Marek Kisiel-Dorohinicki

EVOLUTIONARY MULTI-AGENT SYSTEMS IN NON-STATIONARY ENVIRONMENTS

Abstract

In this article, the performance of an evolutionary multi-agent system in dynamic optimization is evaluated in comparison to classical evolutionary algorithms. The starting point is a general introduction describing the background, structure and behavior of EMAS against classical evolutionary techniques. Then, the properties of energy-based selection are investigated to show how they may influence the diversity of the population in EMAS. The considerations are illustrated by experimental results based on the dynamic version of the well-known, high-dimensional Rastrigin function benchmark.

1. Introduction

Both *evolutionary algorithms* as well as *evolutionary computation* denote a variety of heuristic optimization techniques based on models of organic evolution. The approach is particularly useful when classical computational methods prove ineffective, like in problems with dynamically-changing characteristics [21]. Unfortunately, for evolutionary techniques, there is no guarantee that satisfactory solutions will be found, so their performance for particular problems must be demonstrated empirically. Nonetheless, evolutionary algorithms may often give approximate solutions without any problem-specific knowledge which, in turn, enables adaptation in non-stationary environments.

The application of agent approach in population-oriented computational intelligence techniques such as evolutionary algorithms seems obvious, but surprisingly, one can find relatively few publications on this topic [18, 7]. The idea of an *evolutionary multi-agent system* (EMAS) is the result of a search for a computing model that would be as similar as possible to the real-world evolutionary processes [6]. This search was aimed at creating a technique that would have a number of new useful features in comparison relative to classical evolutionary algorithms. Since being proposed for the first time in more than 15 years, a considerable number of publications have allowed the verification of many of the assumptions regarding the features of EMAS as an optimization technique [2]. Because of non-global resource-based selection, EMAS tends to preserve the diversity of the population. This allows improved results than received with classical evolutionary techniques, even in typical benchmark problems [3].

In this article, the performance of EMAS applied in non-stationary environments is investigated based on the dynamic version of the well-known, high-dimensional Rastrigin function benchmark. The next section aims to provide a general introduction to evolutionary multi-agent systems, with motivation, structure and behavior of EMAS described with reference to classical evolutionary techniques. The dynamic optimization problem and the particular realization of EMAS with resource-based selection is described, in turn, with emphasis on the influence of this mechanism on the diversity of the population. The experimental results show a comparison between the performance of EMAS and a classical parallel evolutionary algorithm.

2. Evolutionary multi-agent systems

Practice proves that evolutionary algorithms work properly if the population consists of fairly different individuals; i.e. the so-called diversity in the population is preserved [1]. Many algorithms tend to prematurely lose this useful diversity, which is particularly important in non-stationary optimization problems [21]. To maintain the population diversity, different evolutionary algorithms introduce additional mechanisms following the most important phenomena in evolutionary biology. Among these, decomposition and co-evolutionary approaches had the most inspiring effect on the genesis of *evolutionary multi-agent systems* (EMAS).

Decomposition approaches of the so-called *parallel evolutionary algorithms* (PEA) model introduce non-global selection/mating and some spatial structure of population [5]. In *co-evolutionary* algorithms, the fitness of each individual is not computed directly based on the definition of the problem to be solved, but rather results from interactions with other individuals in the population [16]. In these approaches, the exact course of evolutionary processes depend on some context of a particular individual or subpopulation (e.g. the location in the population structure in the island model of PEA, as depicted in Fig. 1). The idea of a decentralized model of evolution realized in evolutionary multi-agent systems may be considered as a step further in that direction.
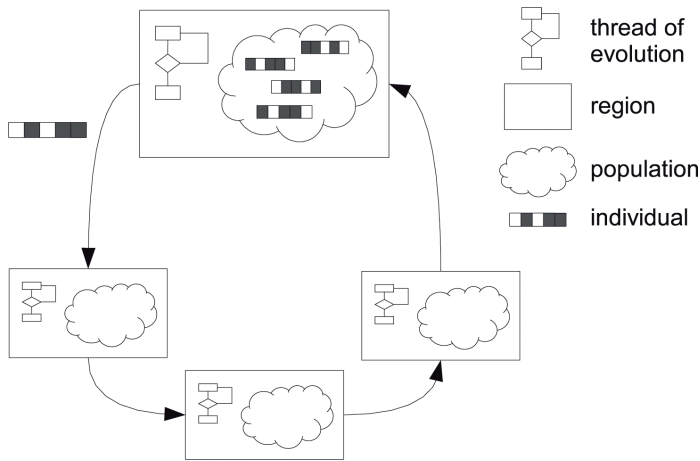
Figure 1. Island model of a parallel evolutionary algorithm (PEA).

