Hierarchical and Massively Interactive Approaches for Hybridization of Evolutionary Computations and Agent Systems—Comparison in Financial Application

Leszek Siwik and Rafal Drezewski

AGH University of Science and Technology, Kraków, Poland,
{siwik,drezew}@agh.edu.pl

Abstract. When we think about hybridizing of evolutionary computations and agent systems in fact two approaches are possible: (1) hierarchical one – where agents are used as the management layer and the evolutionary algorithms are executed inside (sub)populations "within" agents and (2) system realized as the population(s) of evolving agents equipped with "DNA" performing life-steps to obtain their life-goals. In this paper we discuss aforementioned approaches and present their sample realization and application for solving a challenging portfolio optimization problem defined as a multi-objective optimization problem with maximization of the investment profit and minimization of the investment risk level.

1 Motivation

One of the promising computational techniques for solving hard and complex optimization problems (both global and local ones especially when the problem is defined as the multi-objective or multi-modal optimization problem) is applying nature-inspired systems and the evolutionary algorithms in particular since they are insensitive to the complexity of the problem to some extent.

The problem however is that evolutionary algorithm works properly (e.g. in terms of searching for a globally optimal solution) if the population consists of fairly different individuals, i.e. the so-called diversity in the population is preserved [2]. Yet many algorithms tend to prematurely loose this useful diversity and, as a result, there is possibility that population gets stuck in some part of the search space (e.g. in the basin of attraction of some local extrema instead of searching for a global one). Loosing the population diversity also limits the possibilities of the application in some areas such as multi-objective optimisation or multi-modal optimisation.

The above-described situation is related to the fact that the model of evolution employed by simple evolutionary algorithms lacks many important features observed in organic evolution [3]. This includes dynamically changing environmental conditions, neither global knowledge nor generational synchronisation assumed, co-evolution of species, evolving genotype-fenotype mapping, etc. That is why many variations of classical evolutionary algorithms were proposed, introducing additional mechanisms following the most important phenomena in evolutionary biology e.g. dedicated cooperation mechanisms [16], coevolutionary mechanisms [8–10], hierarchical approaches[6]

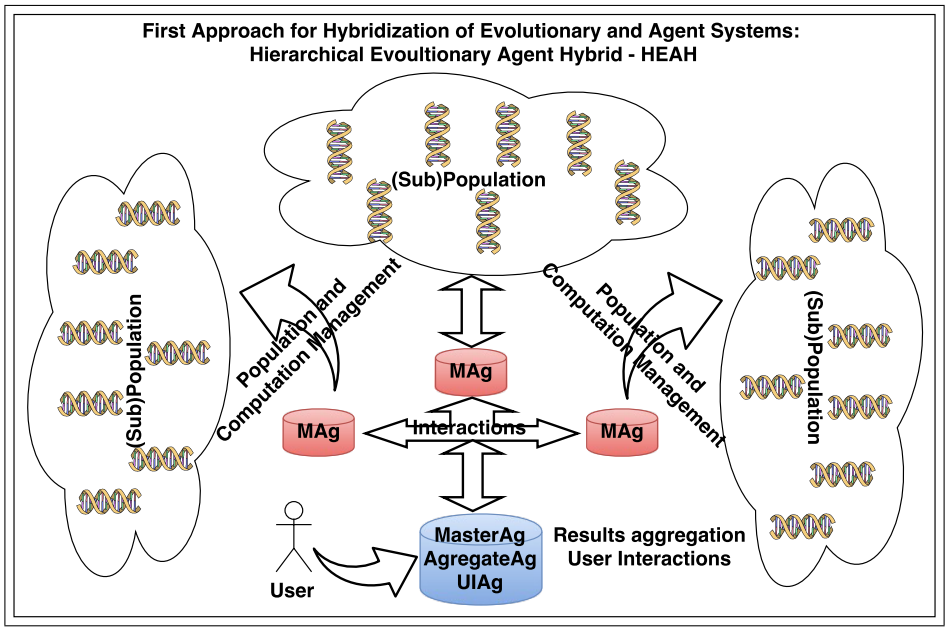

Fig. 1. Agent-based layer used for managing evolutionary computations

or converting problems into multiobjective optimization problems [15]. Yet still obtained results have been not satisfying in many cases.

