Computer Science • 14 (3) 2013 http://dx.doi.org/10.7494/csci.2013.14.3.491

Aleksander Byrski

TUNING OF AGENT-BASED COMPUTING

Abstract *In this paper, an Evolutionary Multi-agent system-based computing process is subjected to a detailed analysis of its parameters in order to establish a base for a better understanding of the meta-heuristics from the practitioner's point of view. After reviewing the concepts of EMAS and its immunological variant, a series of experiments is shown, and results of the influence of the search outcomes by certain parameters is discussed.*

Keywords agent-based computing, agent-based meta-heuristics, biologically-inspired computing

1. Introduction

During the past few decades, intelligent/autonomous software agents have been gaining an ever-increasing number of applications in various domains, such as power systems management [20], flood forecasting [15], business process management [16], intersection management [12], or difficult optimization problem solving [18], just to mention a few. The key to understand the concept of a multi-agent system (MAS) is an intelligent interaction (like coordination, cooperation, or negotiation). Thus, multi-agent systems are ideally suited for representing problems that have many solving methods, involve many perspectives and/or may be solvable by many entities [28]. That is why one of the major application areas of multi-agent systems is large-scale computing [26, 1].

The article deals with a tuning of a hybrid evolutionary-agent approach. In most of similar applications reported in the literature (see e.g. [23], [9] for a review), an evolutionary algorithm is used by an agent to aid in the realisation of some of its tasks, often connected with learning or reasoning, or to support coordination of some group (team) activity. In other approaches, agents constitute a management infrastructure for a distributed realisation of an evolutionary algorithm [24].

Evolutionary processes are by nature decentralised, and therefore one can imagine the incorporation of evolutionary processes into a multi-agent system at a population level. It means that agents are able to *reproduce* (generate new agents), which is a kind of cooperative interaction and may *die* (be eliminated from the system), which is the result of competition (selection). A similar idea with limited autonomy of agents located in fixed positions on some lattice (like in a cellular model of parallel evolutionary algorithms) was developed by e.g. [30]. The key idea of the decentralised model of evolution employed by an *evolutionary multi-agent system*–EMAS [17] was to ensure full autonomy of agents.

Such a system consists of a relatively large number of rather simple (reactive), often homogeneous agents which possess or produce solutions to the same problem (a common goal). The considered problem is rather closed and static, but non-deterministic [22]. Because of both computational simplicity and a huge number of the agents, the influence of each single agent's behaviour on the overall system operation may be neglected, which allows for the efficient realisation in large-scale environments with lightweight infrastructure [6].

In other approaches, agents constitute a management infrastructure for a distributed realisation of an evolutionary algorithm [24]. Yet, evolutionary processes are decentralised by nature and one may indeed imagine the incorporation of evolutionary processes into a multi-agent system at a population level [17]. It means that, apart from interaction mechanisms typical of MAS (such as communication), agents are able to *reproduce* (generate new agents) and may *die* (be eliminated from the system). A similar idea, but with limited autonomy of agents located in fixed positions on some lattice (like in a cellular model of parallel evolutionary algorithms), was developed by e.g., [30].

The key idea of the decentralised model of evolution employed by an *evolutionary multi-agent system* (EMAS) was to ensure full autonomy of agents. Different variants of this model have been successfully applied to different optimisation problems e.g., optimization of neural-network architecture [4], multi-objective optimization [25], multi-modal optimization [13], and financial optimization [14] to name a few (a summary of EMAS-related review has been given in [2]).

Proposing a complex, hybrid technique calls for justification of its applicability and theoretical background. Strong theoretical background have already been supplied (by proving the feature of ergodicity for the Markov-chain-based model of EMAS [5]), and different applications have already been tested (as it was mentioned in the previous paragraph). As EMAS may now be perceived as an effective tool for optimisation, it is tnoteworthy that its configuration contains a vast number of parameters that should be carefully tested before applying these meta-heuristics by other practitioners to their problems. In particular, the important parameters of EMAS, such as the necessary proper tuning of mechanisms of distributed and immunological selection, could help in understanding it and further tuning the computation based on this knowledge.

In the beginning of this paper, the concepts of EMAS and immunological EMAS, along with selected experimental results cited after previous works, are described. Later, an extensive analysis of particular EMAS and iEMAS parameters is presented, concluding with a summary tackling the different interactions between the tested parameters. In the end, the paper is finished.

2. Evolutionary agent-based computing

In this section, two already-introduced flavours of EMAS are shortly described after [2], namely Evolutionary Multi-agent System [8] and immunological Evolutionary Multi-agent System [3].

2.1. Basic model of EMAS

Figure 1 shows the simplest possible model of an evolutionary multi-agent system, with one type of agents and one resource (called *energy*) defined. Genotypes of agents represent feasible solutions to the problem. Energy is transferred between agents in the process of *evaluation*. When the agent discovers that one of its neighbours (e.g. chosen randomly), has lower fitness, it takes part of its neighbour's energy; otherwise, it passes part of its own energy to the evaluated neighbour. The level of life energy triggers the following actions:

- *Reproduction* – performed when the agent's energy raises above a certain level, followed by production of a new individual in cooperation with one of its neighbours, with genotype based on parents' genotypes (crossed over and mutated) and part of energy also taken from its parents.
- *Death* – the agent is removed from the system when its energy falls below a certain level, the remaining energy is distributed amongst its neighbours.

- *Migration* – the agent may migrate when its energy rises above a certain level, then it is removed from one evolutionary island and moved to another according to predefined topology.

Each action is attempted randomly with a certain probability, and it is performed only when basic preconditions are met (e.g. an agent may attempt to perform the action of reproduction, but it will reproduce only if its energy rises above a certain level and it meets an appropriate neighbour).

2.2. EMAS with immunological selection

The main idea of applying immunological inspirations to speeding up the process of selection in EMAS is based on the assumption that 'bad' phenotypes come from 'bad' genotypes. Immune-inspired approaches were applied to many problems, such as classification or optimisation (e.g., [10]). The most frequently used algorithms of clonal and negative selection correspond to their origin, and are used in a variety of applications [27].

The general structure of immunological EMAS (iEMAS) is shown in Figure 2. A new group of agents (acting as lymphocyte T-cells) is introduced [3]. They are responsible for recognising and removing agents with genotypes similar to the genotype patterns of these lymphocytes. Another approach may introduce a specific penalty applied by T-cells to recognised agents (a certain amount of the agent's energy is removed) instead of removing them from the system. Of course, there must be some predefined affinity (lymphocyte-agent matching) function which may be based, e.g. on the difference of percentage between corresponding genes.