Agents of EMAS represent solutions for a given optimization problem (Fig. 2). New agents get solutions inherited from their parent(s) with the use of variation operators (mutation and recombination) in the action of *reproduction*. At the same time, agents are removed from the system when they realize the action of *death*. Selection is based on non-renewable resources, which means that a decisive factor of the agent's activity is the quality of the solution it represents, but expressed by the amount of non-renewable resource it possesses. Selection is then realized in such a way that agents with a lot of resources are more likely to reproduce, while the low level of resources increases the possibility of death [6].

Agents are located on islands (representing distributed structure of computation) which constitute their local environment where direct interactions may take place. Obviously, agents are able to change their location, which allows for the diffusion of information and resources all over the system and supports distributed implementation of the system [14].
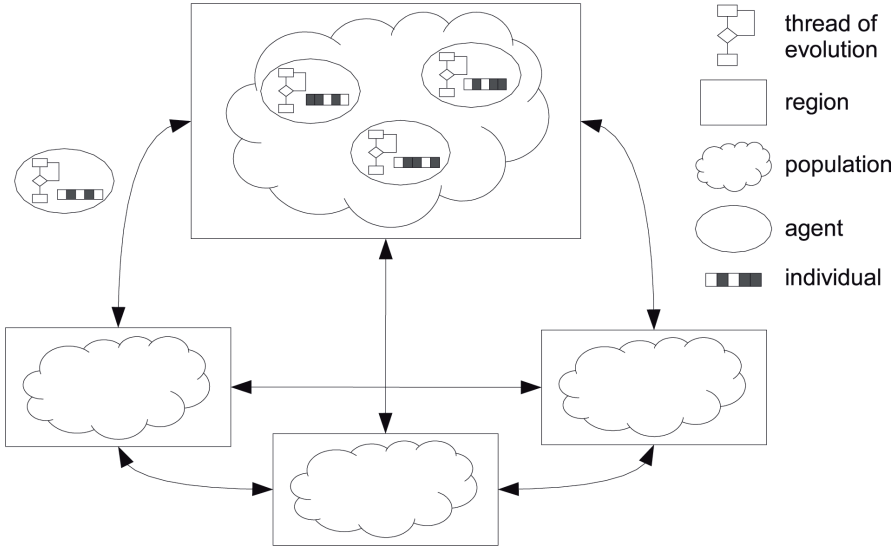
Figure 2. Evolutionary multi-agent system (EMAS).

The main advantage of the approach under discussion is that it covers various specialized evolutionary techniques in one coherent model, which enables the following:

- local selection allows intensive exploration of the search space and explicitly defined living space facilitates deployment in a distributed computational environment, which is similar to parallel evolutionary algorithms,
- evaluation of agents, or more generally, the way a phenotype (behavior of the agent) is developed from a genotype (inherited information) depends on its interaction with the environment, like in co-evolutionary algorithms.

Indeed, both mechanisms support the preservation of population diversity, which allows the search to continue even when conditions in the environment change.

3. Dynamic optimization with EMAS

A global optimization problem is defined by the search space U and a criterion function $F : U \to \mathbb{R}$. The goal is to find the set of extrema $\mathcal{X} \subseteq U$, which may be formulated for maximization problems as:

$$\mathcal{X} = \{gen \in U; \forall_{gen' \in U} F(gen) \geq F(gen')\} \tag{1}$$

where U is a finite genetic universum $\#U = r < +\infty$ [11]. Of course, minimization problems may be easily changed into maximization by negating the fitness value.

A dynamic (non-stationary) optimization problem is also defined by the search space U, but a time-indexed set of quality functions $F^{(t)} : U \to \mathbb{R}, t \in \mathbb{N}_0$ defines the changing search criteria. The goal in this case is to determine the set of all extrema $\mathcal{X}^{(t)} \subset U (t \in \mathbb{N}_0)$:

$$\mathcal{X}^{(t)} = \{gen \in U; \forall_{gen' \in U} F^{(t)}(gen) \geq F^{(t)}(gen')\} \qquad (2)$$

Again in this case, U is a finite genetic universum $\#U = r < +\infty$ [9].