During the last decades intelligent/autonomous software agents have been gaining more and more applications in various domains. The key concept in multi-agent systems (MAS) constitute intelligent interactions (coordination, cooperation, negotiation). Thus multi-agent systems are ideally suited for representing problems that have many solving methods, involve many perspectives and/or may be solved by many entities [17]. Agents play a key role in the integration of AI sub-disciplines, which often leads to hybrid design of modern intelligent systems.

Since evolutionary algorithms are distributed by nature and since agents are able to perform many complex operations it was then natural that the idea of hybridization of evolutionary computations with (multi)agent systems arouse.

In this paper two fundamental approaches for hybridizing evolutionary computation and agent systems i.e. (1) hierarchical approach (HEAH) with agents used as the management layer and (2) the population of evolving agents (MIEAH) equipped with the "DNA" and performing their "life steps" to obtain their "goals" (i.e. better and better solutions of the problem defined) are discussed, applied for solving challenging, discrete investment portfolio optimization defined as the multi-objective optimization problem, and then compared and concluded.

2 Two approaches for hybridization of evolutionary computations and agent systems

In most approaches for hybridization of evolutionary computations and agent systems reported in the literature (see e.g. [13] or [5] for a review) an evolutionary algorithm is used by an agent to aid realisation of some of its tasks, often connected with learning or reasoning, or to support coordination of some group (team) activity.

But when we think about constituting a new hybrid evolutionary-agent computational paradigm in fact two approaches are possible. In the first one agents constitute a management infrastructure for a distributed realisation of an evolutionary algorithm [14].

In such an approach (see fig. 1) each agent has the population of individuals inside of it, and this sub-population is evolving according to one of (classical) evolutionary algorithm. Agents themselves can migrate within the computational environment, from one computational node to another, trying to utilize in a best way, free computational resources.

In contrary, thinking about the hybridization of evolutionary and agent systems one may imagine the population(s) of evolving agents equipped with "DNA" performing life-steps to obtain their life-goals.

Such an idea with agents located in fixed positions on some lattice (like in a cellular model of parallel evolutionary algorithms) was developed by e.g. [18]. This approach yet interesting was disregarding important, powerful and crucial in facts features of agents i.e. their autonomy and mobility.

The full realization of the idea of incorporating evolutionary processes into a multi-agent systems at a population level regarding full autonomy of agents was the decentralised model of evolution employed by an *evolutionary multi-agent system* – EMAS [12].

Agents of EMAS represent or generate solutions for a given optimisation problem. They are located on islands, which constitute their local environment where direct interactions may take places, and represent a distributed structure of computation. Obviously, agents are able to change their location, which allows for diffusion of information and resources all over the system [12].

In EMAS phenomena of inheritance and selection – the main components of evolutionary processes – are modelled via agent actions of *death* and *reproduction* (see fig. 2). Inheritance is accomplished by an appropriate definition of reproduction, like in classical evolutionary algorithms. Core properties of the agent are encoded in its genotype and inherited from its parent(s) with the use of variation operators (mutation and recombination). Besides, an agent may possess some knowledge acquired during its life, which is not inherited. Both inherited and acquired information determines the behaviour of an agent in the system (phenotype).

Assuming that no global knowledge is available (which makes it impossible to evaluate all individuals at the same time) and autonomy of the agents (which causes that reproduction is achieved asynchronously), selection is based on the non-renewable resources [4].