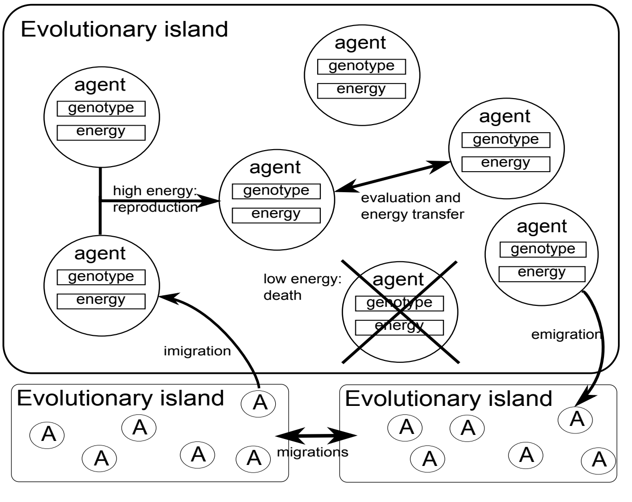


Figure 1. Evolutionary Multi-agent System (EMAS).

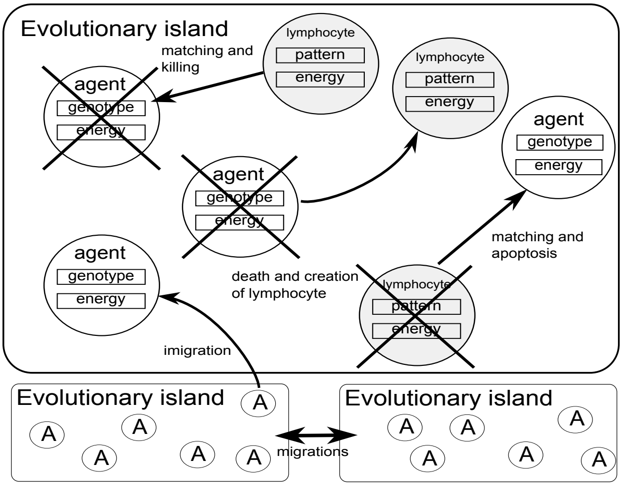


Figure 2. EMAS with immunological selection (iEMAS).

Agents-lymphocytes are created in the system after the action of death. The late-agent genotype is transformed into lymphocyte patterns by means of a mutation operator, and the newly-created lymphocyte (or group of lymphocytes) is introduced into the system. In both cases, new lymphocytes must undergo the process of negative selection. In a specific period of time, the affinity of immature lymphocyte patterns with 'good' agents (possessing a relatively high amount of energy) is tested. If it is high (lymphocytes recognise 'good' agents as 'non-self'), they are removed from the system. If affinity is low, it is assumed that they will be able to recognise 'non-self' individuals ('bad' agents), leaving agents with high energy intact. The life span of lymphocytes is controlled by specific, renewable resource (strength) used as a counter by the lymphocyte agent.

Therefore, EMAS is enhanced by adding lymphocyte agents, altering the action of the agent's death and adding three lymphocyte-related actions:

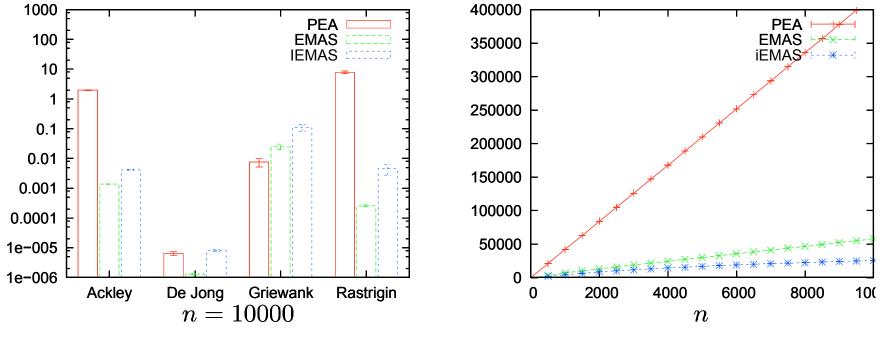
- *Death* – EMAS action of death is redefined: during this action, the agent produces one or more lymphocyte agents, passing along its mutated genotype to them and setting their strength to the maximum value.
- *Killing* – a mature lymphocyte (with energy below a certain level) removes (or weakens) one of its neighbouring agents, if it finds that the genotype of this agent matches its own, using a predefined affinity function. Immature lymphocytes (with strength above a certain level) are checked to confirm they match an agent with high energy; in this case, the lymphocyte is removed from the system,
- *Apoptosis* – lymphocyte with zero level of strength is removed from the system.

- *Give* – this action controls the negative selection process and overall lymphocyte life by simply decreasing the level of lymphocyte strength, allowing it to perform other actions (e.g., killing and apoptosis).

The concept of iEMAS is especially advantageous in applications requiring time-consuming fitness evaluation like the evolution of neural network architecture [3].

2.3. Selected EMAS and iEMAS experimental results

Experiments concerning minimisation of the benchmark functions presented below were reported by [3]. EMAS, iEMAS, and a classical parallel evolutionary algorithm were checked against popular benchmarks [11] in order to test their efficiency (10-dimensional functions of Ackley, De Jong, Griewank and Rastrigin). Variation operators of discrete crossover and uniform mutation were used. In parallel evolutionary algorithm (PEA, real-value encoding, Michalewicz version [21] with allopatric speciation [7]), tournament selection (being the most similar selection mechanism to energetic selection principle in EMAS) was used. The systems consisted of 3 evolutionary islands with 30 agents (or individuals in PEA) in the initial configuration.



(a) Fitness in 10000th step.

(b) Fitness count.

Figure 3. Final result obtained in 10000th step and fitness count for Ackley problem.

The fitness values obtained in the end of experiment (in 10 000th step) are presented in Figure 3(a). It is easy to see that the results of EMAS and iEMAS are better than PEA in three cases and worse in one. However, a more-important result is presented in Figure 3(b), see also Table 1. The number of fitness function calls is far lower for EMAS than for PEA and even lower for iEMAS. This makes these systems good weapons of choice for solving problems with costly fitness function (e.g., inverse problems) [29].

