Computing and Informatics, Vol. 36, 2017, 331–352, doi: 10.4149/cai_2017_2_331

CLASSIC AND AGENT-BASED EVOLUTIONARY HEURISTICS FOR SHAPE OPTIMIZATION OF ROTATING DISCS

Wojciech Korczynski, Aleksander Byrski, Roman Dębski
Marek Kisiel-Dorohinicki

Department of Computer Science
AGH University of Science and Technology
Al. Mickiewicza 30
30-059 Krakow, Poland
e-mail: {wojciech.korczynski, olekb, roman.debski, doroh}@agh.edu.pl

Abstract. The article presents a metaheuristic solution for the problem of shape optimization of a rotating annular disc. Such discs are important structural components of e.g. jet engines, steam turbines or disc brakes. The design goal is to find the disc shape that would ensure its maximal carrying capacity (corresponding to the speed of rotation), which is a variational problem with the objective functional defined by L_∞ norm. Such a definition makes the problem impossible to solve using analytical methods so utilization of metaheuristics is necessary. We present different algorithms to solve the problem starting with a classic evolutionary one, followed by agent-based and hybrid agent-based memetic algorithms, which are the main focus of this paper. The reason for this is that agent-based computing systems proved to be versatile as an optimization technique being especially efficient for the problems with complex fitness functions. The obtained experimental results encourage further application of such an approach to similar engineering problems.

Keywords: Agent-based computing, evolutionary computing, variational problem, metaheuristics, global optimisation

Mathematics Subject Classification 2010: 68U20

1 INTRODUCTION

Complex optimization problems, especially the ones connected with engineering applications, very often prove to be so difficult that their analysis becomes a non-trivial task for most traditional methods. In such cases simulation experiments may be useful to verify the correctness of the proposed solutions, which may be further improved (optimized) by the human designer. However, the globally optimal solution is hard to find this way. This task may be successfully handled by some (meta)heuristic computational techniques such as evolutionary algorithms. This term covers a wide range of search and optimization methods, based on analogies to phenomena of natural evolution [38].

However, one main drawback of employing evolutionary algorithms to solve such problems is their intrinsic feature: they process a number of individuals, and the application of selection and variation operators changing the structure of the population imposes an immense number of computations of fitness function until an acceptable solution is found [27]. When evaluation of fitness for individual solutions has a significant computation cost, as in the case of simulation model described in this work, other means of metaheuristic computing may be needed in order to decrease the total cost of the process. The problem becomes even more significant when additional hybrid techniques (such as e.g. memetic computing [40]) are used.

One of interesting general-purpose metaheuristics that require less computation effort than classic evolutionary algorithms [6] is an agent-based model of evolutionary computation, namely EMAS – Evolutionary Multi-Agent System [2]. EMAS has already proven to be an efficient method in solving different problems, from classic benchmarks [7] to inverse problems [52] and other optimization tasks (see e.g. [20, 19]).

The problem discussed here is the optimization of rotating variable-thickness annular elastic discs based on the proposed simulation model. Rotating discs are important structural components, which can be found for instance in jet engines, steam turbines or disc brakes. The design goal is to find a disc shape that would ensure its maximal carrying capacity (corresponding to the speed of rotation). This is an example of a variational problem with the objective functional defined by L_∞ norm (see Section 2). Optimal shape design problems are often solved by modeling the shape with a fixed class of functions (e.g. hyperbolic, n^{th} order polynomial, see e.g. [17, 1, 25, 41]). In the approach presented here the profile of the disc is defined by spline curves, described by a set of parameters. This introduces many more 'degrees of freedom', which makes it possible to find a more suitable shape [16, 15]. The proposed algorithm is direct (i.e. the Euler-Lagrange equation is not used) – a series of disc shapes is evaluated through simulation. In each simulation the speed of rotation is increased until the maximum value of the observed stress intensity reaches the value of the yield stress. In such a state the disc elastic carrying capacity is exhausted (see Section 2). This approach is similar to the one used by Schwefel and Rechenberg during their early works on Evolution Strategies [49], though their method was experiment-based, ours is simulation-based.