In order to realize the selection process “better” (what means that they simply better solve the given problem) agents are given more resources from the environment (or

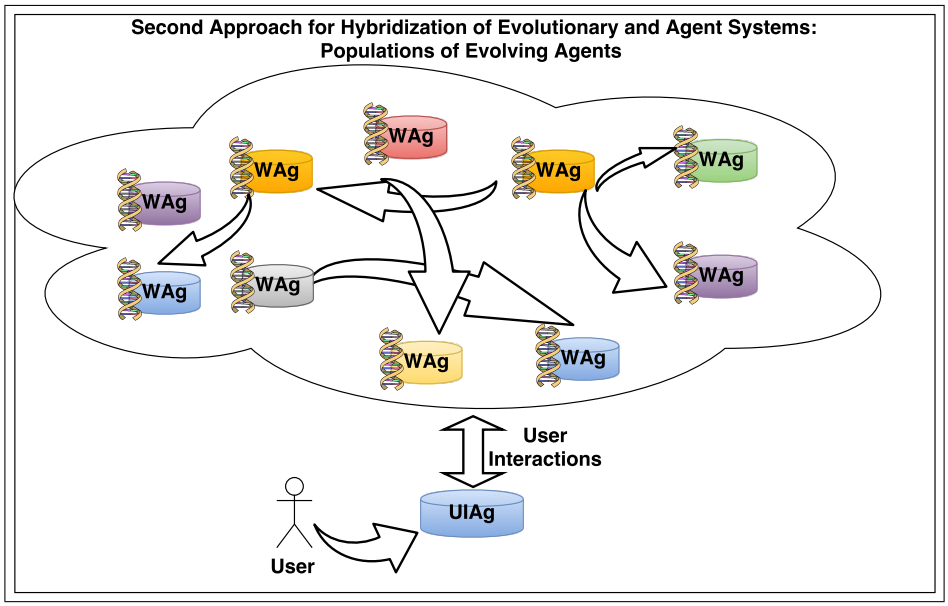

Fig. 2. Evolutionary multi-agent system—population of evolving agents with "DNA" performing life-steps to obtain their life-goals

from other agents) and “worse” agents are given less resources (or should give some of its resources to “better” agents). Such mechanisms result in decentralized evolutionary processes in which individuals (agents) make independently all their decisions concerning reproduction, migration, interactions with other agents, etc., taking into consideration conditions of the environment, other agents present within the neighborhood, and resources possessed.

3 Realization of hierarchical and interactive evolutionary-agent hybrid systems

3.1 Realization of hierarchical evolutionary-agent hybrid system

Hierarchical approach has been implemented using Age—agent-oriented framework [1] (its Java implementation i.e. jAge in fact). The framework supported the authors with implementation of a notion of working agent that was adapted to create an efficient implementation of Master and Slave agents.

Because the representatives of each sub-populations had to be aggregated (in order to form the complete solution) and also because of the necessity of storing the complete non-dominated solutions the system consists of one Master/Aggregate and many slaves/working agents. Master agent is responsible for exchanging information with external world e.g. the user. It is also responsible for forming complete solutions (composed of the representatives of each sub-population and evaluation of the solutions. It

also maintains the set of non-dominated solutions found so far (the definition of domination relation and other issues connected with the Pareto approach to multi-objective optimization can be found for example in [7] or [11]). Each sub-population is responsible only for the selected part of solution, and evolved by one working agent.

Master agent has to deal also with typical management tasks i.e. it is responsible for dispatching optimization tasks among Slaves/Working agents. Working agents can manage the populations evolving according to different algorithms. For experiments, working agents managed the (sub)population of individuals being evolved according to NSGA2 evolutionary algorithm for multi-objective optimization ([7]).

As a result of integration of agent system and NSGA2 ([7]) algorithm the agent-based co-operative version of NSGA2 was created. Thanks to the computed contribution of the given individual to the quality of the complete solution, the fitness computation in agent-based co-evolutionary NSGA2 is realized with the use of non-dominated sorting and crowding distance metric (see [7]). Additionally, the aggregate agent joins the populations of parents and offspring, and chooses (on the basis of elitist selection and within each sub-population separately) individuals which will form the next generation sub-population used for the creation of complete solutions. The applied schema implies that N best (according to non-dominated sorting and crowding distance metric) individuals survive.