Table 1
Fitness count in 10000th step of system's work for EMAS, iEMAS and PEA.

Benchmark	PEA	EMAS		iEMAS	
Ackley	$6 \cdot 10^5$	57 840	±43	25 596	±149
De Jong	$6 \cdot 10^5$	57 769	±43	26 172	±158
Griewank	$6 \cdot 10^5$	58484	±62	30829	±379
Rastrigin	$6 \cdot 10^5$	57 787	±29	27 727	±220

3. Tuning of EMAS parameters

In this section, after presenting the parameters of the tested systems, a detailed discussion related to the process of tuning selected parameters for EMAS and its immunological variant (iEMAS) is given. In particular, distributed selection related parameters, immunological selection parameters, and probabilistic parameters are discussed. In the end, the summary is given in a tabular form. In order to peform the system tuning, one problem had to be selected. It is, in this case, the Rastrigin function [11] described in 50 dimensions. The PEA used is the real-value based, Michalewicz version [21] with allopatric speciation [7].

3.1. Configuration of the tested systems

The configuration of the tested systems is presented as follows.

- Common parameters:
  - normal distribution-based mutation of one randomly-chosen gene,
  - single-point crossover, the descendant gets parts of its parents' genotype after dividing them at one randomly chosen point,
  - 30 individuals located on each island,
  - all experiments were repeated 30 times and standard deviation was computed;
  - allopatric speciation (island model) was used, 3 fully connected islands were present,
  - stopping condition: reaching 3000th step of experiment,
  - genotype of length 50 (50-dimensional Rastrigin function),
  - agent/individual migration probability 0.01.
- PEA-only parameters: mating pool size equals to the number of individuals, individuals migrate independently (to different islands).
- EMAS-only parameters:
  - initial energy: 100, received by the agents in the beginning of their lives,
  - minimal reproduction energy: 90, required to reproduce,
  - evaluation energy win/lose: 40/−40, passed from the loser 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 neighbourhood: Moore's, each agent's neighbourhood consists of 8 surrounding cells,
- size of 2-dimensional lattice as an environment: 10×10,
- all agents that decided to emigrate from one island will immigrate to another island together (the same for all of them).

The following parameters were chosen for iEMAS:

- Energy taken by a lymphocyte from similar agent: 30
- Good fitness factor: 0.97 (percentage of the agent fitness related to average fitness in the population, as minimisation is considered, if fitness is smaller than the average fitness, it is considered "good").
- Similarity measure: Mahalanobis distance [19].
- Similarity threshold: 7.3, if similarity is smaller than this, the lymphocyte is considered to be similar to the tested agent.
- Immaturity duration for lymphocyte: 10.
- Maturity duration for lymphocyte: 20.
- Lymphocytes cannot migrate between the islands.

3.2. Energy-related parameters

Energy-based distributed selection mechanism is an immanent feature of EMAS. Therefore, a detailed examination of its parameters is crucial to a better understanding of the search process and the ability to effectively tune them in order to adopt the meta-heuristics to solving particular problems.

Energy exchange rate The most crucial parameter of the distributed selection mechanism in EMAS is the rate of energy exchange between competing agents. The influence of changing this parameter on the fitness and agent count in the population is shown in Figures 4, 5. It is easy to see that increasing this parameter makes improves the best fitness; but because of the applied logarithmic scale, this gain does not seem to be significant. However, as it could be predicted, this parameter greatly affects the agent count in the system. The higher the energy exchange rate, the lower the average agent count in the system.

Initial energy level The initial energy of the agents in the system is supposed to have a significant influence on the features of the agent population, as it is a main component of the total energy which serves as a base for the distributed selection mechanism. In fact, looking at Figure 6, the influence seems to be strong and straightforward.

The higher initial energy, the bigger the number of the agents during the computation. It is noteworthy that the selection mechanism is stable, as the number of agents does not grow indefinitely nor fall to zero during the whole observed computation process. It is easy to see that changing the initial energy indirectly affects the

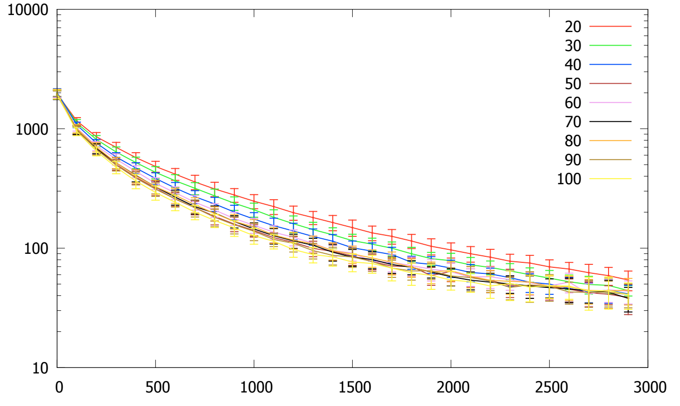


Figure 4. Influence of agent exchange energy on EMAS fitness $bestFitness(step)$.

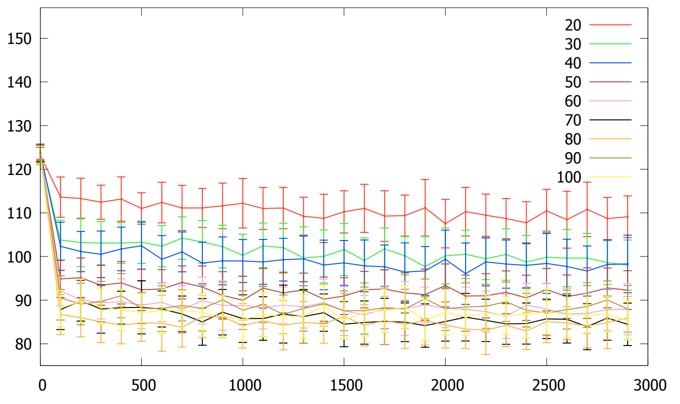


Figure 5. Influence of agent exchange energy on EMAS agent count $agentCount(step)$.

fitness (see Fig. 7), changing the actual number of the agents in the system that are capable of exploring and exploiting the search space. Generally speaking, increasing the initial energy helps achieve better results, though this effect is not very distinct.

Minimal reproduction energy Influence of minimal reproduction energy on the agent count is shown in Figure 8.

Reproduction of minimal energy of the agents is supposed to have a significant influence on the features of the agent population, as it directly affects the distributed selection mechanism by controlling the "maturity" of the agents capable of reproduction. If this parameter value is low, agents that performed few rendezvous will reproduce, while for its high value, only the long-living agents may generate offspring.