Solving such dynamic optimization problems by evolutionary techniques requires constantly maintaining the diversity of the population. Niching and speciation techniques, like decomposition and co-evolutionary approaches mentioned in the previous section, were often considered to achieve this for classical evolutionary algorithms. Also, the approaches known as hypermutation (increasing the mutation rate after the change of the fitness function) or random immigrants (introducing randomly generated individuals into the population) were developed [12]. Other techniques, such as employing dominance and diploidy, were also applied [10]. Some other approaches attempt? to predict the changes in the optima position [19].

EMAS for dynamic optimization was constructed according to the model presented in [4]. In this case, agents possess one non-renewable resource called *life energy* which drives the distributed selection process. It is mainly based on the actions of death and reproduction, performed by agents reaching appropriate levels of energy as described in the previous section. Life energy also triggers the action of agent *migration*. In fact, each action is attempted randomly with a certain probability, but only when the particular agent's energetic level allows for it.

Agents acquire the information about their quality from the environment with respect to the defined criterion function. Yet, their actual evaluation is based on the idea of agents' meetings. Assuming some neighborhood based on population structure, agents evaluate their neighbors and exchange energy. Lesser agents in regards to their fitness are forced to give a fixed amount of their energy to their better neighbors. This flow of energy creates a situation that, in successive generations, surviving agents should represent better approximations of the solution [14]. Depending on the configuration of these energy-based mechanisms, even the worst agent in the population may survive for several steps until the meetings cause it to lose all its energy. This supports maintaining population diversity without any additional techniques.

In a non-stationary environment, the agent in consecutive steps may simply acquire different information about its quality (which indirectly influences the the flow of energy in the population). This means that the structure and behavior of EMAS for dynamic optimization problems are pretty much the same as in the static cases. The only thing which may require fine-tuning is the configuration (parameters) of the energetic profile of an agent: the initial amount of energy, death, reproduction thresholds, and evaluation energy.

4. Experimental results

Having experience in the development of component-based agent-oriented computing platforms (cf. AgE[1] platform [17]), a simplified version of such a discrete-event simulation and computing system was developed using Python technology. The choice of this technology was motivated by a relatively easy implementation process and high portability. Using this software environment, both EMAS and PEA were implemented in one coherent system and used to generate the presented results. All possible parameters of the both systems were set to the same values. The configuration of the both systems is presented as follows.

Common parameters:

- Mutation: continuous distribution-based modification of one randomly chosen gene.
- Crossover: single-point, the descendant gets parts of its parents genotype after dividing them in one randomly chosen point.
- Speciation type: allopatric (island-model).
- Topology of inter-island connections: fully connected.
- number of steps: 10 000.
- benchmark problem: 100-dimensional Rastrigin function (thus the genotype length: 100).
- agent/individual migration probability: 0.01.
- dynamic features: the criterion function was translated every 2000 steps by a vector of: $[0.5, \ldots, 0.5]$.

PEA parameters:

- mating pool size equals to number of individuals,
- individual migrate independently (to different islands),
- two islands, each populated with 50 individuals.

EMAS parameters:

- initial energy: 100, received by the agents in the beginning of their lives,
- minimal reproduction energy: 90, required to reproduce,
- evaluation energy win/loose: 40/-40, passed from the looser to the winner,
- death energy level: 0, used to decide which agent should be removed from the system,
- boundary condition for the intra-island lattice: fixed, the agents cannot cross the borders,
- intra-island neighborhood: Moore's, each agent's neighborhood consists of 8 surrounding cells,
- size of 2-dimensional lattice as an environment: 10×10,

---

[1]http://age.iisg.agh.edu.pl

- all agents that decided to emigrate from one island, will immigrate to another island together (the same for all of them),
- two islands, each populated with 50 agents.