The paper is devoted to experimental verification of the possibility of application of EMAS to solve the variational optimization problems, utilizing one of the important features of EMAS, namely significantly lower time cost measured by the number of fitness function calls required to yield similar results, compared with e.g. parallel evolutionary algorithm (PEA) [7]. Moreover, EMAS was proven to be a general-purpose optimizing method [5].

Based on the defined problem, the authors wanted to compare EMAS with as similar as possible not-agent-based algorithms. It was quite hard to find proper competitor for EMAS, since it is unique, one-of-a-kind system. Several algorithms were considered and evolutionary algorithm constructed according to Michalewicz model [38] turned out to be the most suitable for our needs.

The article begins with a careful formulation of the tackled problem (Section 2). Next, the EMAS structure and features are described, concerning its classical, evolutionary version, as well as its memetic version (Section 3). In Section 4, experimental results are discussed, presenting both qualitative and quantitative results regarding the computing process itself, as well as obtained solutions (profile of the rotating disc). The experimental study is followed by conclusions (Section 5).

2 SHAPE OPTIMIZATION OF ROTATING DISKS

The optimal shape design of rotating discs is a classic one in mechanical engineering. However, most of the classic papers are related to analytical methods for some special (and often very simple) cases like, for instance, flat or hyperbolic discs. Some more recent papers related to numerical or simulation-based optimization are reviewed below.

Berger and Porat [1] analyzed a thin homogeneous rotating disk of variable thickness, considered for the purpose of storing kinetic energy. The objective of the design was to find the optimal shape of the disk for which the Specific Kinetic Energy was maximal.

Decohesive carrying capacity (DCC) of variable-thickness annular perfectly plastic disc was analyzed by Dębski and Życzkowski [17]. They showed that DCC may occur not only at the boundary of the disk but also at a certain point inside the disk. This fact can be crucial during simulation-based optimization. They also discussed in detail the problem of the disk of uniform decohesive carrying capacity.

The disc of uniform carrying capacity was also investigated by Genta and Bassani [25]. The optimization process proposed by the authors was based on genetic algorithms.

Simulation-based optimization of elastic carrying capacity of rotating variable-thickness annular elastic discs by means of classic $(\mu + \lambda)$ and (μ, λ)) evolutionary strategies was discussed by Dębski et al. [16]. The profile of the disc was defined by spline curves.

Mechanisms for maintaining population diversity in (μ, λ)-evolution strategies were investigated by Dębski et al. [15]. The proposed mechanisms (deterministic modification of standard deviations, crowding and elitism) were experimentally verified with the use of the optimal shape design of rotating variable-thickness annular disc problem.

The most important assumptions regarding the physical model of the design problem under consideration are as follows [16]:

- We consider an annular elastic disc of variable thickness $h = h(r)$ rotating with constant angular velocity ω and subject to uniform traction p_b at the outer radius b. The disc is clamped at the inner radius a (see Figure 1).
- The classical theory of thin discs with small gradient dh/dr is assumed and hence the stresses τ_{ry} and σ_y are negligible.
- The material is linear-elastic with Young's modulus E, Poisson's ratio ν and subject to the Huber-Mises-Hencky (H-M-H) yield condition.
- The small-strain theory is applied.