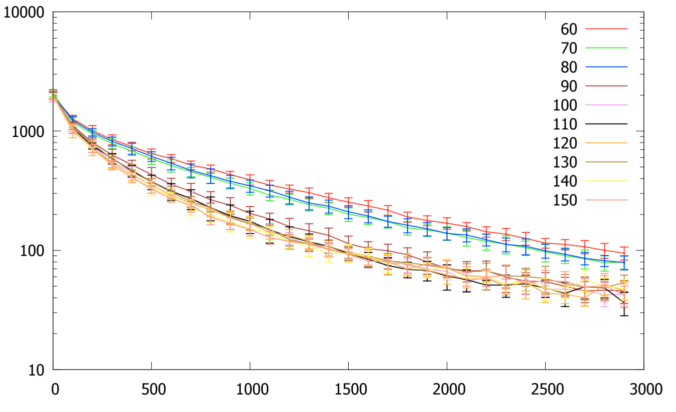


Figure 6. Influence of agent initial energy on EMAS fitness *bestFitness*(*step*).

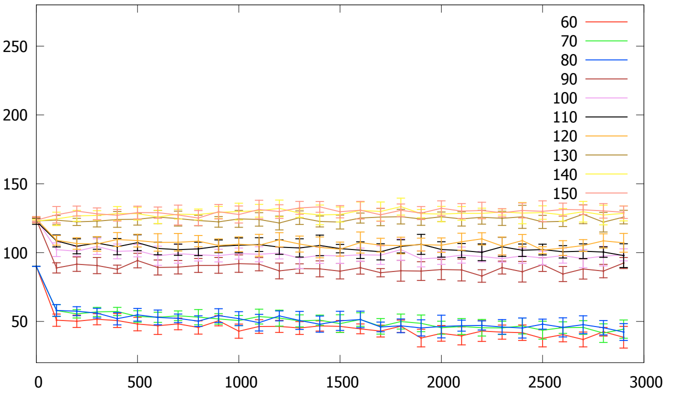


Figure 7. Influence of agent initial on EMAS agent count *agentCount*(*step*).

In fact, looking at Figure 9, the influence seems to be strong and straightforward, just the opposite in the case of initial energy.

It is easy to see that the higher the minimal reproduction level, the lower the number of agents during the computation, as it is harder for them to reproduce. Again, the selection mechanism is stable, as the number of agent does not grow indefinitely nor falls to zero during the whole observed computation process.

The fitness is also affected (see Fig. 8), because the number of the agents varies for different values of the minimal reproduction energy. The system is able to quicker and better explore the search space for lower levels of this parameter (the final results of the search are better for lower values of minimal reproduction energy, and the search is quicker as the graph curvature is higher).

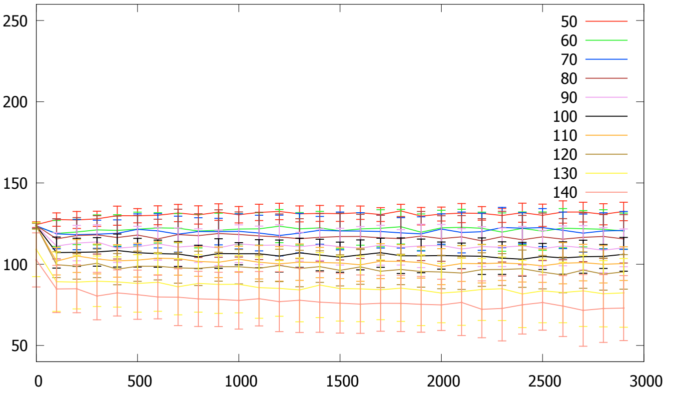


Figure 8. Influence of minimum reproduction energy on EMAS agent count $agentCount(step)$.

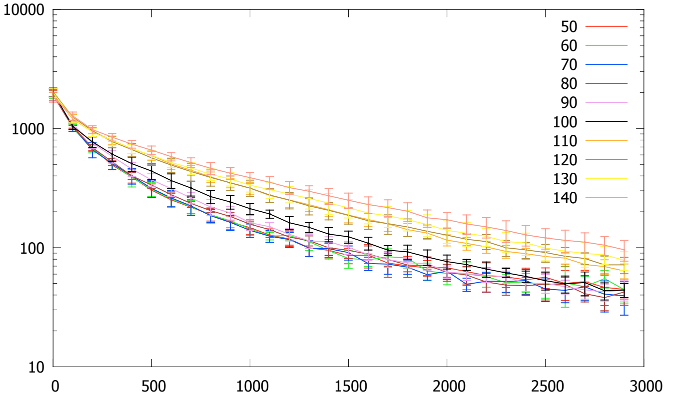


Figure 9. Influence of minimum reproduction energy on EMAS fitness $bestFitness(step)$.

3.3. Probabilistic parameters

Stochastic nature of the systems brings flexibility into the computation; however in order to effectively use EMAS and other related techniques, a detailed examination of the most important probabilistic parameters is necessary.

Migration probability Existence of migration phenomenon between the sub-populations should positively affect the value of fitness. It seems to be straightforward, because such techniques as niching and speciation are meant to increase the exploration efficiency of the algorithm. The straightforwardness of this effect does not allow us to draw any sophisticated conclusions; however, as a base, it is easy to see that introducing migration into the system is connected with enhancing the quality of results and diversity of the population (see Fig. 10). However the observed effect is almost discrete–if the probability is non-zero, the obtained results are significantly better. But further tuning of this parameter does not produce distinguishable changes in the fitness value.

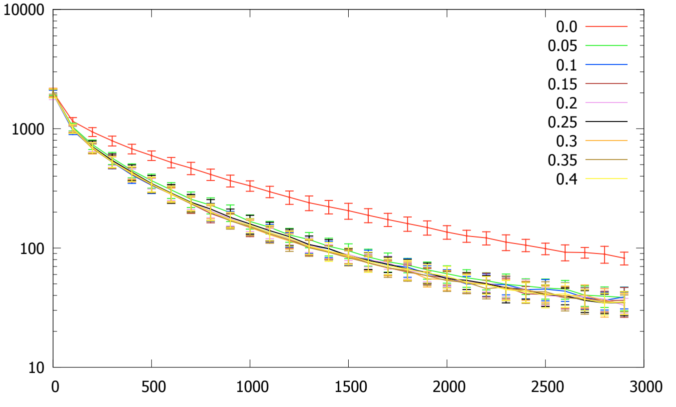


Figure 10. Influence of migration probability on EMAS fitness $\mathit{bestFitness}(\mathit{step})$.

