The Mechanism of Supervising the Evolution Process Realized in Multi Agent World

Krzysztof CETNAROWICZ [*]
e-mail: cetnar@agh.edu.pl

Rafal DREZEWSKI [†]
e-mail: drezew@agh.edu.pl

Abstract

It is supposed that there are only two main factors in the process of genetic search: population diversity and selective pressure. Many parameters that are used in order to control the process of genetic search indirectly influence population diversity and selective pressure. But such a control of the evolution process seems to be very unrealistic from the biological point of view. With the Evolutionary Multi Agent Systems (EMAS) it is possible to model biological mechanisms of species formation, rivalry and competition among species, social behavior and so on. In this paper we introduce the supervising mechanism based on two non-classical evolution operators: aggregation and escape. The application of these new evolution operators made it possible to construct the evolution centers. In the evolution centers it is possible to control values of various parameters of the environment in order to influence the course of the evolution process, to supervise this process and thereby to control the evolution process. Also, the application of this supervising mechanism to the construction of evolutionary multi-agent prediction system is presented. We discuss the results of simulation experiments carried out with the SWARM-based evolutionary prediction system, which show how the use of such supervising mechanism affects the modeled evolution process.

1 Introduction

Perhaps the most important and challenging part of research in the field of evolution programs is how to control the modeled evolution process. It is supposed that there are only two main factors in the process of genetic search: population diversity and selective pressure. Many parameters that are used in order to control the process of genetic search indirectly influence population diversity and selective pressure [8]. But such a control of the evolution process seems to be very unrealistic from the biological point of view. It is questionable if we should interfere in the natural process using artificial and unrealistic mechanisms. In order to supervise the modeled evolution process we should rather try to utilize possibilities provided by the biological evolution itself.

The technology of Evolutionary Multi Agent Systems (EMAS) [3], that arises as a result of realization of the evolution process in multi-agent world, can be treated as a new approach to the construction of evolution programs and a new way of developing multi-agent systems.

With the Evolutionary Multi Agent Systems it is possible to model biological mechanisms of species formation, rivalry and competition among species, social behavior and so on. It was not possible to model all these mechanisms in the case of classical evolution programs [6, 8].

With the use of these biological mechanisms we can supply the modeled evolution process with new impulses and try to direct it towards new and desirable (from our point of view) possibilities. In this paper we introduce the supervising mechanism based on two non-classical evolution operators: *aggregation* and *escape*. The aggregation operator enables emergence of social relations among agents. With the use of the escape operator agents can migrate towards different environments. Thus this operator enables modeling of the species formation process. The application of

[*] Institute of Computer Science, University of Mining and Metallurgy, Cracow, Poland
[†] Institute of Computer Science, University of Mining and Metallurgy, Cracow, Poland

these new evolution operators made it possible to construct the evolution centers. In evolution centers it is possible to control values of various environment parameters in order to influence the course of the evolution process, to supervise this process and thereby to control the evolution process.

2 Elements of theory

2.1 Model of evolving multi-profile agent

In evolutionary system, prediction agent is defined as a multi-profile agent a which contains two basic profiles. First profile (a_1) is in charge of realization of crossover and mutation algorithms. Second profile (a_2) realizes aggregation and escape operations.

In the presented model of agent, within the framework of a_1 profile, a set of agents is constructed as an image of observed world. These agents are potential candidates for the realization of crossover operation with particular agent. Preliminary selection of candidates and final choice is made on the basis of potential candidate's characteristic features. Agent's profile a_2 is in charge of realization of aggregation operation. Similarly to the profile a_1, also in this case a set of potential candidates to the realization of aggregation operation with particular agent is constructed on the basis of observation of the environment. The choice of candidate, similarly to the profile a_1, is made on the basis of set of candidate's features. The criterions of choice and set of features are different from those that are used in the profile a_1.

Agent's evaluation is made on the basis of two basic features: intellectual level and energetic states (the quantity of life energy).

The intellectual level determines the quality of prediction that is made by agent. It is defined as a percentage of correct predictions made in a fixed period of time. "Intellectual" state of an agent affects its attractiveness for other agents in decision making process concerning the choice of partners for evolution operations.

Every agent possesses a "life" energy reserve. All operations made by an agent, including evolutionary operations, require the use of some amount of agent's energy. In the process of realization of these operations, agent's energetic state p is reduced by some (fixed for particular operation) amount.

On the contrary, positive results of some actions supply agent with energy. Acting in the environment, agent executes operations and according to the results of these operations it can obtain or lose some fixed amounts of energy, increasing or decreasing the amount of its energy.

The final decision concerning the choice of one of possible actions is made by an agent with the use of decision function which takes into account the actual level of energy and intellectual level of particular agent and of an agent that particular agent is going to cooperate with. In this case, intellectual level and energetic level (with different weights) determine attractiveness of an agent — candidate to cooperation.

2.2 Aggregation and escape operations

The agent-environment relation is the basic reason that forces agents to participate in evolution process. If state of this relation is not satisfied for particular agent, it can choose one of the following actions:

- agent may change itself — adapting itself to the conditions of the environment with the use of mutation and crossover operations,

- agent may change the environment with, among others, the use of aggregation and escape operations.

The evolution process in which autonomous agent take part, may be realized with the use of following operators that affect single agent or fixed group of agents.

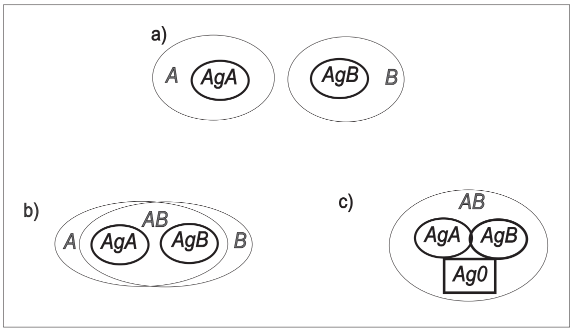

Figure 1: The principle of functioning of the aggregation operator

Mutation, crossover and reproduction operators in the agent environment have the same form as presented in [5, 6, 8] with such an exception that the evolution process of each agent has its own (individual for each agent) characteristics, it depends on agent's decisions, and it takes place according to its own independent cycle.

The idea of aggregation operation may be considered as a creation (by agents) of new environment in which agents act. A group of agents (such that conditions of actual environment do not suit them) make an agreement, which goal is to take over the control of the part of existing environment. Thus agents change the parameters of controlled environment, which we mentioned above, to make it better adapted to their requirements. After the creation of new environment, a group of agents, mentioned above, owing to specialization and cooperation, maintain desirable parameters in the environment created by them. This group (when it is seen from outside) act like a new agent with new characteristic features that arose owing to the aggregation operation. To sum up, the aggregation operator makes it possible to change the agents-environment relation with the change of environment's parameters.