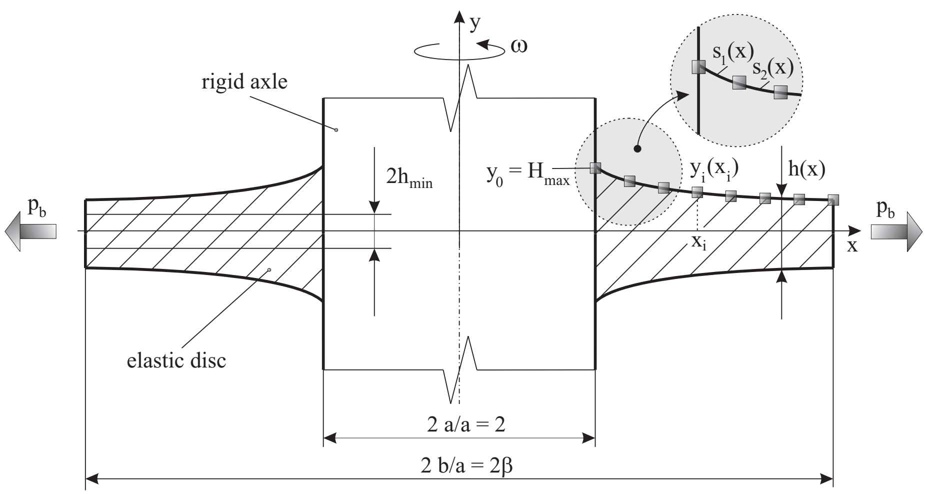


Figure 1. Profile representation and constraints of the shape of the annular disc under consideration. The profile of the disc is represented by the 3^{rd} order spline built on equidistant nodes (note: $x = r/a$).

The condition of internal equilibrium in polar coordinates may be then expressed as follows:

$$\frac{1}{h}\frac{\mathrm{d}}{\mathrm{d}r}(h\sigma_r) + \frac{\sigma_r - \sigma_\theta}{r} + \rho_m \omega^2 r = 0 \tag{1}$$

where ρ_m stands for mass density and h for the disc thickness. Constitutive equations (described be Hooke's Law) take the following form:

$$\begin{cases} \sigma_r = \frac{E}{1-\nu^2}\left(\frac{\mathrm{d}u}{\mathrm{d}r} + \nu\frac{u}{r}\right), \\ \sigma_\theta = \frac{E}{1-\nu^2}\left(\frac{u}{r} + \nu\frac{\mathrm{d}u}{\mathrm{d}r}\right). \end{cases} \tag{2}$$

They combine radial displacement u with radial σ_r and circumferential σ_θ stresses. Making use of constitutive Equations (2), after simple calculations the condition of internal equilibrium takes the form:

$$\frac{\mathrm{d}^2u}{\mathrm{d}r^2} + \frac{1}{r}\left(1 + \frac{r}{h}\frac{\mathrm{d}h}{\mathrm{d}r}\right)\frac{\mathrm{d}u}{\mathrm{d}r} - \frac{1}{r^2}\left(1 - \nu\frac{r}{h}\frac{\mathrm{d}h}{\mathrm{d}r}\right)u = -C\omega^2 r \tag{3}$$

where $C = \rho_m(1-\nu^2)/E$. Boundary conditions

$$\begin{cases} u(a) = 0, \\ \frac{E}{1-\nu^2}\left(\frac{\mathrm{d}u(b)}{\mathrm{d}r} + \nu\frac{u(b)}{b}\right) = p_b \end{cases} \tag{4}$$

allow us to find a numerical solution depending on external loadings (angular velocity ω and uniform traction p_b). The stress intensity, calculated according to Huber-Mises-Hencky hypothesis

$$\sigma_i^2 = \sigma_r^2 + \sigma_\theta^2 - \sigma_r\sigma_\theta \tag{5}$$

takes its maximal value at the boundary of the disc (usually at its inner radius) or at a certain point r_o inside the disc. It obviously depends on the shape of the disc (see [17]). When the maximum value of the stress intensity reaches the value of yield stress σ_0:

$$\max_{a\le r\le b}\sigma_i(r) = \sigma_0 \tag{6}$$

the elastic carrying capacity is exhausted. Therefore, Equation (6) allows us to find the external loadings value (in general a combination of angular velocity ω and uniform traction p_b), which we call the *elastic carrying capacity* of the disc.