The experiments were repeated 30 times, and the standard deviation was computed as a measure of repeatability. The results presented below concern the observation of the best fitness in consecutive steps of the computation. Moreover, the observation of diversity was also performed using two selected measures:

- Morrison-De Jong (MOI) measure is based on concept of moment of inertia for centroid (center of gravity computed for points distributed in multi-dimensional space) and allows for measurement of mass distribution into arbitrarily high dimensionality spaces [15].
- Maximum standard deviation (MSD) of each gene computed for all individuals in the population focuses on dispersion of the average values computed for individual genes.

In Figure 3, a simple comparison of the results of computation, namely observation of the best fitness in each step of the computation, may be seen.
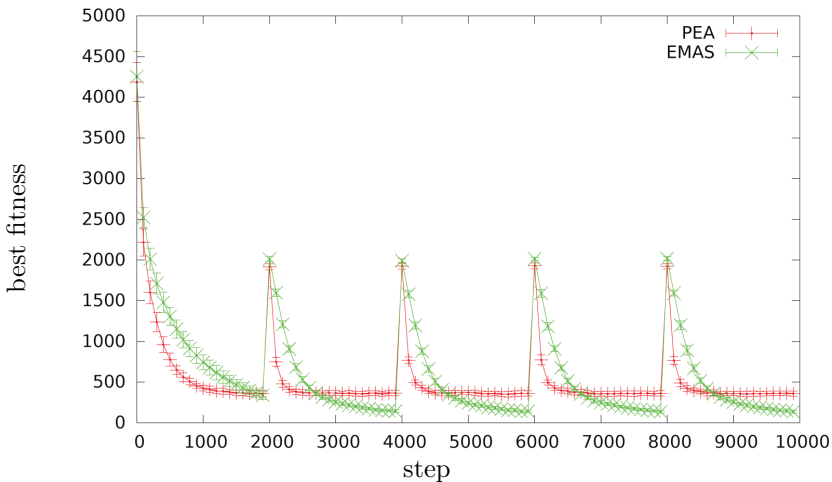

Figure 3. Results obtained: EMAS vs. PEA.

It turns out that EMAS obviously outperforms PEA.

In Figures 5 and 4, the diversity computed according to MSD and MOA schemes were shown. It is easy to see that the diversities of EMAS and PEA are comparable in both cases; however, in the case of MOI measurement, there are clearly visible peaks of diversity for EMAS that are visible in the moments of changing of fitness function. This seems to be a promising result: the system increases its diversity when it loses track of the current optimum.
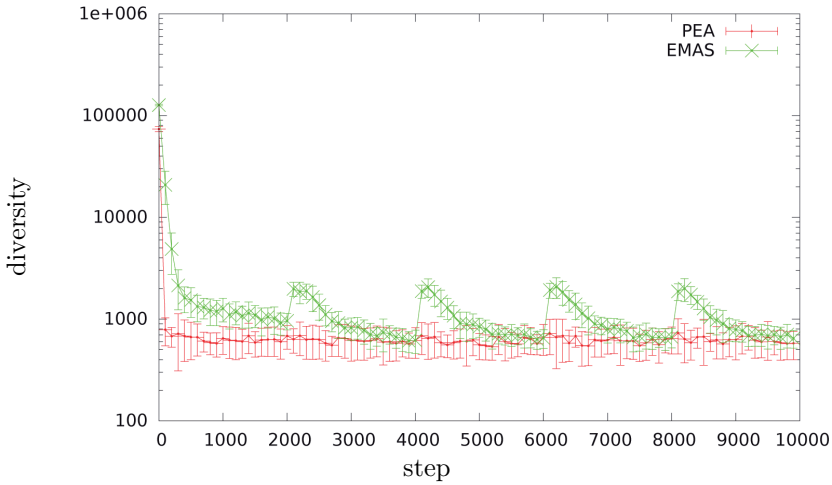
Figure 4. MOI diversity: EMAS vs. PEA.
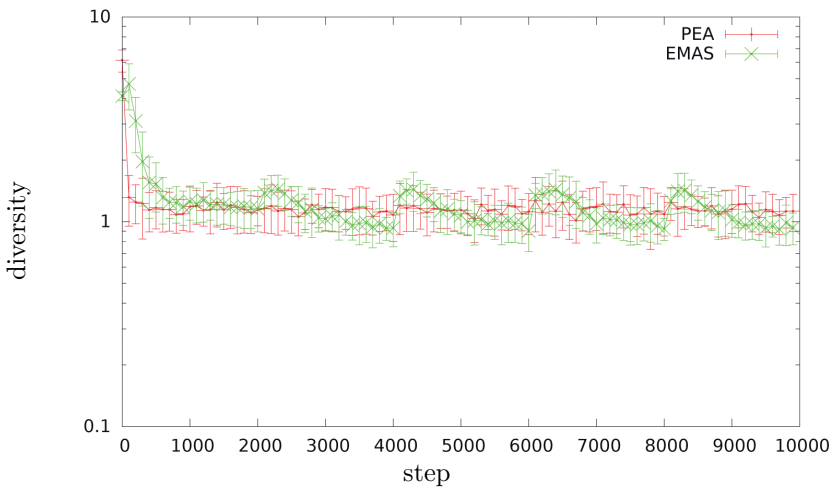
Figure 5. MSD diversity: EMAS vs. PEA.