The second operator that makes it possible to change the agents-environment relation with the change of environment's parameters is the escape operator. Let us make an assumption that the evolution process takes place in several environments, and that agents can migrate among these environments. The evolution processes that take place in each of these environments differ in some range of their parameters from each other. If in one of the mentioned environments the agent is created as a result of mutation, crossover or aggregation operation that is not well adapted to this environment, it may migrate towards different environment (with different characteristic parameters), where it can act better than in the previous environment. Then it may start there a new population of agents with valuable characteristic features.

2.3 Evolution centers as a method of evolution process control

The application of aggregation and escape operators makes it possible to control the evolution process, among others, through the mediation of organizing of, so called, evolution centers.

Let us consider the process of realization of aggregation operator (Fig. 1), which consists of the following stages:

1. Let two parameters characterize the particular environment: A and B (values of A and B belong to $\{0, 1\}$). There are agents AgA and AgB in this environment. Agent AgA (agent AgB) has the ability to influence its environment in such a way that it can maintain the value of parameter A (B) equal to one in its neighborhood. Agents A and B require (prefer) the values of both parameters (A and B) equal to one in their neighborhood.

2. Owing to the ability to move within the environment, agent AgA may stay near the agent AgB (and similarly agent AgB may stay near the agent AgA).

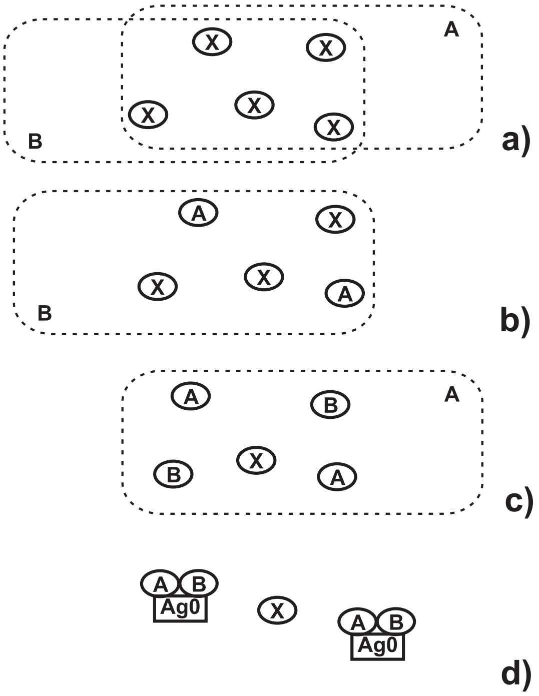

Figure 2: The principle of functioning of the evolution center

3. Agents AgA and AgB make an agreement and decide to aggregate together and create a new environment (with values of parameters satisfying requirements of both agents — $A = 1$ and $B = 1$) by taking over the control of the part of existing environment.

4. As a result of aggregation, the new agent $Ag0$ is created. This agent maintains the information about the configuration of agents A and B. The group of agents AgA, AgB, $Ag0$, and the part of environment that they control (with values of parameters A and B equal to one) constitute the new agent. Agent $Ag0$ keeps (for example encoded in its genes) information that is required to the reproduction of this aggregated agent.

With the use of aggregation and escape operators we can consider the organization of "evolution center", in which it is possible to control the values of environment parameters, what implies that we can supervise the course of the evolution process. Let us consider the example presented above, in which the realization of aggregation operator is completed with the application of remaining evolution operators and the control of environment parameters. Let us make an assumption that there exist such agents of type X in the environment that they can not set the environment parameters A and B in their neighborhood (i.e. set values $A = 1$ and $B = 1$). At the same time, agents of type X prefer the environment which parameters A and B are set ($A = 1$ and $B = 1$). Agent of type X may obtain (as a result of mutation and crossover operations) the ability to set the parameter A ($A = 1$) or B ($B = 1$) in its neighborhood (but not the ability to set both of them). Then it becomes the agent of type A or B. Let us make an assumption that the part of environment — the evolution center — is selected. We can control this evolution center from outside and change the values of parameters A and B (setting their values to 0 or 1). The example of supervising the evolution center may be considered as following steps:

- Parameters A and B are set ($A = 1$ and $B = 1$) in selected areas of evolution center. Agents of type X gather in this area (Fig. 2a).

- Parameter A is turned off periodically. This causes that some agents of type X obtain (as a result of mutation and crossover operations) the ability to set the parameter A ($A = 1$) in their neighborhood. Thus they became type A agents (Fig. 2b).

- After some time, the parameter A is turned on ($A = 1$), and parameter B is turned off ($B = 0$) periodically. This causes that some agents of type X obtain (as a result of mutation and crossover operations) the ability to set the parameter B ($B = 1$) in their neighborhood. Thus they became type B agents (Fig. 2c).
- After some time, both parameters are turned off ($A = 0$ and $B = 0$). Owing to the fact that agents of type A and agents of type B remain within the same area and they neighbor each other, they can form the aggregates AB (as a result of aggregation operation). These aggregates do not depend on the fact that parameters A and B are turned on or off, what implies that they can live in any environment (Fig. 2d).

Owing to the application of the idea of evolution centers (and the application of aggregation and escape operators) we may obtain the possibility of supervising the evolution process by changing the values of environment parameters, or rather, to say it more precise, by selecting the local sets of parameters.

3 The methodology of modeling

It is typical in the research work made with the use of computer simulation that we try to observe and investigate properties of phenomena, which adequate mathematical model cannot be formulated. We can only formulate models of subsystems (i.e. components of a system of interest) and define some basic interactions among them. But the relations that we are really interested in are nowhere explicitly encoded. They rather emerge and become accessible for observation as a result of interactions among the subsystems that we simulate.

The general research program of investigating complex dynamical phenomena using simulation can be expressed in two main steps [9]:

1. It is necessary to formulate lower-level models of the important underlying subsystems (those that define the elemental subsystems and the element-element or object-object interactions).
2. We must create the framework in which the simulation of the subsystems in interaction is composed, and embody the system representation in that framework so that the phenomena of interest can be generated and analyzed.

In the first part of research program we will use the model based on the concept of M-Agent architecture [1, 2, 4]. This model enables us to formalize the properties of the multi-agent system (environment, agents, agent-agent and agent-environment relations).

The second part of research made with the use of computer simulation requires appropriate programming tools. These tools should enable us to obtain reliable and repeatable results of our simulations. In our paper we propose the SWARM simulation system for this part of research. This system has all the features, which are indispensable in the EMAS and artificial life simulations.

3.1 The simulation tool: SWARM system

The SWARM simulation system has been created at the Santa Fe Institute [10, 11]. This system can be very useful for the researchers working in the field of computer simulations, especially in the area of multi-agent systems and artificial life. The main goal of its authors was to create such a simulation tool that the results obtained with the use of it would be reliable and repeatable. In order to achieve this goal SWARM has the following features:

1. Simulation writing is brought up to a higher level of expression. Applications are written with reference to a standard set of simulation tools.