3.2 Predator-prey co-evolutionary multi-agent system as the realization of massively interactive approach

As it was stated in section 1 one of two main approaches for hybridization of evolutionary and agents systems is equipping agents with their "DNA" and constructing populations of evolving agents, "living" in their environment, interacting with the other agents realizing their own goals defined (usually) as obtaining the best possible approximations of optimal solutions of single- or multi-objective, local or global optimization problem(s).

Obviously the "live-step" of evolving agents as well as their interaction can be defined in many possible ways—from very simple until complex, respecting many possible species and nations of agents.

One of possible realization is the system of co-evolving (arm-racing in fact) two species of agents: predators and preys which is called the co-evolutionary multi-agent system with predator-prey interactions (*PPCoEMAS*) which is generally discussed in this section (the formal model and detailed presentation of co-evolutionary multi-agent system with predator-prey interactions is given in [11]).

According the the general description of evolutionary multi-agent system given in the section 2 the system is composed of environment with graph-like structure, interacting agents and resources. There are two species of agents: predators (their goal is to remove less fitted prey agents) and preys (which represent solutions of multi-objective problem). Agents exist within the environment, they can migrate from node to node (if only there exists connection between nodes and agent has enough resource). Resources (which are possessed only by agents—there is no resource within the environment itself) are used for every activity like migration and reproduction. Agents without resources die and are removed from the system.

Agents of prey species can reproduce when they have sufficient amount of resource. When two ready for reproduction prey agents meet within the same node they reproduce—new agent is created with the use of intermediate recombination and mutation with self-adaptation operators (floating point representation is used). Some amount of resource is transferred from parents to the newly created offspring.

Predators do not reproduce. They can only migrate within the environment and seek for less fitted preys. Each predator has one criteria associated with it (it is encoded within predator's genotype) and it uses this criteria to seek for the worst prey that is located within the same node. Then predator takes all resources from the chosen prey, which dies as a result of this action.

The whole amount of resource within the system is constant—resource is possessed only by predators and preys. As a result of interactions between agents the resource may be transferred from prey to predator (predator-prey interaction) and from prey to another prey (prey-prey interaction).

4 Problem formulation

As it was stated: the goal of this paper is to compare two approaches for hybridization of evolutionary and agent-based computational paradigms run against the problem of building effective portfolio.

The first question to be answered is how such a problem should be formally defined or which well-known definition should be chosen. Practically, there are some well known models describing building of effective portfolio i.e. Modern Portfolio Theory (MPT), one-factor Sharpe model, CAPM—Capital Asset Pricing Model, APT—Arbitrage Pricing Theory, Post Modern Portfolio Theory (PMPT) etc.

Taking all the pros and cons into consideration—one-factor Sharpe model has been chosen for our experiments so it is discussed below more precisely.

According to one-factor Sharpe model the algorithm of computing the expected risk level and income expectation related to the portfolio of p assets is formulated as in alg. 1.

The meanings of the symbols used in alg. 1 are as follows:

p is the number of assets in the portfolio;
n is the number of periods taken into consideration (the number of rates of return taken to the model);
α_i,β_i are coefficients of the equations;
ω_i is percentage participation of i-th asset in the portfolio;
e_i is random component of the equation;
R_{it} is the rate of return in the period t;
R_{mt} is the rate of return of market index in period t;
R_m is the rate of return of market index;
R_i is the rate of return of the i-th asset;
R_p is the rate of return of the portfolio;
${s_i}^2$ is the variance of the i-th asset;
${s_{e_i}}^2$ is the variance of the random index of the i-th asset;
${s_{e_p}}^2$ is the variance of the portfolio;