As mentioned above, the profile of the disc is represented by the 3$^{\text{rd}}$ order spline (see Figure 1) built on equidistant nodes (with δ being the distance between nodes). Coefficients of the spline may be found from the set of linear equations:

$$\begin{bmatrix} 2 & 1 & 0 & \dots & 0 \\ 1 & 4 & 1 & \dots & 0 \\ 0 & 1 & 4 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \dots & 1 & 4 & 1 \\ 0 & \dots & 0 & 1 & 2 \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ \vdots \\ M_{n-1} \\ M_n \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ \vdots \\ d_{n-1} \\ d_n \end{bmatrix} \tag{7}$$

where

$$d_j = \frac{6}{\delta^2}\left(y_{j-1} - 2y_j + y_{j+1}\right), \quad j = 1,\dots,n-1, \tag{8}$$

$$\begin{cases} d_0 = \frac{6}{\delta^2}\left(y_1 - y_0\right), \\ d_n = \frac{6}{\delta^2}\left(y_n - y_{n-1}\right). \end{cases} \tag{9}$$

After some calculations we may find the value of the interpolating function as well as its derivative at any point from the range $r \in [a, b]$ or, after introducing dimensionless radius $x = r/a$, $x \in [1, b/a]$:

$$
\begin{aligned}
s(x) = {} & \frac{1}{6\delta}\left[M_{j-1}\left(x_j - x\right)^3 + M_j\left(x - x_{j-1}\right)^3\right] \\
& + \left[\frac{y_j - y_{j-1}}{\delta} - \frac{\delta}{6}\left(M_j - M_{j-1}\right)\right]\left(x - x_{j-1}\right) \\
& + y_{j-1} - M_{j-1}\frac{\delta^2}{6},
\end{aligned} \tag{10}
$$

$$
\begin{aligned}
\frac{\mathrm{d}s(x)}{\mathrm{d}x} = {} & \frac{1}{2\delta}\left[-M_{j-1}\left(x_j - x\right)^2 + M_j\left(x - x_{j-1}\right)^2\right] \\
& + \frac{y_j - y_{j-1}}{\delta} - \frac{\delta}{6}\left(M_j - M_{j-1}\right)
\end{aligned} \tag{11}
$$

where $x \in [x_{j-1}, x_j]$ and $j = 1, \ldots, n$.

For numerical calculations we introduce the following dimensionless quantities:

$$
\beta = \frac{b}{a}, \quad x = \frac{r}{a}, \quad y = \frac{h}{2a}, \quad w = \frac{u}{a}. \tag{12}
$$

The above group of parameters is related only to the geometry of the disc while the following one describes the material of the disc and its external loadings:

$$
S = \frac{\sigma_0}{E}, \quad \Omega = \frac{\sqrt{3}\rho_m \omega^2 a^2}{2\sigma_0}, \quad p = \frac{p_b}{\sigma_0}. \tag{13}
$$

We assume that external loading p (i.e. uniform traction) is a function of Ω

$$
p = f(\Omega) = C\ \Omega = \frac{3 + \nu}{4 + \sqrt{3}}\beta^2 \frac{y_0}{y_n}\ \Omega. \tag{14}
$$

Substitution of Equations (12), (13) and (14) into Equations (3) and (4) gives the final form of the rotating disc simulation model

$$
\frac{\mathrm{d}^2 w}{\mathrm{d}x^2} + \frac{1}{x}\left(1 + \frac{x}{y}\frac{\mathrm{d}y}{\mathrm{d}x}\right)\frac{\mathrm{d}w}{\mathrm{d}x} - \frac{1}{x^2}\left(1 - \nu\frac{x}{y}\frac{\mathrm{d}y}{\mathrm{d}x}\right)w = -\frac{2}{\sqrt{3}}\left(1 - \nu^2\right)S\Omega x, \tag{15}
$$

$$
\begin{cases}
w(1) = 0, \\
\frac{1}{S(1-\nu^2)}\left(\frac{\mathrm{d}w}{\mathrm{d}x}(\beta) + \nu\frac{w(\beta)}{x}\right) = \frac{3+\nu}{4+\sqrt{3}}\beta^2\frac{y_0}{y_n}\Omega.
\end{cases} \tag{16}
$$