2. The task of managing concurrency is made manageable. SWARM insulates the author of a simulation from all the computer science knowledge that is usually required to implement distributed and concurrent systems reliable. In addition SWARM forces experimenters to make their concurrency assumptions explicit.

SWARM is implemented in the Object-Oriented Programming language Objective C [7]. Computation in a SWARM application is made via objects sending messages to each other.

SWARM applications are structured around the concept of the "Swarm". Swarms are the basic building blocks of the SWARM simulation. A Swarm is combination of a collection of objects and a schedule of activity over those objects. The collections are like the matter of the Swarm and the schedule is like the arrow of time.

The core of every application is the "model swarm". It encapsulates the simulated model, i.e. agents, physical properties and structure of the space etc. In addition to the object collection, the model swarm also contains a schedule of activity over these objects. Model swarm consists of a set of inputs and outputs. The inputs are the model parameters and the outputs are the data, which are the result of the agents' activity.

Second very important part of the system is "observer swarm". The most important object in an observer swarm is the model swarm that is being studied. In addition observer swarm has a collection of objects (instrumentation), a schedule of activity and a set of inputs and outputs. With the use of this instrumentation we can observe our artificial world, collect data for future analysis etc. The inputs to the observer swarm are configurations of the observer tools. The outputs are the observations. The observer swarm can run in graphics mode or in batch mode. In batch mode we cannot interact with the simulation. The batch swarm reads the data from configuration file and writes the data to the other files for future analysis.

The SWARM system has large number of class libraries, which provides users tools that are indispensable in the process of creation of the simulation. Detailed description of all these libraries can be found in [11].

4 Evolutionary multi-agent prediction system with evolution centers

In this section we will describe sample application of the Evolutionary Multi-Agent Systems (EMAS) technology to the 0/1 time sequence prediction system [2, 3].

In such a system the main goal of the population of agents is to predict the changes of the environment. In the environment a parameter $\alpha \in \{0, 1\}$ is defined. Variations of the parameter α in discrete moments of time may be represented by the binary sequence $x(t)$, where $x(t)$ is the value of α in the time t. Value of the parameter α is available for all the agents acting in the environment. Each agent tries to predict the value that the α will take in the time $t + 1$. So the agent a_i generates the binary sequence y_i, such that $y_i(t) = \tilde{x}(t + 1)$, where $\tilde{x}(t + 1)$ is predicted value of parameter α in the time $t + 1$.

4.1 Structure of the environment

The space in which agents remain $E = (R, T)$, where R is a configuration of resources, and T is a topology of space [2].

Resources available for all the agents are parameter $\alpha = x(\tau)$, where $\tau = 0, \ldots, t$ and energy of the environment P_e. So, in the case of our system, $R = \{\alpha, P_e\}$. The topology T of the environment is graph. Every node of this graph has connections with its eight neighbors.

4.2 Internal structure of the agent

In the considered system the agent consists of one or more cells. The cell consists of the finite automaton, information about its age and statistics. Each cell makes its predictions with the use

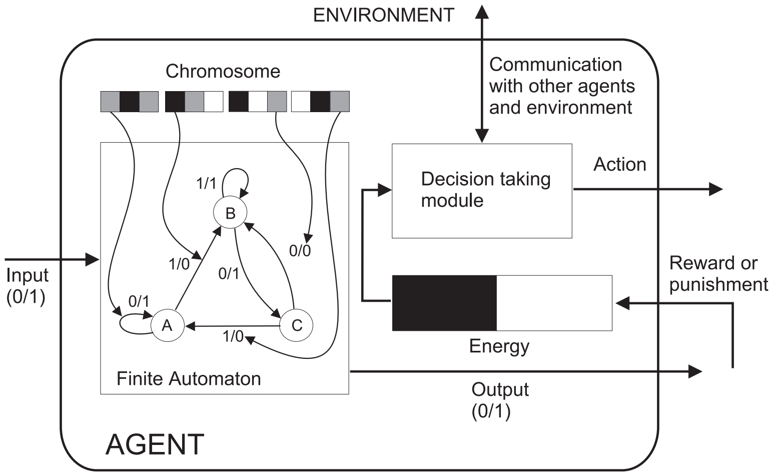

Figure 3: Internal structure of the agent (with one cell)

of finite automaton, which input/output language consists of symbols 0 and 1. The concept of predicting changes of the environment by evolving population of finite automatons is known as "evolutionary programming" [5, 6, 8]. The idea of the application of finite automaton as agent's chromosome is presented in [3]. Particular realization of this conception in system described in this paper is shown in the Figure 3.

There are four chromosomes that code the finite automaton:

- transitions between states when the input symbol is 0,
- transitions between states when the input symbol is 1,
- output symbols when the input symbol is 0,
- output symbols when the input symbol is 1.

Number of genes in each chromosome corresponds with the maximal number of states of the finite automaton (n^s_{max}).

There are two parameters that show the quality of prediction made by particular cell:

1. $\Psi^{S_j(0)}_{ik}$ — probability of correct prediction connected with the transition through 0 from the state S_j of the k-th cell of agent a_i;
2. $\Psi^{S_j(1)}_{ik}$ — probability of correct prediction connected with the transition through 1 from the state S_j of the k-th cell of agent a_i.

Two other parameters show the quality of prediction made by particular agent:

1. $\Psi_i(t)$ — probability of correct prediction made by agent a_i;
2. $\Psi^i_{max}(t)$ — maximal probability of correct prediction made by agent a_i during its lifetime.

4.3 Intellectual profile

An agent can be described from two points of view that correspond with two profiles: intellectual and energetic [2].

The model of the environment of agent a_i in the time t is

$$m_i(t) = \langle (Y_i(t), \Omega_i(t)), (Y_i(t+1), \Omega_i(t+1)), \dots \rangle, \tag{1}$$

where

$$Y_i(u) = \{y_{i1}(u), \ldots, y_{in}(u) : n \text{ is the number of cells}\}, \tag{2}$$

$$\Omega_i(u) = \left\{ \Psi_{i1}^{S_1^u(x_1(u))}, \ldots, \Psi_{in}^{S_n^u(x_n(u))} : n \text{ is the number of cells} \right\}, \tag{3}$$

for $u = t, t+1, \ldots$

$$y_{ik}(u) = t_{ik}^{S_k^u}(x_k(u)) \tag{4}$$

is the output connected with the transition of the k-th finite automaton through $x_k(u)$, S_k^u is the state of the k-th finite automaton in the time u,

$$x_k(u) = \begin{cases} x(u) & \text{for } u = t \\ y_{ik}(u-1) & \text{for } u = t+1, t+2, \ldots \end{cases} \tag{5}$$

for $k = 1, \ldots, n$, n is the number of cells of agent a_i.