Algorithm 1. The algorithm (based on the one-factor Sharpe model) of computing the expected risk level and income expectation

1 Compute the arithmetic means on the basis of rate of returns;
2 Compute the value of α coefficient $\alpha_i = \overline{R_i} - \beta_i \overline{R_m}$;
3 Compute the value of β coefficient $\beta_i = \frac{\sum_{t=1}^{n}(R_{it}-\overline{R_i})(R_{mt}-\overline{R_m})}{\sum_{t=1}^{n}(R_{mt}-\overline{R_m})^2}$;
4 Compute the expected rate of return of asset i $R_i = \alpha_i + \beta_i R_m + e_i$;
5 Compute the variance of random index ${s_{e_i}}^2 = \frac{\sum_{t=1}^{n}(R_{it}-\alpha_i-\beta_i R_m)^2}{n-1}$;
6 Compute the variance of market index ${s_m}^2 = \frac{\sum_{t=1}^{n}(R_{mt}-\overline{R_m})^2}{n-1}$;
7 Compute the risk level of the investing portfolio $\beta_p = \sum_{i=1}^{p}(\omega_i \beta_i)$;
8 ${s_{e_p}}^2 = \sum_{i=1}^{p}(\omega_i^2 {s_{e_i}}^2)$;
9 $risk = \beta_p^2 {s_m}^2 + {s_{e_p}}^2$;
10 Compute the portfolio rate of return $R_p = \sum_{i=1}^{p}(\omega_i R_i)$;

$\overline{R_i}$ is arithmetic mean of rate of return of the i-th asset;
$\overline{R_m}$ is arithmetic mean of rate of return of market index;

The goal of the optimization is to maximize the portfolio rate of return and minimize the portfolio risk level. The task consists in determining values of decision variables $\omega_1 \ldots \omega_p$ forming the vector $\Omega = [\omega_1, \ldots, \omega_p]^T$, where $0\% \leq \omega_i \leq 100\%$ and $\sum_{i=1}^{p} \omega_i = 100\%$ and $i = 1 \ldots p$ and which is the subject of minimization with respect of two criteria $F = [R_p(\Omega) * (-1), risk(\Omega)]^T$.

We gain a classical multiobjective optimization problem the **M**ultiobjective **O**ptimization of **I**nvesting **P**ortfolio **P**roblem (**MOIPP**) with two contradictory objectives the risk and expected income which can be formulated as follows:

$$MOIPP \equiv \begin{cases} Max: \ R_p = \sum_{i=1}^{p}(\omega_i R_i) \\ Min: risk = \beta_p^2 {s_m}^2 + {s_{e_p}}^2 \\ Taking\ into\ consideration: \\ \ R \geq 0 \ \ and\ risk \geq 0 \\ \sum_{i=1}^{p} \omega_i = 100\% \\ \ 0\% \leq \omega_i \leq 100\% \ \ and \ \ i = 1 \ldots p \end{cases}$$

In the course of this paper multi-objective optimization in the Pareto sense is considered, so solving defined MOIPP problem means determining of all feasible and non-dominated alternatives from the set ($\mathcal{D}$). Such defined set is called Pareto set ($\mathcal{P}$) and in objective space it forms so called Pareto frontier ($\mathcal{PF}$).

5 Results

Defined portfolio optimization problem has been solved using the hierarchical evolutionary multi-agent system discussed in section 3.1 and the massively interactive evolutionary multi-agent system with predator-prey interactions discussed in section 3.2.