Finally, we can formulate the optimization problem by defining a decision variable vector, a feasible region and an objective function. The decision variables vector

$$
\boldsymbol{y} = (y_0, y_1, \ldots, y_n) \in \mathcal{D} \subset \mathbb{R}^{n+1} \tag{17}
$$

represents the shape of the disc in $n + 1$ equidistant points. The feasible region is defined as follows:

$$\mathcal{D} = \left\{ \ \boldsymbol{y} \in \mathbb{R}^{n+1} \mid [y_0 = H_{max} \ \wedge \ (h_{min} \leq y_j \leq H_{max}, \forall j = 1, \ldots, n)] \ \right\}. \qquad (18)$$

Lastly, after the introduction of a dimensionless angular velocity (speed of rotation) $\bar{\Omega}$ corresponding to the disc elastic carrying capacity (expressed by Equation (6)), we can formulate the objective function in the following manner (i.e. the design (optimization) goal is to find the disc shape that would ensure its maximal carrying capacity (corresponding to the speed of rotation), see Equation (6)):

$$\Phi = \bar{\Omega}\,(\boldsymbol{y}) \rightarrow max \qquad (19)$$

and its value for particular $\boldsymbol{y}_p$ can be found only through simulation (this simulation-based optimization process is as follows: (1) assume a shape (i.e. assume $\boldsymbol{y}$) $\rightarrow$ (2) find the interpolating spline $\rightarrow$ (3) perform simulation (i.e. find the elastic carrying capacity $\bar{\Omega}$ corresponding to the assumed shape) $\rightarrow$ if not StopCondition go to (1)).

3 EVOLUTIONARY AGENT-BASED COMPUTING

In this section an idea of EMAS as an efficient computing system, especially suited for solving problems with costly evaluation of fitness function is presented.

In the last decades intelligent and autonomous software agents have been widely applied in various computing systems, such as power systems management [37], flood forecasting [26], business process management [29], intersection management [18], or solving difficult optimisation problems [35], just to mention a few. The key to understand the concept of a multi-agent system (MAS) is intelligent interaction (like coordination, cooperation, or negotiation).

Agents play an important role in the integration of artificial intelligence subdisciplines, which is often related to a hybrid design of modern intelligent systems [45, 8]. In most similar applications reported in the literature (see e.g. [46, 14] for a review), 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. In other approaches, agents constitute a management infrastructure for a distributed realisation of an evolutionary algorithm [50].

In 1996, Krzysztof Cetnarowicz [13] proposed an idea of an evolutionary multi-agent system (EMAS), which is an agent-oriented computing metaheuristic with interesting features like distributed selection and lack of global control. Since then the idea of EMAS has been applied to different problems (e.g. single, multimodal and multicriteria optimisation). EMAS turned out to be a very good base for introducing different extensions (e.g. using memetic or immunological mechanisms), and encouraged to conduct research at different levels (e.g. formal modelling [11, 48], framework development [23], experimental research [3, 43, 10]).

Agents in EMAS represent solutions to a given optimisation problem. They are located on islands representing a distributed structure of the computation. The islands constitute local environments, where direct interactions among agents may take place. In addition, agents are able to change their location, which makes it possible to exchange information and resources all over the system [31].

In EMAS, phenomena of inheritance and selection – the main components of evolutionary processes – are modelled via agent actions of *death* and *reproduction* (see Figure 2). As in the case of classical evolutionary algorithms, inheritance is accomplished by an appropriate definition of reproduction. Core properties of agent are encoded in its genotype and inherited from its parent(s) with the use of variation operators (mutation and recombination). Moreover, an agent may possess some knowledge acquired during its life, which is not inherited. Both inherited and acquired information (phenotype) determines the behaviour of an agent. It is noteworthy that it is easy to add mechanisms of diversity enhancement, such as allotropic speciation (cf. [12]) to EMAS. It consists in introducing population decomposition and a migration action for agents (see Figure 2).