Meeting probability Rendezvous probability is an important parameter affecting the frequency of the meetings between the agents (as the decision whether the agent performs a rendezvous or not is based on the outcome of probabilistic sampling). The higher the rendezvous probability is, the more frequently the agents will meet and exchange their energy.

However, this parameter does not influence the number of agents in the population (see Fig. 11), as the same number of agents simply exchange the energy faster or slower (also in memetic versions of EMAS). Again, the selection mechanism is stable, as the number of agent does not grow indefinitely nor falls to zero, during the whole observed computation process.

Increasing the rendezvous probability makes it possible to achieve the desired solutions quicker (see Fig. 12), as the energy flow from the "worse" agents to the "better" ones is faster, so the "better" ones may reproduce quicker. Therefore, the final results of the search are better for higher values of rendezvous probability, and the search is quicker as the graph curvature is higher. Again, changing this parameter does not affect gravely-memetic modifications of EMAS.

Very important information may be obtained when observing the diversity shown in Figures 13, 14. Increasing the rendezvous probability decreases the diversity. As having a diverse population is important in a population-based search [7], one should choose the value of this parameter in such a way that the desired solution is approached as fast as it was planned (as a result of exploitation), and the diversity is high enough (to maintain the exploration). Choosing an appropriate value of this

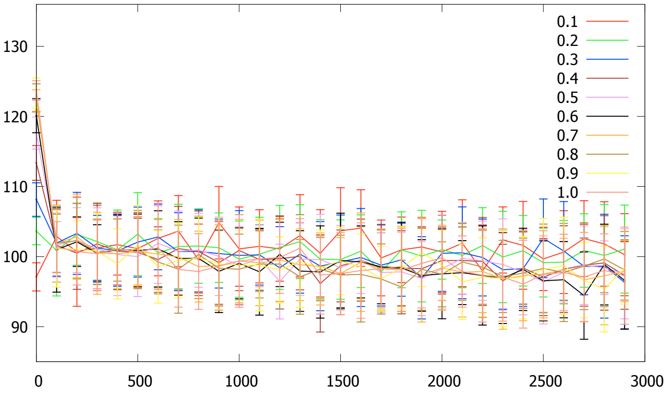


Figure 11. Influence of meeting probability on EMAS agent count $agentCount(step)$.

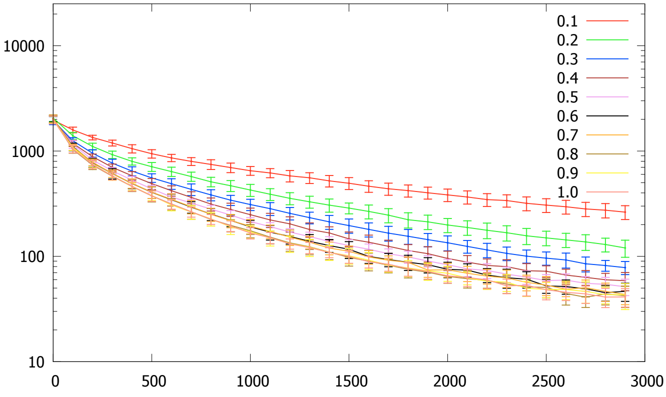


Figure 12. Influence of meeting probability on EMAS fitness $bestFitness(step)$.

parameter seems to be crucial to maintain the balance between the exploration and exploitation for EMAS and its variations.

3.4. Immunological parameters

As it was stated in [2], an immunological variant of EMAS (iEMAS) is an important weapon of choice when dealing with problems which have a complex fitness function. Therefore, examination of selected parameters influencing the immunological selection is necessary.

Penalty threshold One of them surely is the penalty threshold (quantity of energy taken from the agent that turns out to be similar to a lymphocyte during affinity testing). It is easy to see that changing this parameter significantly influences the

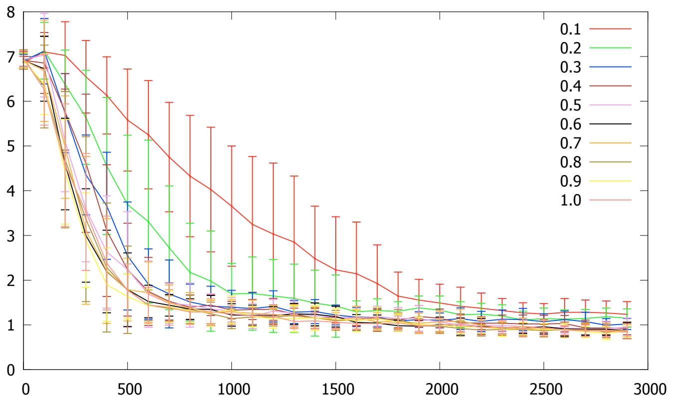


Figure 13. Influence of meeting probability on EMAS MSD diversity $divMSD(step)$.

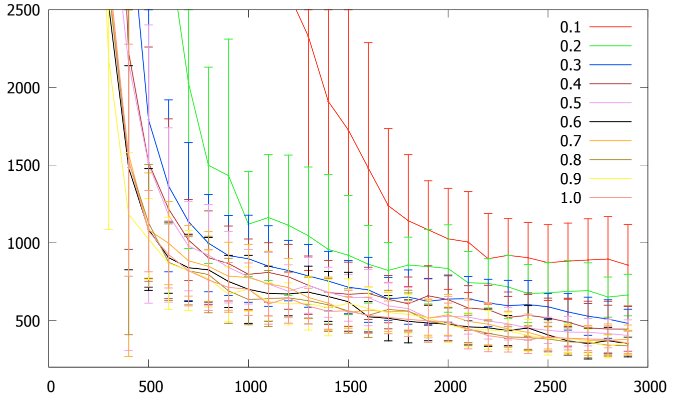
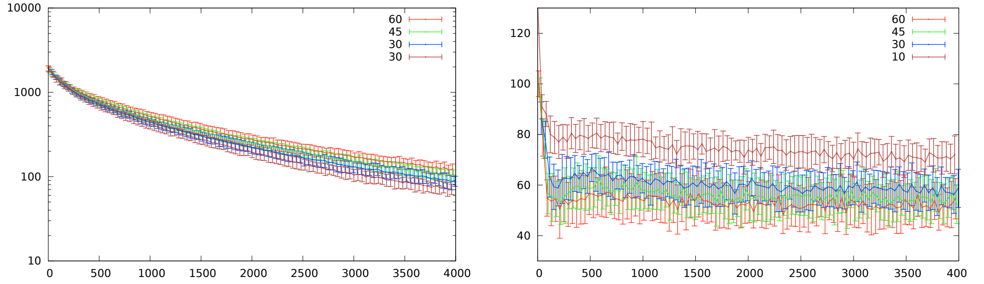


