determining the dependence of the rule on the history of the derivation can correspond to the utilization of some kind of ordering in the case of URLs. Grammars that impose some global context condition on the employment of productions can be suitable candidates for capturing the behaviour of focused/topic specific/topical crawlers. Contrary to grammars with prescribed sequences or with dependence of the rule on the history of the derivation, these regulated rewriting mechanisms do not determine the sequence of applicable productions in advance, since it is controlled by the generated sequence of the sentential forms, i.e. by the environmental string produced through the joint actions of the foragers and the web environment. Crawlers may seek information either on a single (identical) topic, or on different ones [12]. The idea of information harvest on similar topics can be expressed by regulated rewriting mechanisms employing some sort of parallelism. Some regulated rewriting mechanism may also be combined with another. In this way, several different aspects of search strategies can be formalized and more sophisticated techniques can be realized. Furthermore, certain regulated rewriting devices may be incorporated into Lindermayer systems.

Since the objective of the paper was to introduce a language theoretic approach to the information retrieval of Internet crawlers, the experimental evaluation (see [12]) of the theoretical aspects lies beyond the scope of this work, though, the mathematical results may open up new directions in the development of further applications. In addition to the lifetime constraint, other restrictions may be placed on eco–foraging systems. The investigation of the characteristics as well as the generative power of eco–grammar systems consisting of regulated rewriting devices given the various limitations are the subjects of further research.

Acknowledgements. The authors express their gratitude to the reviewers for their useful comments and suggestions.

References

[1] Barabási A. L., Albert R., Jeong H., *Scale–Free Characteristics of Random Networks: the Topology of the* World Wide Web, Physica A, vol. **281**, pp. 69–77, 2000.

[2] Chakrabarti S., van der Berg M., Dom B., *Focused Crawling: a New Approach to Topic–Specific Web Resource Discovery*, Computer Networks, vol. **31**, no. 11–16, pp. 1623–1640, 1999.

[3] Cho J., Garcia–Molina H., *Effective Page Refresh Policies for Web Crawlers, ACM Transactions on Database Systems*, vol. **28**, no. 4, pp. 390–426, 2003.

[4] Csuhaj–Varjú E., *Eco–Grammar Systems: Recent Results and Perspectives*, in G. Păun (ed.) *Artificial Life: Grammatical Models*, pp. 79–103, Black Sea University Press, Bucharest, 1995.

[5] Csuhaj–Varjú E., Dassow J., Kelemen J., Păun G., *Grammar systems. A Grammatical Approach on Distribution and Cooperation*, Gordon and Breach, London, 1994.

[6] Csuhaj–Varjú E., Kelemen J., Kelemenová A., Păun G., *Eco–Grammar Systems: A Grammatical Framework for Studying Lifelike Interactions*, Artificial Life, vol. **3**, pp. 1–28, 1997.

[7] Dassow J., Păun G., *Regulated Rewriting in Formal Language Theory*, Springer–Verlag, New York, 1996.

[8] Diligenti M., Coetzee F., Lawrence S., Lee Giles C., Gori M., *Focused Crawling Using Context Graphs*, Proceedings of the 26th International Conference on Very Large Databases, Morgan Kaufmann, San Francisco, CA, USA, 2000.

[9] Haykin S., *Neural Networks: A Comprehensive Foundation*, Prentice Hall, New Jersey, USA, 1999.

[10] Kelemen J., Kelemenová A., *A Grammar Theoretic Treatment of Multiagent Systems, Cybernetics and Systems. An International Journal*, vol. **23**, no. 6, pp. 621–633, 1992.

[11] Lázár K., *A Language Theoretic Approach to the Crawlers' Problem*, Proceedings of the 5^{th} International Workshop on Emergent Synthesis (IWES'04), Kanji Ueda, László Monostori, András Márkus (eds.), pp. 19–26, 2004.

[12] Lőrincz A., Lázár K. A., Palotai Z., *Efficiency of Goal–Oriented Communicating Agents in Different Graph Topologies: a Study with* Internet *Crawlers*, Physica A, vol. **378**, no. 1, pp. 127–134, 2007.

[13] MacKay D. J. C., *Information Theory, Inference and Learning Algorithms*, Cambridge University Press, Cambridge, UK, 2003.

[14] Marchioria M., Latora V., *Harmony in the Small–World*, Physica A, vol. **285**, pp. 39–56, 2000.

[15] McCallum A., Nigam K., Rennie J., Seymore K., *A Machine Learning Approach to Building Domain–Specific Search Engines*, Proceedings of the 16th International Joint Conference on Artificial Intelligence, 1999, pp. 662–667.

[16] McCallum A., Nigam K., Rennie J., Seymore K., *Automating the Construction of* I*nternet Portals with Machine Learning, Information Retrieval*, vol. **3**, pp. 127–163, 2000.

[17] Menzer F., Pant G., Srinivasan P., *Topical Web Crawlers: Evaluating Adaptive Algorithms*, ACM Transactions on Internet Technology, vol. **4**, no. 4, pp. 378–419, 2004.

[18] Pant G., Srinivasan P., *Link Contexts in Classifier–Guided Topical Crawlers*, IEEE Transactions on Knowledge and Data Engineering, vol. **18**, no. 1, pp. 107–122, January 2006.

[19] Pant G., Srinivasan P., Menzer F., *Exploration versus Exploitation in Topic Driven Crawlers*, Proceedings of the Second International Workshop on Web Dynamics, 2002.

[20] Pinkerton B., *Finding what People Want: Experiences with the WebCrawler, First World Wide Web Conference*, 1994, Chicago, IL.

[21] Rennie J., McCallum A. K., *Using Reinforcement Learning to Spider the Web Efficiently*, Proceedings of ICML–99, 16th International Conference on Machine Learning, Morgan Kaufmann Publishers, San Francisco, US, Ivan Bratko, Saso Dzeroski (eds.), pp. 335–343, 1999.

[22] Rozenberg G., Salomaa A., *The Mathematical Theory of L Systems*, Academic Press, New York, 1980.

[23] Rozenberg G., Salomaa A. (eds.), *Handbook of Formal Language*s, Springer–Verlag, Berlin–Heidelberg, 1997.

[24] Salomaa A., *Formal Languages*, Academic Press, New York, 1973.

[25] Watts D. J., Strogatz S. H., *Collective Dynamics of Small–World Networks*, Nature, vol. **393**, pp. 440–44, 1998.