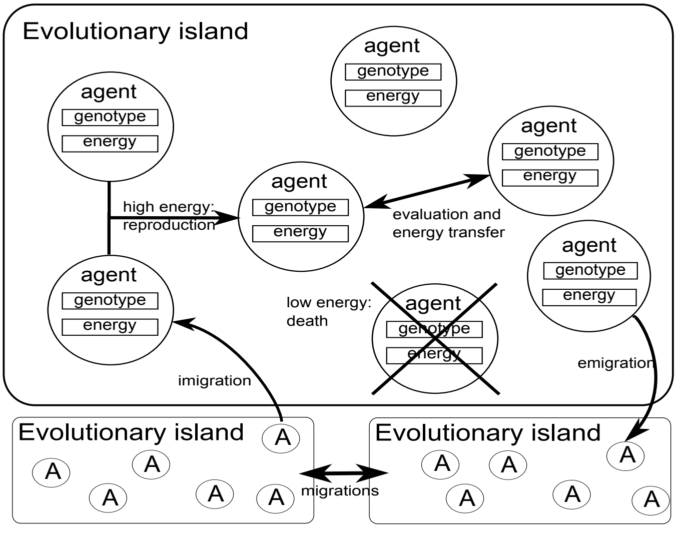


Figure 2. Evolutionary multi-agent system (EMAS)

Assuming that no global knowledge is available, and the agents are autonomous, selection mechanism based on acquiring and exchanging non-renewable resources is introduced [13]. It means that a decisive factor of the agent's fitness is still the quality of solution it represents, but expressed by the amount of non-renewable resource it possesses. In general, the agent gains resources as a reward for "good" behaviour, and looses resources as a consequence of "bad" behaviour (behaviour here may be understood as e.g. acquiring sufficiently good solution). Selection is then realised in such a way that agents with a lot of resources are more likely to reproduce, while a low level of resources increases the possibility of death. So according to classical Franklin's and Graesser's taxonomy – agents of EMAS can be classified as Artificial Life Agents (a kind of Computational Agents) [24].

Agent-based evolutionary computing may be further enhanced using memetic algorithms. Memetic algorithms belong to a class of cultural algorithms and historically are evolutionary algorithms enhanced by hybridisation with local-search methods. The first successful approach was made by Pablo Moscato [40], who hybridised the evolutionary search with a local improvement, using simulated annealing to solve Traveling Salesman Problem. The evolutionary algorithm utilises the local-search method (in the simplest case, the greedy local search or more sophisticated local search techniques, such as simulated annealing or tabu search) within its evolutionary cycle. This might happen in the course of evaluation (according to so called Baldwin effect [28]) or mutation (like in the Lamarckian model of evolution).

In the Lamarck's theory the characteristics of individuals acquired in the course of life may be inherited by their descendants [22]. This method is usually implemented as a local search procedure called in the course of execution of mutation or crossover operator. The search for a mutated individual is based not only on a stochastic one-time sampling from the solution space, it may be a much more complex process, being an outcome of a local search starting from this individual. In the same way the memetic crossover may be implemented, by trying different combinations of parents' genotypes, until a satisfactory match is found. In Lamarckian evolution, individuals improve during their lifetime through local search and the improvement is passed to the next generation. The individuals are selected based on improved fitness and are transferred to the next generation with the improvement incorporated in the genotype.

It is noteworthy, that the ability to solve optimization problems was also formally proven for EMAS (expressed as ergodicity of appropriately constructed Markov chain [4], similarly to the works of Vose [51]) [4, 5, 47], making EMAS a fully-fledged global optimization technique.

EMAS-like systems were implemented many times, using different programming languages and environments such as JAVA, .NET, Scala, Python or Erlang (see e.g. [30, 9]). One of the most mature environments was constructed using JAVA and named AgE. In this environment the agents are placed in tree-like structures (following the composite design pattern) making possible easy decomposition of the populations. The behaviour of the environment is designed according to discrete event simulation [44]. Thus each agent is equipped with a "step" function that is called by the environment whenever it is particular agent's turn (according to simple round-robin strategy). During its step, the agent based on its energy and selected other parameters of the environment, may undertake actions (such as reproduction, migration, meeting and evaluation or local search). The environment was constructed keeping in mind such features as scalability, flexibility and efficiently deliver necessary services such as distributed communication (cf. [23], monitoring [32]), and easy reconfiguration [42]. It is to note that the AgE environment is much more general and not only EMAS can be implemented using this framework (actually AgE was applied to different computing and simulation tasks during the over last 10 years).