Figure 14. Influence of meeting probability on EMAS MOI diversity $divMOI(step)$.

number of agents in the system; however a very interesting fact is that the fitness remains almost unchanged (see Fig. 15(a)) for the examined range of parameters. This observation clearly indicates that introducing such a defined distributed tabu mechanism does not hamper the search capabilities of the system. Of course, the higher the penalty, the more agents are removed from the system; therefore the relation visible in (Fig. 15(b)) is predictable.

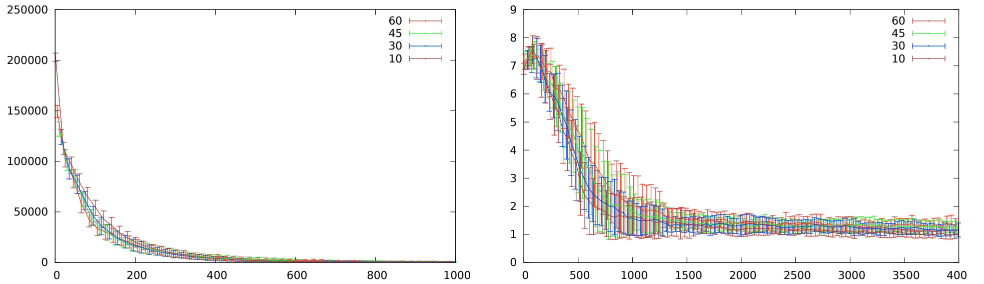
Observation of the diversity measures (see Fig. 3) shows that changing the penalty threshold (at the same time changing the immunological selection pressure) does not hamper the diversity. Moreover, quicker removal of "bad" agents makes the system more diverse (in the means of MSD metric).

Penalty threshold also has a predictable influence on the number of lymphocytes in the system (see Fig. 17), closely connected with reducing of the agent population.



(a) $bestFitness(step)$ (b) $agentCount(step)$

Figure 15. Influence of penalty threshold on fitness and agent count in iEMAS.



(a) $divMOI(step)$ (b) $divMSD(step)$

Figure 16. Influence of penalty threshold on diversity in iEMAS.

In such cases when the number of agents is lower, the same total sum of energy is distributed amongst the individuals of the smaller population; therefore, the average value of energy per agent is higher and agents die less frequently than when the population is bigger, generating smaller number of lymphocytes.

Lymphocyte life length Longer lymphocyte life (see Fig. 18) again does not significantly worsen the fitness; however, certain influence may be observed as the fitness becomes a little better in the case of shorter lymphocyte life. At the same time, of course, agent count is decreased with the rise of lymphocyte life as the lymphocytes may require more time to remove the individuals from the population.

At the same time, manipulating the life length of the lymphocytes does not hamper the diversity measures, though a little positive influence may be observed in the case of MSD diversity, when the lymphocyte life is longer (see Fig. 19).

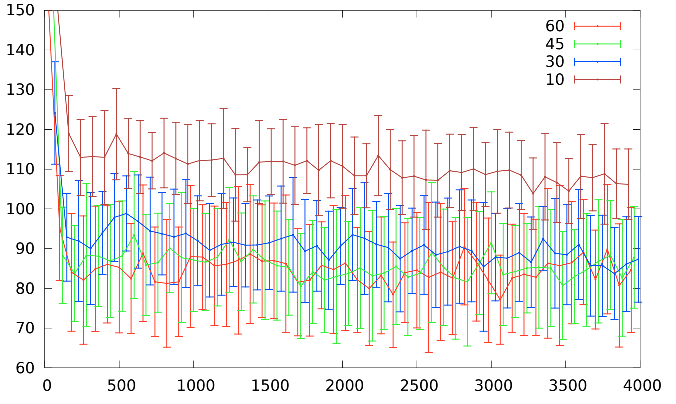
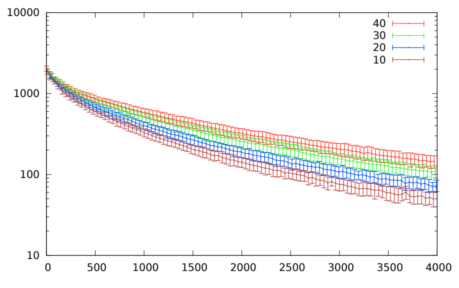
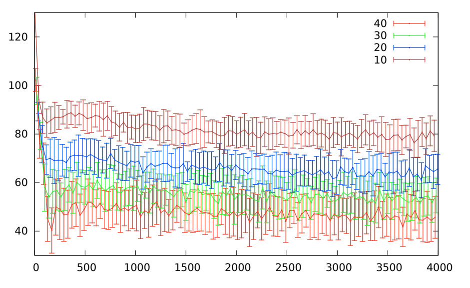


Figure 17. Influence of penalty threshold on lymphocyte count.



(a) *bestFitness*(*step*)

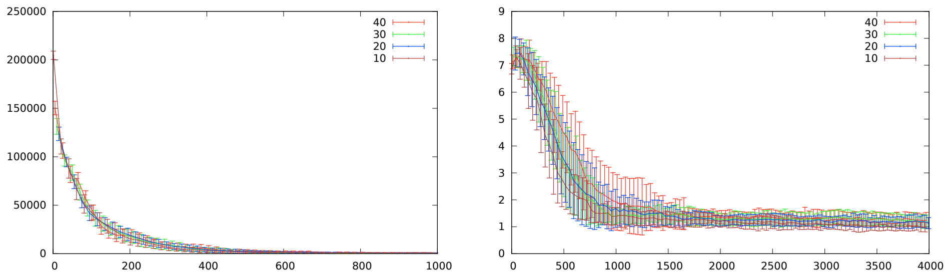


(b) *agentCount*(*step*)

Figure 18. Influence of lymphocytes' life lenght on fitness and agent count.