In Table 1 (cf. [20]), the values of fitness gathered just before the change of optimum were shown. It is easy to see that EMAS outperforms the genetic algorithm, just as it was shown in Figure 3.

Moreover, exact values of *accuracy* (computed as best fitness right before change of the optimum averaged over the entire run) and *adaptability* (computed as best fitness over the entire run) the measures proposed also in [20], were given in Table 2.

Table 1

Fitness before changes of the optimum.

Step no.	EMAS	StDev	PEA	StDev
1900	337.53	51.20	357.55	40.42
3900	142.69	15.92	365.15	33.40
5900	141.57	14.24	362.56	31.98
7900	135.94	15.74	362.48	33.52

Table 2

Accuracy and adaptability of EMAS and PEA.

|	EMAS	PEA
Accuracy	189.43	361.94
Adaptability	627.24	547.37

It is easy to see that EMAS is far more accurate than PEA, although is less adaptable. However, this feature allows a better reaction to the changes of the optima.

At the very end, it is worth seeing what happens when one tries to make EMAS preserve a higher population diversity by weakening the selective pressure. It is quite easy to do this simply by changing the amount of energy transferred during meetings (the so called evaluation energy). Fig. 6 shows how it influences the results obtained in a single experiment with evaluation energy of 20: after the change of criterion function, the process of adaptation is much slower for EMAS than for PEA.
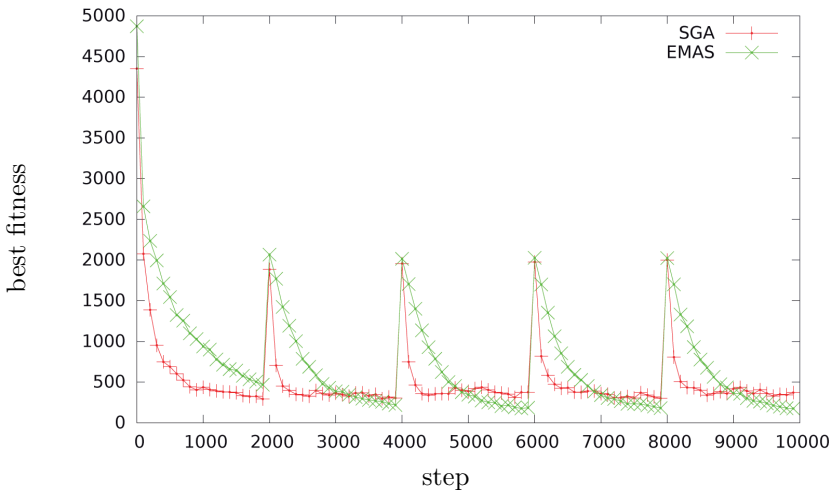
Figure 6. Results obtained for weakened selective pressure in EMAS vs. PEA.