The environment is available as open-source project. Moreover, many environments followed the AgE philosophy, however utilising different technologies and languages (e.g. Python, Erlang, Scala, see e.g. [30, 9]). General description of the AgE environment may be found at `http://age.agh.edu.pl`. For Java-based platform and source code refer to: `http://age.iisg.agh.edu.pl`. For more information about AgE-inspired Python-based PyAge environment used to conduct the experiments presented in this paper, refer to: `http://github.com/maciek123/pyage`.

4 EXPERIMENTAL RESULTS

In order to examine the constructed model and explore the optimization capabilities of the claimed-to-be efficient EMAS, an appropriately defined set of experiments was carried out. As the main competitor of EMAS, PEA (Michalewicz version [38]) was used. Both systems were configured in a similar way (the same variation operators, similar selection parameters etc). Thus one of goals of the conducted research was to check the influence of agency on the solving efficacy and efficiency for this particular engineering problem. This section presents and discusses the obtained results.

4.1 Experimental Setup

Simulations were performed with the use of PyAge computing and simulation platform written in Python [30]. Using this software environment, both EMAS and PEA were implemented and used. In the main part of the experiments a shape of simulated disc was represented in 10 equidistant points. The configuration is included in Table 1.

Parameter	EMAS	PEA
Disc radius	$x \in [1.0, 2.0]$	
Disc minimal thickness	$h_{min} = 1.0$	
Disc maximal thickness	$H_{max} = 3.0$	
Mutation	Uniform, of one randomly chosen gene	
Crossover	Single point	
Speciation	Allopatric	
Number of evolutionary islands	3, fully connected	
Numbers of individuals on each island	50	
Agent/individual migration probability	0.05	
Initial energy	100	–
Energy transferred from loser to winner	40	–
Agent's death energy level	0	–
Minimal energy required to reproduce	90	–
Minimal energy required to migrate	120	–
Selection	–	tournament (tournament size: 15)

Table 1. Experiments configuration

All experiments were repeated 11 times and common statistical data was computed. Simulations were run until the number of 80 000 evaluation events was reached.

4.2 Computation Time

A very important feature of the considered class of engineering optimization problems is a high computational cost of solution evaluations. In fact all the considered algorithms spend over 99 % of time computing fitness of processed solutions. In a typical run of PEA, all evaluation events lasted 42107.1 seconds, while the whole computation time was 42 164.4 seconds. That is, the fitness computations took about 99.86 % of computation time. In case of EMAS, evaluation events took 42 723.6 seconds, that is 99.69 % of the whole computation which lasted 42 854.3 seconds. All experiments described in this paper were run on Intel Core i5 with CPU of 1.8 GHz with 4 GB of RAM.

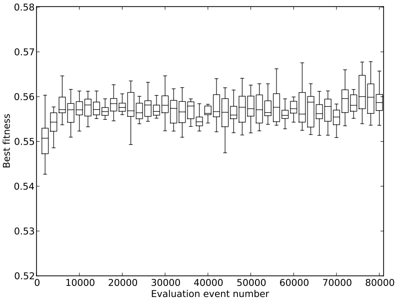
These results show that all mechanisms but evaluation have a negligible effect on efficiency. Therefore using such techniques as EMAS is of significant importance in similarly defined problems as the one tackled in this paper because of its reported lower computation time measured by the number of fitness function calls when compared to other general optimization techniques, e.g. PEA [52, 7]. Therefore the graphs presented in this section are based on evaluation event number as a derivative of time to measure the efficiency of the examined systems.