After applying the strategy realization operator X [2] the agent's model of the environment is

$$m_i'(t) = \langle (y_i(t), \omega_i(t)), (y_i(t+1), \omega_i(t+1), \ldots) \rangle, \tag{6}$$

where $y_i(u) = y_{ik}(u)$ for $u = t, t+1, \ldots$, k is such that

$$\omega_i(u) = \Psi_{ik}^{S_k^u(z(u))} = \max \left\{ \Psi_{i1}^{S_1^u(z(u))}, \ldots, \Psi_{in}^{S_n^u(z(u))} : n \text{ is the number of cells} \right\} \tag{7}$$

for $u = t, t+1, \ldots$,

$$z(u) = \begin{cases} x(u) & \text{for } u = t \\ y_i(u-1) & \text{for } u = t+1, t+2, \ldots \end{cases} \tag{8}$$

The goal of the intellectual profile [2] is to make correct predictions:

$$q(m_i(t), m_i'(t)) = \begin{cases} 1 & \text{when } y_i(t) = x(t+1) \\ 0 & \text{when } y_i(t) \neq x(t+1) \end{cases} \tag{9}$$

Now the time is incremented and the observation operator I [2] generates the model

$$m_i''(t) = \langle (Y_i(t), \Omega_i(t)), (Y_i(t+1), \Omega_i(t+1)), \ldots \rangle, \tag{10}$$

by getting the value of $x(t)$ from the environment. This model is the initial model for next iteration. The adaptation operator $L = \{L_m\}$ [2] is consisted of all the evolution operators.

4.4 Energetic profile

Energetic profile of agent a_i [2] is represented by the parameter $P_i(t) \in \mathbb{R}$, $P_i(t) \in [0, P_{max}]$, P_{max} is the maximal level of agent's energy. The main goal of the agent in this profile is to maximize the value of $P_i(t)$. In order to realize this goal the energetic strategy s_p [2] is realized by applying one of the evolution operators. Every action (such as move or reproduction) costs some energy. Energy may be obtained only through correct prediction. All decisions of the agent (concerning reproduction, aggregation, and direction of the move) depend on actual level of its energy. The agent's energy also serves as a mechanism of elimination of the agents with low level of fitness to the environment conditions (i.e. making incorrect predictions) from the system. Agent a_i dies when $P_i(t) = 0$.

The evaluation of the energy of agent a_i during its lifetime may be expressed as follows:

$$P_i(t) = P_i(t_i^0) + \sum_{k=t_i^0+1}^{t} (\delta_i^p(k) + \delta_i^a(k)), \tag{11}$$

where t_i^0 is the time when agent a_i was born, $P_i(t_0^i)$ is the initial energy of agent a_i (the energy which agent gets from its parents),

$$\delta_i^p(k) = \begin{cases} \delta > 0 & \text{when } y_i(k-1) = x(k) \\ \delta < 0 & \text{otherwise,} \end{cases} \tag{12}$$

$$\delta_i^a(k) = \begin{cases} \delta > 0 & \text{when agent } a_i \text{ formed an aggregate with other agent} \\ 0 & \text{otherwise.} \end{cases} \tag{13}$$

The total energy of the system in time t:

$$P(t) = P_e(t) + \sum_{i=1}^{m(t)} P_i(t), \tag{14}$$

where $P_e(t)$ is the energy of environment, $P_i(t)$ is the energy of agent a_i, $m(t)$ is the number of agents that remain within the system in time t. The parameter $P(t)$ prevents excessive growth of the population of agents.

4.5 The evolution operators

4.5.1 The reproduction, mutation, and crossover operators

The reproduction of agent a_i in time t occurs with the probability p_r when

$$P_i^r(t) \le P_i(t) \tag{15}$$

where $P_i^r(t) \in [P_{min}^r, P_{max}]$ is pseudo-random number, P_{min}^r is the minimal level of energy required for reproduction operation. If agent a_i has more than one cell in time t then for reproduction will be chosen k-th cell, such that

$$\Psi_{ik}^{S_k^t(x(t))} = \max\left\{\Psi_{i1}^{S_1^t(x(t))}, \ldots, \Psi_{in}^{S_n^t(x(t))} \colon\ n \text{ is the number of cells}\right\} \tag{16}$$

Mutation operator may alter output symbol and transition between particular states. The mutation occurs with the probability p_m.

The crossover operation that mixes homologous chromosomes of two agents a_i and a_j occurs with the probability p_c when agent a_i (which initializes the process of reproduction) satisfies the equation (15) and

$$\exists\, a_j, \text{ such that } d(a_i, a_j) \le d_{max}^c \wedge P_j^c(t) \le P_j(t) \tag{17}$$

where $d(a_i, a_j)$ is the length of the shortest path between agents a_i and a_j in graph-like environment, d_{max}^c is the maximal length of the shortest path, $P_j^c(t) \in [P_{min}^c, P_{max}]$ is pseudo-random number, P_{min}^c is the minimal level of energy required for crossover operation.

4.5.2 The aggregation operator

The aggregation operator serves as a mechanism of emergence of social relations among agents. The idea of this operator is presented in [3]. In this section we present the realization of this idea in prediction system with evolution centers.

The agent consists of one or more cells:

$$a_i = \{c_1, \ldots, c_n \colon\ 1 \le n \le n_{max}^c\}, \tag{18}$$

where c_i is a cell, n_{max}^c is the maximal number of cells contained in the aggregate. Each cell can live from T_{min} to T_{max} units of time.

Two agents a_i and a_j form the aggregate in the time t with probability p_a if the following conditions are satisfied

$$d(a_i, a_j) \leq d^a_{max} \wedge n^c_i + n^c_j \leq n^c_{max} \wedge \Psi_i(t) \geq \Psi_j(t) \wedge \Psi^i_{max}(t) \leq \Psi^j_{max}(t) \tag{19}$$

where d^a_{max} is the maximal length of the shortest path, n^c_i and n^c_j are the numbers of cells of agent a_i and a_j, respectively.

The formed aggregate consists of cells of both agents, and the aggregate's energy (when agent a_i absorbs cells of agent a_j) is

$$P_i(t+1) = \begin{cases} P_i(t) + P_j(t) & \text{if } P_i(t) + P_j(t) < P_{max} \\ P_{max} & \text{otherwise.} \end{cases} \tag{20}$$

4.5.3 The escape operator

The escape operator enables agents to migrate towards different environments (i.e. where agents must predict values of the parameter α taken from different pseudo-random sequences). The existence of several environments causes that groups of agents are geographically separated, thus the process of species formation can take place in the population of agents.

If agent a_i remains within the environment E_j, which is controlled by agent a^{ec}_j, then the distance between them will change in the following way:

$$d_{ij}(t) = \begin{cases} d_{ij} < d_{ij}(t-1) & \text{if } y_i(t-1) = x_j(t) \\ d_{ij} > d_{ij}(t-1) & \text{if } y_i(t-1) \neq x_j(t). \end{cases} \tag{21}$$

If agent a_i moves outside the environment E_j (i.e. if $d_{ij}(t) > r_j$, r_j defines the size of environment E_j) then it will move randomly until its energy value P_i will be reduced to the level P^m_{min}. In such a situation agent will move towards the nearest agent a^{ec}_k $(k \neq j)$.