If the criterion function changes too often, EMAS may be too slow to adequately react. Fig. 7 shows single experiment results when the criterion function translation occurs every 1000 steps while evaluation energy still equals to 20. In this case, classical evolutionary algorithms may prove better.
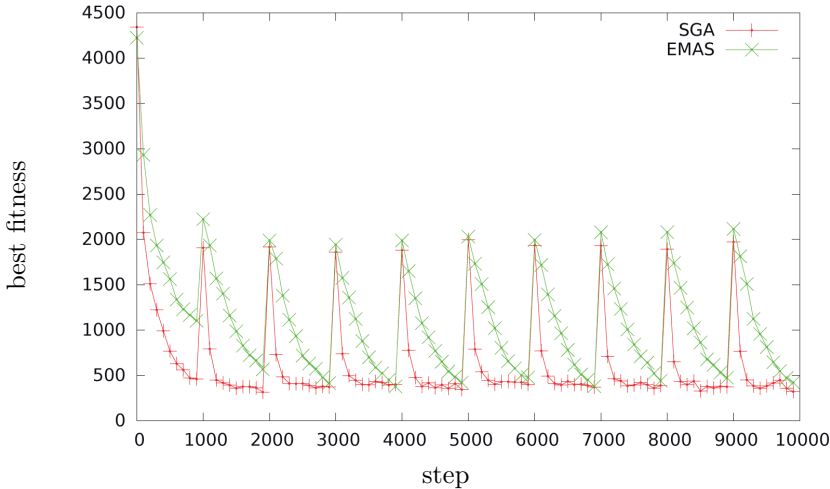

Figure 7. Results obtained for more frequent criterion function changes: EMAS vs. PEA.

An even more interesting phenomenon may be observed with frequent and periodic criterion function changes with a much larger translation vector. As presented in Fig. 8 in an experiment with criterion function periodic translation (twice by vector $[3, \ldots, 3]$ and then (twice by vector $[-3, \ldots, -3]$) occurring every 1000 steps, it might be said that EMAS 'remembers' its 'location' in the search space when, every two translations, the criterion function returns to the same configuration. PEA looks like it must learn from the beginning.

5. Conclusions

This paper presents the application of agent-based evolutionary computation paradigm realized as an evolutionary multi-agent system (EMAS) to dynamic (non-stationary) optimization problems. The results of both the conceptual and experimental comparative studies are interesting and promising, considering the diversity of the population and the quality of obtained solutions in both. The EMAS-based approach inevitably shares some of the flaws and limitations of evolutionary algorithms, including difficulties in their configuration.

In the future, similar experiments will be conducted in order to compare EMAS with other agent-based computing systems; e.g., hierarchic genetic search [13] or co-evolutionary optimization [8]. Research on EMAS continues, and one of its main directions is to build infrastructure and tools which support the implementation of
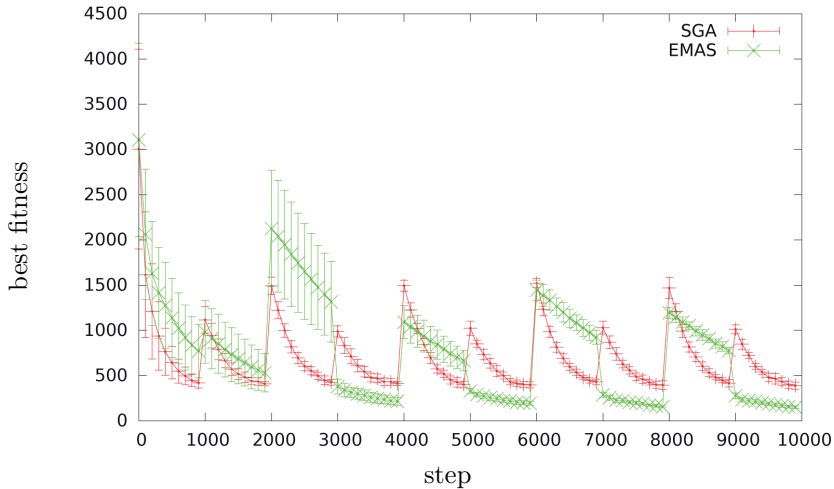
Figure 8. Results obtained for periodic criterion function changes: EMAS vs. PEA.

the systems in various application domains. Applying EMAS to solve optimization problems in different areas with varying features and characteristics makes it necessary to utilize flexible configuration mechanisms. This, in turn, drives the further development of distributed software frameworks that will allow the construction of agent-based computing systems consisting of reusable components [17].

Acknowledgements

References

[1] Bäck T., Fogel D., Michalewicz Z., eds.: *Handbook of Evolutionary Computation.* IOP Publishing and Oxford University Press, 1997.

[2] Byrski A., Dreżewski R., Siwik L., Kisiel-Dorohinicki M.: *Evolutionary Multi-Agent Systems. The Knowledge Engineering Review.* 2012 (Accepted for publication).