It is interesting that the length of the lymphocytes' life does not at all affect the number of lymphocytes in the system (see Fig. 20). It points out that the immunological selection mechanism is stable and the lymphocytes do not tend to overpopulate the agents; though, the average number counted has a significant diversity because of fully stochastic nature of the selection mechanism.

Percentage of "good" fitness During the negative selection process, the lymphocytes are removed when they are still considered immature, though they match a "good" agent in the population. This is the case when an immature lymphocyte matches an agent having fitness certainly related to the average fitness in the population (an appropriate percentage is considered). In Figure 21, the results of changing this percentage are shown along with the MSD diversity of the population. It is easy to see that these two graph sets are related. When the population is diverse (mostly



(a) $divMOI(step)$ (b) $divMSD(step)$

Figure 19. Influence of lymphocytes' life length on diversity.

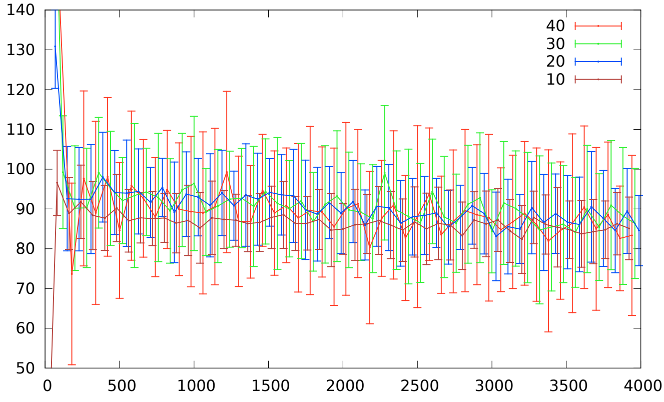
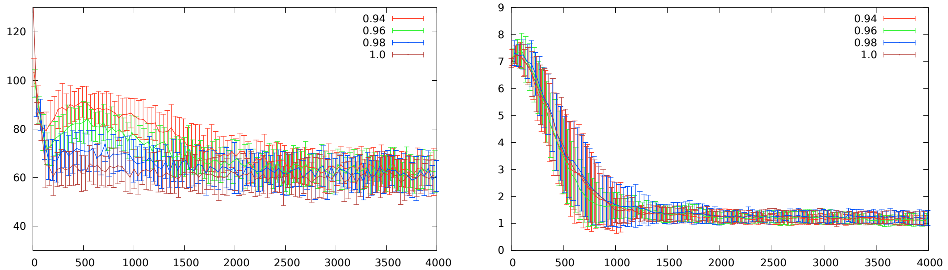


Figure 20. Influence of lymphocytes' life length on lymphocyte count.

in the beginning of the computation), the level of "good" fitness is lower than later when the diversity falls down. So, the lymphocytes tend to be removed more often; therefore the population of agents is higher.

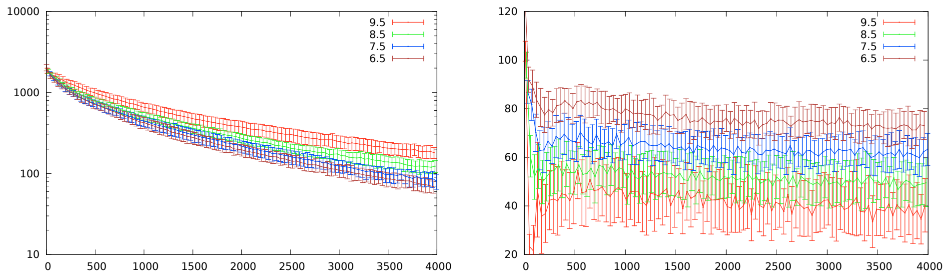
Other important parameters such as fitness, MOI diversity, and lymphocyte count are quite similar to the ones obtained earlier, being unchanged in the relation with the considered parameter.

Affinity measure In order to measure the affinity of the lymphocytes to the examined agents, Mahalanobis distance was used [19]. Lower value of distance meant that lymphocyte must match closer agent before penalising it (and vice versa). Therefore, it is easy to see that increasing the distance slightly hampers the obtained fitness, and of course, decreases the number of agents in the system (see Fig. 22).



(a) *agentCount*(*step*) (b) *divMSD*(*step*)

Figure 21. Influence of "good" fitness percentage on agent count and diversity.



(a) *bestFitness*(*step*) (b) *agentCount*(*step*)

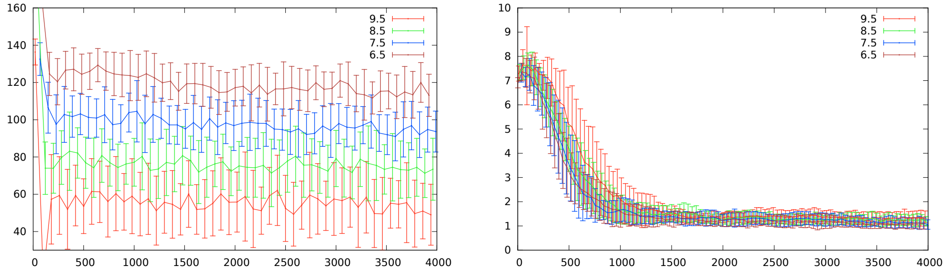
Figure 22. Influence of similarity threshold on fitness and agent count.

At the same time, observation of the lymphocyte count reveals that, if the distance is lower, more lymphocytes are created in the system as it is easier to remove the agent. It is connected, of course, with the similarity measure (this effect has been already observed before) that removing lymphocytes increases the MSD similarity measure (see Fig. 23).

3.5. Parameters tuning recapitulation

The performed experiments may surely become a base for researchers who are willing to apply the EMAS-like computing to their problems. In order to make this easier, the summary of the parameters' tuning is presented in Table 2.

Based on the results presented in this table, in order to appropriately parametrise the computation, one must focus not only on attaining the specified goal (e.g., good fitness), but also check whether other parameters comply with this goal (e.g., need to



(a) Lymphocyte count (b) MSD diversity

Figure 23. Influence of similarity threshold on lymphocyte count and MSD diversity.

Table 2
Parameters tuning summary.

Increase of the parameter	Fitness	Agent count	MOI	MSD	Lymphocyte count
Energy exchange rate	↗	↘	↗	—	
Initial energy level	↗	↗	—	—	
Minimal reproduction energy	↘	↘	↘	—	
Migration probability	↗	—	↗	—	
Meeting probability	↗	—	↘	↘	
Penalty level	—	↘	—	↗	↘
Good fitness percentage	—	↘	—	—	—
Lymphocyte life length	↘	↘	—	↗	—
Similarity distance	↘	↘	—	↗	↘

reduce the fitness function calls that is closely connected with the number of agents in the population).