Fig. 3. Pareto frontier approximations after 1000 steps obtained by mixed hierarchical approach and massively interactive approach with predator-prey mechanisms for building effective portfolio consisting of 3 and 17 stocks

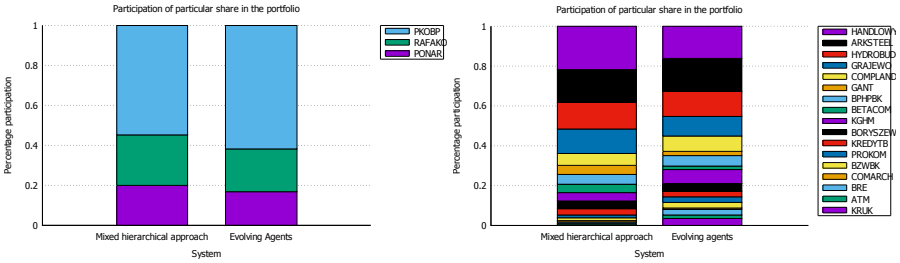

Fig. 4. Effective portfolio consisting of three and seventeen stocks found by hierarchical approach and massively interactive approach i.e. coevolving agents with predator-prey interactions

Each individual evolved during experiments has been represented as a p-dimensional vector. Each dimension represents the percentage participation of i-th ($i \in 1 \ldots p$) share in the whole portfolio.

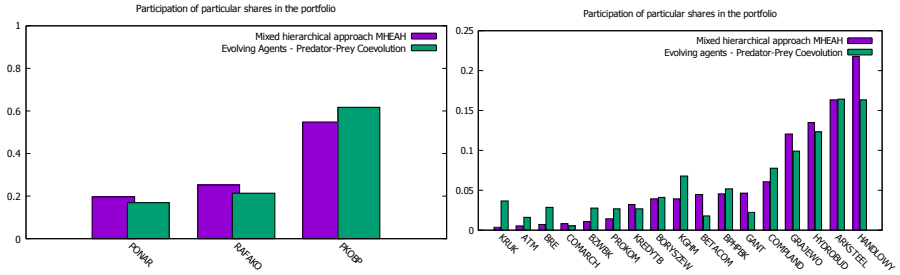

Fig. 5. Effective portfolio consisting of three and seventeen stocks found by hierarchical approach and massively interactive approach i.e. coevolving agents with predator-prey interactions

During presented experiments—Warsaw Stock Exchange quotations from 2003-01-01 until 2005-12-31 were taken into consideration. Simultaneously, the portfolio consists of the following three (experiment I) or seventeen (experiment II) stocks quoted on the Warsaw Stock Exchange: in experiment I: RAFAKO, PONARFEH, PKOBP, in experiment II: KREDYTB, COMPLAND, BETACOM, GRAJEWO, KRUK, COMARCH, ATM, HANDLOWY, BZWBK, HYDROBUD, BORYSZEW, ARKSTEEL, BRE, KGHM, GANT, PROKOM, BPHPBK. As the market index, WIG20 has been taken into consideration.

In figure 3 the sample approximation of Pareto frontiers (i.e. sets of non-dominated solutions) for both compared evolutionary-agent hybridized systems are presented.

As one may see in first–simpler–experiments consisting in looking for the optimal portfolio consisting of 3 stocks both hybridization approaches have been able to obtain a similar and comparable portfolios taking into consideration defined objectives i.e. expected (maximized) profit and (minimized) investment risk. In particular both compared approaches have been able to obtain portfolios with very similar level of profit and risk in the first one-third part of the Pareto frontier. Also both systems located the majority of final non-dominated individuals in this part of the Pareto frontier. The second two-third part of the frontier is visibly worse probed. Also the difference between obtained approximation of the Pareto frontiers is slightly bigger in this part of the frontier.

The model Pareto frontier for the problem defined is evenly dense on its full extent. Since the high-quality solution of the multi-objective optimization problem in the Pareto sense is the set of non-dominated solutions spread over the full extent of the Pareto frontier–concentration of found non-dominated solution in the first one-third part of the frontier observed in figure 3 in both cases—it is for sure the space for further improvements. Anyway, since the goal of this paper is to compare hierarchical and massively interactive hybrids of evolutionary and multi-agent computational paradigms, it can be said that both approaches are comparable taking the quality of obtained results into account.