If more than one environment exist, agent a_i, which lives in the environment E_k, where $k > 1$, is forced to move towards the main environment E_1 (in which there are hardest living conditions) in time t when

$$\Psi_i(t) > S^{mig}_{min} \wedge P_i(t) > P^{mig}_{min} \wedge T_{mig}(t) \leq t_i \tag{22}$$

where S^{mig}_{min} is the minimal statistics required to migrate towards the environment E_1, P^{mig}_{min} is the minimal energy, t_i is the age of agent a_i $(t_i = t - t^0_i)$, $T_{mig}(t) \in [T_{min}, T_{max}]$ is pseudo-random number, T_{min} and T_{max} are, respectively, minimal and maximal possible age of cell.

Each move operation costs some energy $P_{mv}(t)$, where $P_{mv}(t) \in [P^{mv}_{min}, P^{mv}_{max}]$ is pseudo-random number, P^{mv}_{min} is minimal move energy, and P^{mv}_{max} is maximal move energy.

The escape operator made it possible to construct the evolution centers.

4.5.4 The evolution center

In our system the evolution centers are represented by agents a^{ec}_k, $k = 1, \ldots, n_{ec}$, where n_{ec} is the number of evolution centers in particular simulation. Agents a^{ec}_k are placed in some nodes of graph-like space. These agents are placed permanently and cannot move during the simulation.

Agent a^{ec}_j is responsible for generating values of parameter α_j in the environment E_j (i.e. for generating sequence $x_j(t)$), and for setting the conditions of evolution taking place in environment E_j.

Parameters of evolution center, represented by agent a^{ec}_j are:

$x_j(t)$ — binary pseudo-random sequence, which length is l_j,

r_j — defines the size of evolution center,

δ — amount of energy transferred to prediction agent as a prize for correct prediction,

p_r, p_m, p_c, p_a — probability of reproduction, mutation, crossover, and aggregation operations,

P^r_{min}, P^c_{min} — minimal level of energy required for reproduction and crossover operations,

d^c_{max}, d^a_{max} — maximal length of the shortest path between two agents when crossover and aggregation operations are allowed,

n^c_{max} — maximal number of cells contained in an aggregate,

T_{min}, T_{max} — minimal and maximal possible age of cell,

S^{mig}_{min} — minimal statistics required to migrate towards environment E_1,

P^{mig}_{min} — minimal energy required to migrate towards environment E_1,

P^{mv}_{min}, P^{mv}_{max} — minimal and maximal energy required for each move.

5 Simulation experiments

The main goal of the simulation experiments was to investigate whether the application of the evolution centers can improve the quality of prediction made by agents and how the use of such supervising mechanism affects the modeled evolution process. The intensity of mutation and crossover were not taken into consideration in this research.

In each experiment the maximal number of states of the finite automaton $n^s_{max} = 5$, and maximal energy of agent $P_{max} = 100$.

5.1 The configuration of evolution centers

5.1.1 The configuration of evolution center E_1

In the main evolution center (E_1) the sequence $x_1(t)$ consisted of sequences $x_2(t)$ and $x_3(t)$, each of them repeated 5 times alternately. The values of parameters were: $\delta = 2.5$, $p_r = 0.95$, $p_m = 0.05$, $p_c = 0.9$, $p_a = 0.9$, $P^r_{min} = 75$, $P^c_{min} = 60$, $d^c_{max} = 3$, $d^a_{max} = 3$, $n^c_{max} = 2$, $T_{min} = 300$, $T_{max} = 2000$, $S^{mig}_{min} = 1$, $P^{mig}_{min} = 100$, $P^{mv}_{min} = 0.5$, $P^{mv}_{max} = 1.5$.

5.1.2 The configuration of evolution center E_2

In evolution center E_2 the length of sequence $x_2(t)$ was $l_2 = 10$. This sequence was repeated periodically giving as a result infinite sequence. The values of parameters were $p_a = 0.0$ (i.e. the aggregation was not allowed), $S^{mig}_{min} = 0.8$, $P^{mig}_{min} = 50$. The values of other parameters were such as in the evolution center E_1.

5.1.3 The configuration of evolution center E_3

In evolution center E_3 the length of sequence $x_2(t)$ was $l_2 = 15$. This sequence was repeated periodically giving as a result infinite sequence. The values of other parameters were $\delta = 2.7$ (this value is greater than in the E_2 because the sequence is longer what implies that living conditions are harder), $p_a = 0.0$, $S^{mig}_{min} = 0.8$, $P^{mig}_{min} = 50$. The values of other parameters were such as in the evolution center E_1.

5.2 Types of experiments

5.2.1 Experiment with one evolution center

In this kind of experiment the population of agents lived in one evolution center (E_1) (Fig. 4). Two other evolution centers were out of action ($r_2 = 0$ and $r_3 = 0$). Agents could form aggregates consisted of two cells.