4. Conclusion

In the course of this paper, Evolutionary Multi-agent system and its immunological variant (iEMAS) became a base for extensive testing of selected parameters. After presenting the concepts of the systems and citing important EMAS-related experimental results, the configuration of the systems was described and the detailed tests of selected parameters (distributed selection related parameters, immunological selection parameters and probabilistic parameters) were presented. The summary was given in a tabular form, in order to get an insight into relations between different parameters of the system. This work may be used as a reference for the practitioner who would like to apply EMAS meta-heuristic in a particular case.

Acknowledgements

The work presented in this paper was partially supported by Polish National Science Centre research project No. N N516 500039 "Biologically inspired mechanisms in planning and management of dynamic environments".

References

[1] Bouvry P., González-Vélez H., Kołodziej J.: *Intelligent Decision Systems in Large-Scale Distributed Environments*. Springer, 2011.

[2] Byrski A., Dreżewski R., Siwik L., Kisiel-Dorohinicki M.: Evolutinoary multi-agent systems. *The Knowledge Engineering Review*, 2013 (accepted for printing).

[3] Byrski A., Kisiel-Dorohinicki M.: Agent-based evolutionary and immunological optimization. In *Computational Science – ICCS 2007, 7th International Conference, Beijing, China, May 27–30, 2007, Proceedings*. Springer, 2007.

[4] Byrski A., Kisiel-Dorohinicki M., Nawarecki E.: Agent-based evolution of neural network architecture. In Hamza M., editor, *Proc. of the IASTED Int. Symp.: Applied Informatics*. IASTED/ACTA Press, 2002.

[5] Byrski A., Schaefer R., Smołka M.: Asymptotic guarantee of success for multi-agent memetic systems. *Bulletin of the Polish Academy of Sciences–Technical Sciences*, (1), 2013.

[6] Byrski A., Debski R., Kisiel-Dorohinicki M.: Agent-based computing in an augmented cloud environment. *Computer Systems Science and Engineering*, 27(1), 2012.

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

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

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

[10] Dasgupta D., Nino L.: *Immunological Computation Theory and Applications*. Auerbach, 2008.

[11] Digalakis J., Margaritis K.: An experimental study of benchmarking functions for evolutionary algorithms. *International Journal of Computer Mathemathics*, 79(4): 403–416, April 2002.

[12] Dresner K., Stone P.: A multiagent approach to autonomous intersection management. *Journal of Artificial Intelligence Research*, 31:591–656, 2008.

[13] Dreżewski R.: Co-evolutionary multi-agent system with speciation and resource sharing mechanisms. *Computing and Informatics*, 25(4): 305–331, 2006.

[14] Dreżewski R., Sepielak J., Siwik L.: Classical and agent-based evolutionary algorithms for investment strategies generation. In Brabazon A., O'Neill M.,

editors, *Natural Computing in Computational Finance*, vol. 185 of *Studies in Computational Intelligence*, pp. 181–205. Springer-Verlag, 2009.

[15] George J., Gleizes M., P.Glize., C.Regis.: Real-time simulation for flood forecast: an adaptive multi-agent system staff. In *Proceedings of the AISB'03 Symposium on Adaptive Agents and Multi-Agent Systems*. University of Wales, 2003.

[16] Jennings N., Faratin P., Johnson M., Norman T., OBrien P., Wiegand M.: Agent-based business process management. *International Journal of Cooperative Information Systems*, 5(2–3): 105–130, 1996.

[17] Kisiel-Dorohinicki M.: Agent-oriented model of simulated evolution. In Grosky W. I., Plasil F., editors, *SofSem 2002: Theory and Practice of Informatics*, vol. 2540 of *LNCS*. Springer-Verlag, 2002.

[18] Lobel B., Ozdaglar A., Feijer D.: Distributed multi-agent optimization with state-dependent communication. *Mathematical Programming*, 129(2): 255–284, 2011.

[19] Mahalanobis P.: On the generalised distance in statistics. *Proceedings of the National Institute of Sciences of India*, 2(1): 49–55, 1936.

[20] McArthur S., Catterson V., Hatziargyriou N.: Multi-agent systems for power engineering applications part i: Concepts, approaches, and technical challenges. *IEEE Transactions On Power Systems*, 22(4), November 2007.

[21] Michalewicz Z.: *Genetic Algorithms Plus Data Structures Equals Evolution Programs*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1994.

[22] Moya L. J., Tolk A.: Towards a taxonomy of agents and multi-agent systems. In *Proceedings of the 2007 spring simulation multiconference – Vol. 2*, pp. 11–18, San Diego, CA, USA, 2007. Society for Computer Simulation International.

[23] Sarker R., Ray T.: *Agent-Based Evolutionary Search*, volume 5 of *Adaptation, Learning and Optimization*. Springer, 1 edition, 2010.

[24] Schaefer R., Kołodziej J.: Genetic search reinforced by the population hierarchy. *Foundations of Genetic Algorithms*, 7, 2003.

[25] Siwik L., Dreżewski R.: Agent-based multi-objective evolutionary algorithms with cultural and immunological mechanisms. In dos Santos W. P., editor, *Evolutionary computation*, pp. 541–556. In-Teh, 2009.

[26] Uhruski P., Grochowski M., Schaefer R.: A two-layer agent-based system for large-scale distributed computation. *Computational Intelligence*, 24(3): 191–212, July 2008.

[27] Wierzchoń S.: Function optimization by the immune metaphor. *Task Quaterly*, 6(3): 1–16, 2002.

[28] Wooldridge M.: *An Introduction to Multiagent Systems*. John Wiley & Sons, 2009.

[29] Wróbel K., Torba P., Paszyński M., Byrski A.: Evolutionary multi-agent computing in inverse problems. *Computer Science (accepted for printing)*, 2013.

[30] Zhong W., Liu J., Xue M., Jiao L.: A multiagent genetic algorithm for global numerical optimization. *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, 34(2): 1128–1141, 2004.

Affiliations

Aleksander Byrski
AGH University of Science and Technology, Department of Computer Science, Krakow, Poland, olekb@agh.edu.pl

Received: 8.03.2013
Revised: 21.03.2013
Accepted: 7.04.2013