Similar situation can be observed when we look at the figure 3(b) presenting pareto frontiers approximation obtained by both systems when the portfolio consisting of 17 stocks has been optimized. Also in this case both approaches concentrated their solution in the first one-third part of the Pareto frontier. But this time also the model Pareto frontier (not presented here because) is concentrated in this area.

What is interesting, there is as the matter of fact, some slight 'shift' between pareto frontier approximation obtained by compared hybridization approaches and the massively interactive approach has been able to cover slightly better the first half of the frontier whereas the hierarchical approach covered slightly better its second half–what gives for sure the space for further improvements. But again it can be said for sure that both approaches have been able to obtain a really close approximation of the model Pareto frontiers and obtained sets of non-dominated solutions are pretty close and similar.

From the financial point of view it is interesting how (financially) effective is the portfolio proposed by both approaches as the optimal one. Obviously, since we are in the space of multiobjective optimization in the Pareto sense the solution is not the one, single optimal solution but the whole set of non-dominated alternatives. Anyway, in the figure 5 the comparison of non-dominated portfolios found by both systems and located closely on the frontiers diagram are presented.

As one may see obtained portfolios are really close and similar especially in the first–simpler experimental case (optimizing portfolio consisting of 3 stocks). In both cases the main part of proposed portfolio is PKOBP what is the biggest bank in Poland what is absolutely natural and expected. Probably every single human-being investor working without any computational tools would also build his portfolio around stable and profitable banking institution.

Obviously, analyzing the portfolio consisting of 17 stocks the greater variety can be observed nevertheless the general trends are also really close and similar in both cases.

For easier analysis, selected, found, non-dominated portfolios presented in figure 5(a) and (b) are presented from different perspective as the percentage share of the portfolio in figure 4. Also in this case it is clear that both evolutionary multi-agent hybridized systems have been able to find reasonable and similar (non-dominated) portfolios.

6 Summary and Conclusions

One of the promising computational techniques for solving hard and complex optimization problems (both global and local ones especially when the problem is defined as the multi-objective or multi-modal optimization problem) is applying nature-inspired systems and the evolutionary algorithms in particular since they are insensitive to the complexity of the problem to some extent.

The problem however is the premature loose of population (and solution) diversity and, as a result getting stuck in the basin of attraction of some local extrema instead of searching for a global one. The solution may be the hybridization of evolutionary algorithms with agent systems since the autonomy, mobility and generally saying the "intelligence" of agents may prevent the evolution from getting stuck.

When we think about hybridizing of evolutionary computations and agent systems in fact two approaches are possible: (1) hierarchical one – where agents are used as the management layer and the evolutionary algorithms are executed inside (sub)populations "within" agents and (2) system realized as the population(s) of evolving agents equipped with "DNA" performing life-steps to obtain their life-goals.

The analysis of any economical and financial phenomena is extremely complex and difficult mainly because of many-dimensional relations and dependencies among particular components and participants of the market-game. No wonder so that it is also so difficult to develop really efficient and effective algorithms and computer systems supporting modeling, analyzing and finally—solving market oriented problems. In the consequence the systems for financial or economical modeling and analysis are more and more complex and complicated.

As one may see on presented experimental comparison, there is for sure the space for further improvements, anyway, the general conclusion coming from the comparison is that both systems realizing two different approaches for hybridization of evolutionary computations and agent systems turned out to be comparably effective obtaining similar sets of non-dominated portfolios.

Acknowledgments

The research presented in this paper was partially supported by the AGH University of Science and Technology Statutory Fund no. 11.11.230.124.

References

1. In A. Byrski, Z. Oplatková, M. Carvalho, and M. Kisiel-Dorohinicki, editors, *Advances in Intelligent Modelling and Simulation*, volume 416 of *Studies in Computational Intelligence*. 2012.