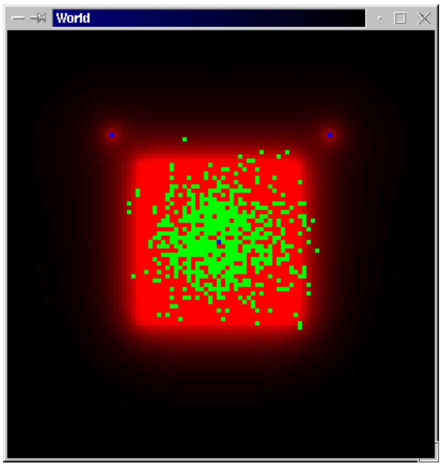

Figure 4: Experiment with one evolution center

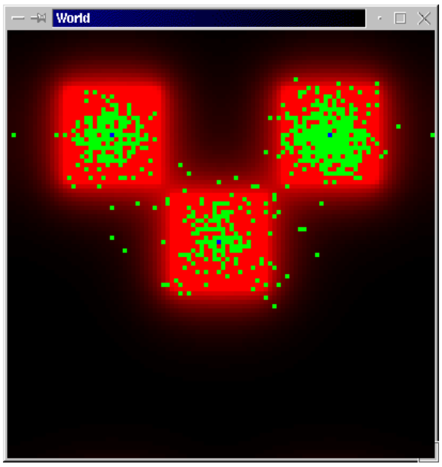

Figure 5: Experiment with three evolution centers

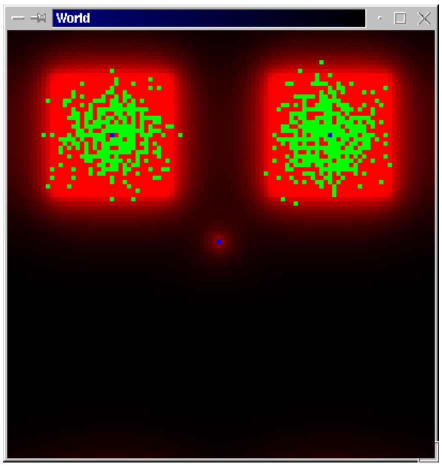

Figure 6: Experiment with two evolution centers

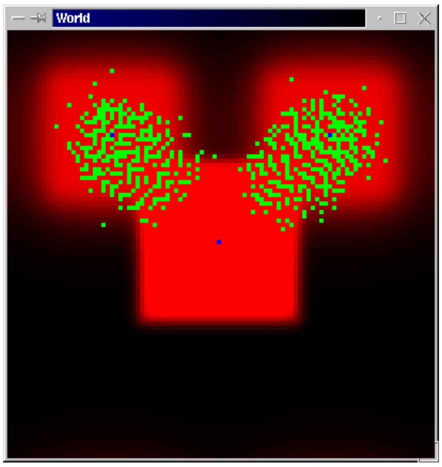

Figure 7: Population is migrating towards evolution center E_1

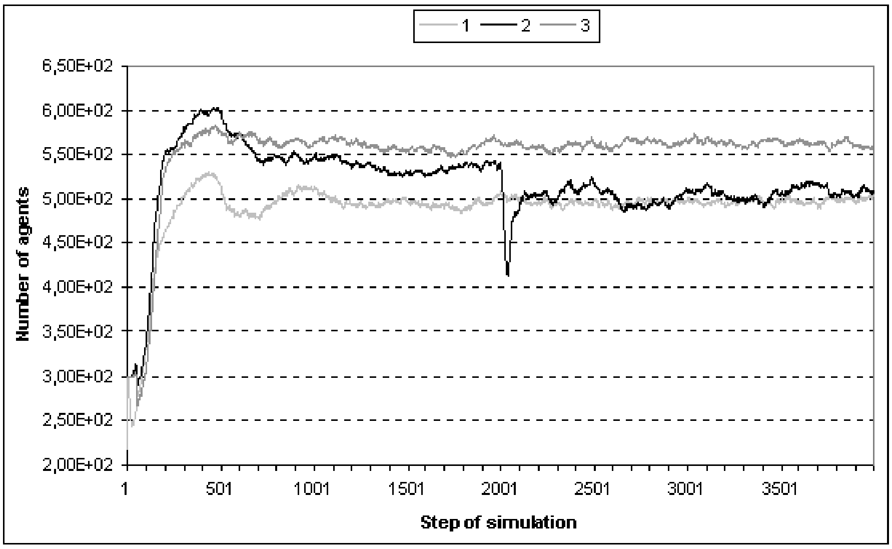

Figure 8: Number of agents in population

5.2.2 Experiment with two evolution centers

This experiment consisted of two parts. The first half of the simulation (2000 steps) was the process of species formation. Agents lived in E_2 and E_3 evolution centers (Fig. 6). In these evolution centers the aggregation was not allowed. Evolution center E_1 was out of action ($r_1 = 0$). Then, they were forced to move towards evolution center E_1 by setting parameters $r_2 = 0$, $r_3 = 0$, and $r_1 = 20$ (Fig. 7). In evolution center E_1 agents could form aggregates consisted of two cells.

5.2.3 Experiment with three evolution centers

There were three active evolution centers during the whole simulation (Fig. 5). In E_2 and E_3 evolution centers the aggregation was not allowed. These two evolution centers served as areas where the process of species formation took place. In the evolution center E_1 agents could form aggregates consisted of two cells. Agents could freely move from one evolution center to another.

5.3 Discussion of the results

Because of the stochastic nature of the experiments all the results presented in this section are average values of three simulations carried out with the use of 0/1 random sequences $x_2(t)$ and $x_3(t)$. The length of all the simulations was 4000 steps. Everywhere in the figures shown below: 1 means results of the experiment carried out with the use of one evolution center (Section 5.2.1), 2 — with the use of two evolution centers (Section 5.2.2), and 3 — with the use of three evolution centers (Section 5.2.3).

Figure 8 shows the changes of the number of agents (aggregates) in the population during the simulations. The initial number of agents is 300. In each case we can observe the decline of the number of agents just at the beginning of simulation. But after a short period of time the number of agents rapidly grows and after several slight falls reaches a balance state. Only in the case of simulation with two environments we can again observe a decline in 2000-th step of simulation. This fact is caused by the process of migration of agents from evolution centers E_2 and E_3 to evolution center E_1.

In the second half of the simulation experiment with two evolution centers the number of agents is smaller than in the first half because in the second half of the simulation agents remain within