[3] Byrski A., Kisiel-Dorohinicki M.: Agent-Based Evolutionary and Immunological Optimization. In: *Computational Science – ICCS 2007, Proc. of 7th Int. Conf.*, *LNCS*, vol. 4488. Springer, 2007.

[4] Byrski A., Schaefer R.: Markov Chain Analysis of Agent-Based Evolutionary Computing in Dynamic Optimization. In: *Proc. of Int. Conf. on Computational Science, ICCS 2013*, *Procedia Computer Science*, vol. 18, pp. 1475–1484. Elsevier, 2013.

 [5] Cantú-Paz E.: *A summary of research on parallel genetic algorithms. IlliGAL Report No. 95007. University of Illinois*, 1995.

 [6] Cetnarowicz K., Kisiel-Dorohinicki M., Nawarecki E.: The application of evolution process in multi-agent world (MAW) to the prediction system. In: *Proc. of the 2nd Int. Conf. on Multi-Agent Systems (ICMAS'96)*, M. Tokoro, ed. AAAI Press, 1996.

 [7] Chen S.H., Kambayashi Y., Sato H.: *Multi-Agent Applications with Evolutionary Computation and Biologically Inspired Technologies*. IGI Global, 2011.

 [8] Dreżewski R.: Co-Evolutionary Multi-Agent System with Speciation and Resource Sharing Mechanisms. *Computing and Informatics*, **vol. 25**(4), 2006.

 [9] Goldberg D.: Genetic Algorithms in Search, Optimization, and Machine Learning. Massachusetts: Addison-Wesley, 1989.

[10] Goldberg D., Smith R.: Nonstationary function optimization using genetic algorithms with dominance and diploidy. In: *Proc. of the Second International Conference on Genetic Algorithms*, pp. 59–68. 1987.

[11] Horst R., Pardalos P.: *Handbook of Global Optimization*. Kluwer, 1995.

[12] Jin Y., Branke J.: Evolutionary Optimization in Uncertain Environment—A Survey. *IEEE Transactions on Evolutionary Computation*, **vol. 9**, pp. 303–317, 2005.

[13] Jojczyk P., Schaefer R.: *Global Impact Balancing in the Hierarchic Genetic Search. Computing and Informatics*, **vol. 28**(2), 2008.

[14] Kisiel-Dorohinicki M.: Agent-Oriented Model of Simulated Evolution. In: *SofSem 2002: Theory and Practice of Informatics*, W.I. Grosky, F. Plasil, eds., *LNCS*, vol. 2540. Springer-Verlag, 2002.

[15] Morrison R.W., Jong K.A.D.: Measurement of Population Diversity. In: *Artificial Evolution: 5th Int. Conf., Evolution Artificielle, EA 2001*, P. Collet, et al., eds., *LNCS*, vol. 2310, pp. 31–41. Springer, 2002.

[16] Paredis J.: Coevolutionary Computation. *Artificial Life*, **vol. 2**(4), pp. 355–375, 1995.

[17] Pietak K., Woś A., Byrski A., Kisiel-Dorohinicki M.: Functional integrity of multi-agent computational system supported by component-based implementation. In: *Proc. of 4th Int. Conf. on Industrial Applications of Holonic and Multi-agent Systems*, *LNAI*, vol. 5696. Springer, 2009.

[18] Sarker R., Ray T.: Agent-Based Evolutionary Search, *Adaptation, Learning and Optimization*, vol. 5. Springer, 2010.

[19] Simoes A., Costa E.: Improving prediction in evolutionary algorithms for dynamic environments. In: *Proc. of the 2009 Genetic and Evolutionary Computation Conference*, pp. 875–888. 2009.

[20] Trojanowski K., Michalewicz Z.: Searching for Optima in Non-Stationary Environments. In: *Proc. of the Congress on Evolutionary Computation. Washington, USA.*, vol. 3, pp. 1843–1850. IEEE Press, 1999.

[21] Weicker K.: *Evolutionary Algorithms and Dynamic Optimization Problems*. Ph.D. thesis, University of Stuttgart, 2003.

Affiliations

Marek Kisiel-Dorohinicki

AGH University of Science and Technology, Krakow, Poland, `doroh@agh.edu.pl`