2. T. Bäck, D. Fogel, and Z. Michalewicz, editors. *Handbook of Evolutionary Computation*. IOP Publishing and Oxford University Press, 1997.
3. T. Back, U. Hammel, and H.-P. Schwefel. Evolutionary computation: Comments on the history and current state. *IEEE Trans. on Evolutionary Computation*, 1(1), 1997.
4. K. Cetnarowicz, M. Kisiel-Dorohinicki, and E. Nawarecki. The application of evolution process in multi-agent world (MAW) to the prediction system. In M. Tokoro, editor, *Proc. of the 2nd Int. Conf. on Multi-Agent Systems (ICMAS'96)*. AAAI Press, 1996.
5. S.-H. Chen, Y. Kambayashi, and H. Sato. *Multi-Agent Applications with Evolutionary Computation and Biologically Inspired Technologies*. IGI Global, 2011.
6. E. Ciepiela, J. Kocot, L. Siwik, and R. Dreżewski. Hierarchical approach to evolutionary multi-objective optimization. In M. Bubak, G. D. van Albada, J. Dongarra, and P. M. A. Sloot, editors, *Computational Science — ICCS 2008*, volume 5103 of *LNCS*, pages 740–749, Berlin, Heidelberg, 2008. Springer-Verlag.
7. K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, 2001.
8. R. Dreżewski and L. Siwik. Co-evolutionary multi-agent system with sexual selection mechanism for multi-objective optimization. In *Proceedings of the IEEE World Congress on Computational Intelligence (WCCI 2006)*. IEEE, 2006.
9. R. Dreżewski and L. Siwik. Multi-objective optimization using co-evolutionary multi-agent system with host-parasite mechanism. In V. N. Alexandrov, G. D. van Albada, P. M. A. Sloot, and J. Dongarra, editors, *Computational Science — ICCS 2006*, volume 3993 of *LNCS*, pages 871–878, Berlin, Heidelberg, 2006. Springer-Verlag.
10. R. Dreżewski and L. Siwik. The application of agent-based co-evolutionary system with predator-prey interactions to solving multi-objective optimization problems. In *Proceedings of the 2007 IEEE Symposium Series on Computational Intelligence*. IEEE, 2007.
11. R. Dreżewski and L. Siwik. Co-evolutionary multi-agent system for portfolio optimization. In A. Brabazon and M. O'Neill, editors, *Natural Computation in Computational Finance*, pages 271–299. Springer-Verlag, Berlin, Heidelberg, 2008.
12. M. Kisiel-Dorohinicki. Agent-oriented model of simulated evolution. In W. I. Grosky and F. Plasil, editors, *SofSem 2002: Theory and Practice of Informatics*, volume 2540 of *LNCS*. Springer, 2002.
13. R. Sarker and T. Ray. *Agent-Based Evolutionary Search*. Springer, 2010.
14. R. Schaefer and J. Kołodziej. Genetic search reinforced by the population hierarchy. *Foundations of Genetic Algorithms*, 7, 2003.
15. L. Siwik and R. Dreżewski. Evolutionary multi-modal optimization with the use of multi-objective techniques. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, and J. M. Zurada, editors, *Artificial Intelligence and Soft Computing*, volume 8467 of *Lecture Notes in Computer Science*, pages 428–439. Springer International Publishing, 2014.
16. T. Wójtowicz, K. Rzecki, P. Pławiak, M. Niedźwiecki, T. Sośnicki, K. Smelcerz, and Z. Wojtoń, Tomasz amd Tabor. Emergence of cooperation as a result of mutation and inheritance in pd/pg-like game. *Technical Transactions, Fundamental Sciences*, 18(1-NP/2015):71–84, 2015.
17. M. Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons, 2009.
18. W. Zhong, J. Liu, M. Xue, and L. Jiao. A multiagent genetic algorithm for global numerical optimization. *IEEE Trans. on Systems, Man, and Cybernetics, Part B: Cybernetics*, 34(2):1128–1141, 2004.