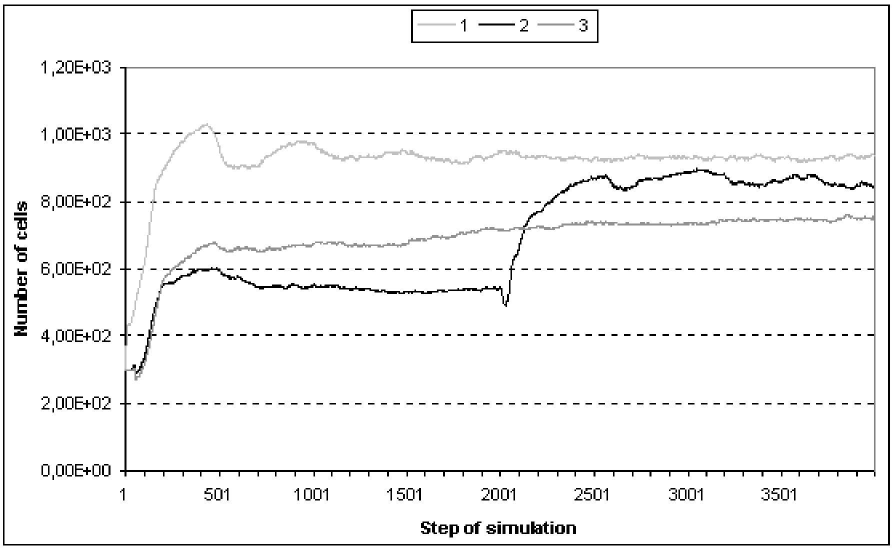

Figure 9: Number of cells in population

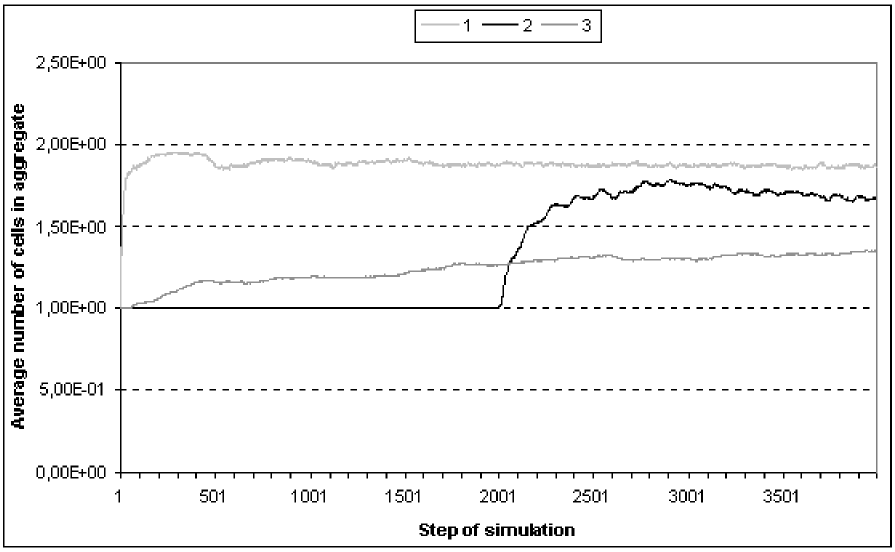

Figure 10: Average number of cells in aggregate

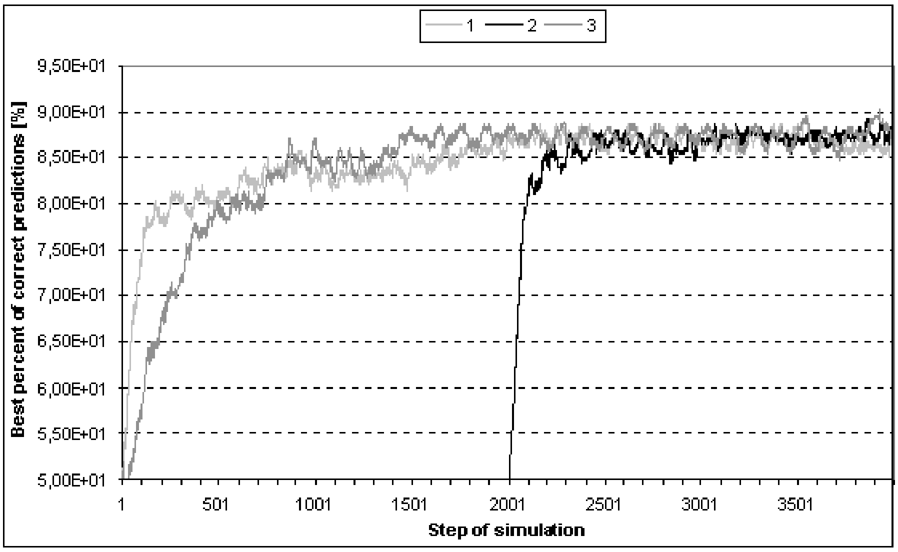

Figure 11: Best percent of correct predictions in population

the area of evolution center E_1, where the aggregation operator is active. In the Figure 8 we can see that in the second half of this experiment the number of agents is almost identical as it is in the case of simulation with one evolution center (similar conditions of the experiment). But there are less cells in the experiment with two evolution centers (see Figures 9 and 10).

In the case of experiment with three evolution centers one part of the population of agents remains within evolution centers E_2 and E_3 (in which aggregation is not allowed). We can therefore observe the biggest number of agents (Fig. 8) and the smallest number of cells in this experiment (Fig. 9 and 10).

In the Figure 11 we can see the comparison of best percent of correct predictions in sub-population that predicts values of the α parameter in evolution center E_1 (see Section 5.1.1). In the case of results obtained with the use of two evolution centers the rapid growth near the 2000-th step of simulation is the result of migration of the population from evolution centers E_2 and E_3 to evolution center E_1 (there were no agents in E_1 because $r_1 = 0$ until this moment). In all the cases the best percent of correct predictions in population is between 85 and 90%, although we can see slightly better results obtained with the use of two and three evolution centers by the end of simulation.

Figure 12 shows average percent of correct predictions in sub-population that predicts values of the α parameter in the evolution center E_1. When two and three evolution centers are used, average percent of correct predictions exceeds the level of 75%, which is not possible when only one evolution center is used.

Figures 13, 14, and 15 show changes of percent of correct predictions' distribution in population during the simulations with one, two, and three evolution centers. In the first step of simulation all the agents have 50% of correct predictions (this is the initial value). The next generations of agents are descendants of the agents with best fitness to the environment conditions. We can therefore observe systematic growth of the number of agents with higher levels of correct predictions.

In the case of simulation experiment with one evolution center the new maximum (about 41% of population) is rising in the group of agents with 75-80% of correct predictions (Fig. 13). In the last step of simulation there are about 13% of agents with 80-85% of correct predictions and 2% of agents with 85-90% of correct predictions in the population (Fig. 13 and 16).

In the Figure 14 we can observe the shift of maximum from the group of agents with 85-90% of correct predictions (about 65% of population) to the group with 80-85% of correct predictions

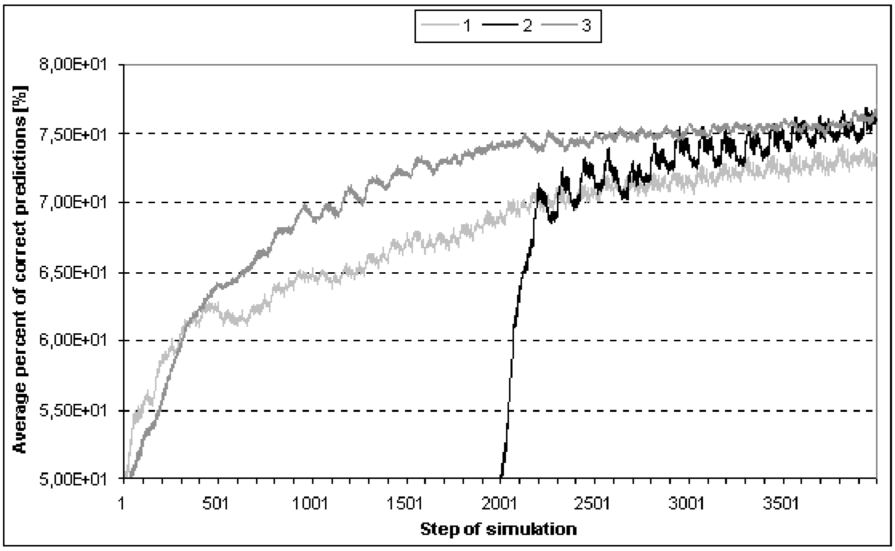

Figure 12: Average percent of correct predictions in population

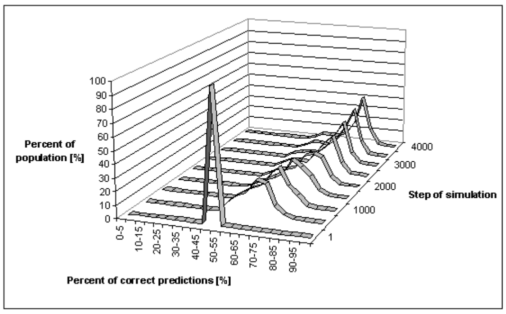

Figure 13: Changes of percent of correct predictions' distribution in population (one evolution center)

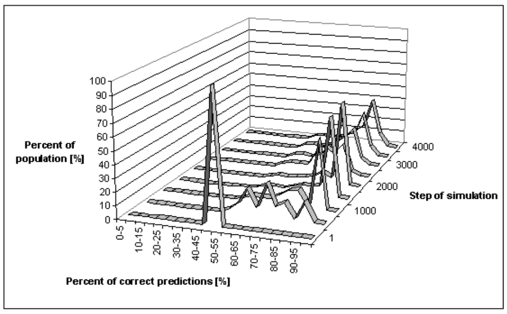

Figure 14: Changes of percent of correct predictions' distribution in population (two evolution centers)

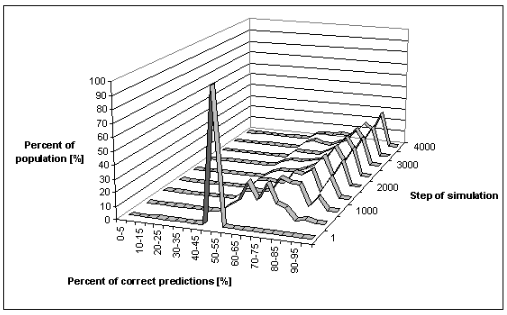

Figure 15: Changes of percent of correct predictions' distribution in population (three evolution centers)

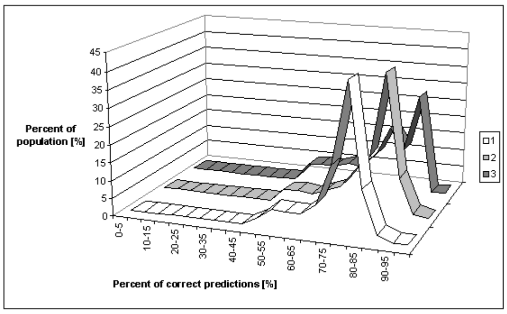

Figure 16: Percent of correct predictions' distribution in population in the last step of simulation

(about 39% of population in the last step of simulation) after 2000-th step of simulation. This fact is caused by the migration of population from evolution centers E_2 and E_3 to evolution center E_1 (the whole population has to predict the values of parameter α taken from the longest sequence). But there still exists the group of agents (about 10% of population) with 85-90% of correct predictions and the group of agents (about 16% of population) with 75-80% of correct predictions (Fig. 14 and 16).

In the Figure 15 we can observe the results of the simulation experiment carried out with the use of three evolution centers. There are three evolution centers during the whole simulation in this type of experiment what implies that one sub-population remains within evolution center E_1 and two others remain within evolution centers E_2 and E_3 (where the values of parameter α are taken from the shorter sequences). We can therefore observe that the new maximum (about 28% of population) is rising in the group of agents with 85-90% of correct predictions. Also, there is the group of agents (about 12% of population) with 80-85% of correct predictions and the group (about 19% of population) with 75-80% of correct predictions (Fig. 15 and 16).

To sum up the presented results:

- In the case of all three experiments the best percent of correct predictions in population is almost identical.
- The average percent of correct predictions is highest when two and three evolution centers are used.
- Also, there are many more agents with high percentage of correct predictions in the population when two and three evolution centers are used and the process of species formation can take place.

6 Conclusions

Evolution process realized in multi-agent world offers us two new possibilities:

1. Avoiding limitations of the classical evolution programs.
2. The mechanism of adaptation towards changing environment for agents.

With the Evolutionary Multi Agent Systems it is possible to model biological mechanisms of species formation, rivalry and competition among species, social behavior and so on. With the use of these biological mechanisms we can supply the modeled evolution process with new impulses and try to direct it towards new and desirable (from our point of view) possibilities.

In this paper we have introduced the supervising mechanism based on two non-classical evolution operators: aggregation and escape. The aggregation operator enables the emergence of social relations among agents [3]. The escape operator enables agents to migrate towards different environments. The application of these evolution operators made it possible to construct the evolution centers. The evolution centers are geographically separated areas, with different environment conditions and characteristic parameters. It is therefore possible to model the process of species formation in the population of agents. In the evolution centers we also can control values of various environment parameters in order to supervise the evolution process.

We have applied the idea of evolution centers to the construction of evolutionary multi-agent prediction system. It has been shown that the application of evolution centers significantly improved the evolution process realized in the multi-agent prediction system.

Future research in the field of Evolutionary Multi Agent Systems should be focused on introducing new evolution mechanisms based on the biological evolution. Therefore, it will be possible to research different phenomena that exist in the process of biological evolution and apply EMAS technology to solving complex and hard problems.

References

[1] E. Cetnarowicz, E. Nawarecki, and K. Cetnarowicz. Agent oriented technology of decentralized systems based on the m-agent architecture. In *Proceedings of the Management and Control of Production and Logistics Conference – MCPL'97*, Sao Paulo, Brazil, 1997. IFAC, PERGAMON.

[2] K. Cetnarowicz. M-agent architecture based method of development of multi-agent systems. In *Proceedings of the 8-th Joint EPS-APS International Conference on Physics Computing*, Krakow, Poland, 1996. ACC Cyfronet.

[3] K. Cetnarowicz, M. Kisiel-Dorohinicki, and E. Nawarecki. The application of evolution process in multi-agent world to the prediction system. In *Proceedings of the Second International Conference on Multi-Agent Systems — ICMAS'96*, Osaka, Japan, 1996. AAAI Press.

[4] K. Cetnarowicz, E. Nawarecki, and M. Zabinska. M-agent architecture and its application to the agent oriented technology. In *Proceedings of the International Workshop – Distributed Artificial Intelligence and Multi-Agent Systems — W–DAIMAS'97*, St. Petersburg, Russia, 1997.

[5] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence Through Simulated Evolution.* John Wiley, Chichester, UK, 1966.

[6] D. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, 1989.

[7] D. Larkin and G. Wilson. *Object-Oriented Programming and the Objective-C Language.* NeXT Software, Inc., 1995.

[8] Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer -Verlag, 1996.

[9] S. Rasmussen and C. L. Barrett. Elements of a theory of simulation. In *ECAL'95, Lecture Notes in Computer Science.* Springer-Verlag, 1995.

[10] Swarm Development Group. *Brief Overview of Swarm.* Santa Fe Institute, Santa Fe, USA, 2000. http://www.swarm.org.

[11] Swarm Development Group. *Reference Guide for Swarm 2.1.1.* Santa Fe Institute, Santa Fe, USA, 2000. http://www.swarm.org.