4.3 Comparison of PEA and EMAS

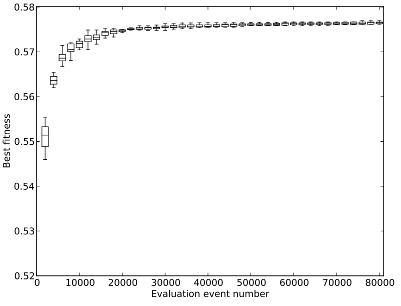
Figure 3 illustrates a comparison of the best fitness (understood as an angular velocity ω) in the successive evaluation events for PEA and EMAS. The values presented in the figure were computed as a mean of all 11 iterations. Moreover, standard deviation was calculated. It turns out that EMAS significantly outperforms PEA. While EMAS continues to improve results, PEA seems to be stuck in some local extremum quite in the initial stage of simulation. Besides, Figure 3 shows that, while in case of PEA the results vary throughout the consecutive iterations (the standard deviation is relatively high), in EMAS the dispersion of results is smaller. In order to provide more exact statistical information, the so-called box and whiskers plots are illustrated in Figures 3 a) and 3 b). These box and whiskers plots present some additional data: the first and the third quartiles, the median, the minimal and the maximal values. The presented results clearly show that the conducted experiments are repeatable.

4.4 PEA and EMAS with Memetics

Apart from classic versions, simulations with Lamarckian local search were performed. Memetic operator was implemented according to the steepest descent algorithm based on choosing the best from 3 mutated individuals. Such a procedure was



a) PEA best fitness

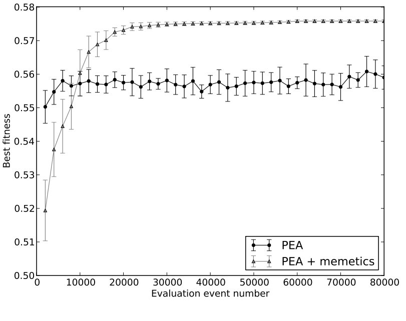


b) EMAS best fitness

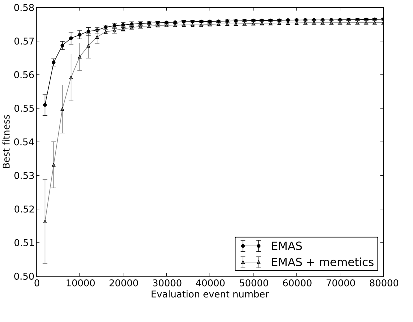
Figure 3. Comparison of PEA and EMAS best fitness in successive evaluation events (box and whiskers)

repeated 3 times and the individual with the best fitness was selected. The exact local search algorithm was based on the isotropic mutation. This strategy generates points uniformly on N-dimensional hypersphere [36].

Figures 4 a) and 4 b) illustrate how the memetics affects the results. It might be observed that PEA was enhanced remarkably. Although the fitness is worse in the initial evaluation events, PEA with memetics outperforms its standard version rapidly. Besides, memetics helps reach the repeatability of the results.



a) PEA and PEA with memetics



b) EMAS and EMAS with memetics

Figure 4. Comparison of PEA and EMAS versions with and without memetics

The final results shown in Figure 4 are quite similar, yet it is noteworthy, that even a small improvement is very significant from the engineering point-of-view, therefore EMAS prevails. Though obtained suboptimal solution (even for PEA) might be acceptable, it is to note that EMAS reached these results quicker that its counterparts (after about 10 000 evaluations of the fitness function, while PEA in the best case – with memetics used – needs at least two times more fitness function evaluations).

Different remarks have to be made in case of EMAS. As one can see, the application of memetics did not influence the result reached at the end of computation at all. The best fitness is neither better nor worse and it remains on the more or less constant level. To sum up, it seems that memetics is not needed in this particular case, to make EMAS better.

4.5 Diversity

Main observations were focused on the best fitness, however the diversity was also calculated according to the following definitions of this measure:

- Morrison-De Jong (MOI) measure based on the concept of moment of inertia [39],
- maximum standard deviation (MSD) of each gene computed for all individuals in the population.

Figures 5 a), 5 b), 5 c) and 5 d) illustrate how MOI and MSD diversity measures changed throughout the simulation process and how these two measures depended on the application of memetics. It might be seen that the application of memetics significantly decreases diversity, both for PEA and EMAS. Though diversity for EMAS is slightly lower than in case of PEA, the obtained results are of the same range or better (see Section 4.7) maintaining the lower computation cost (see Section 4.2).

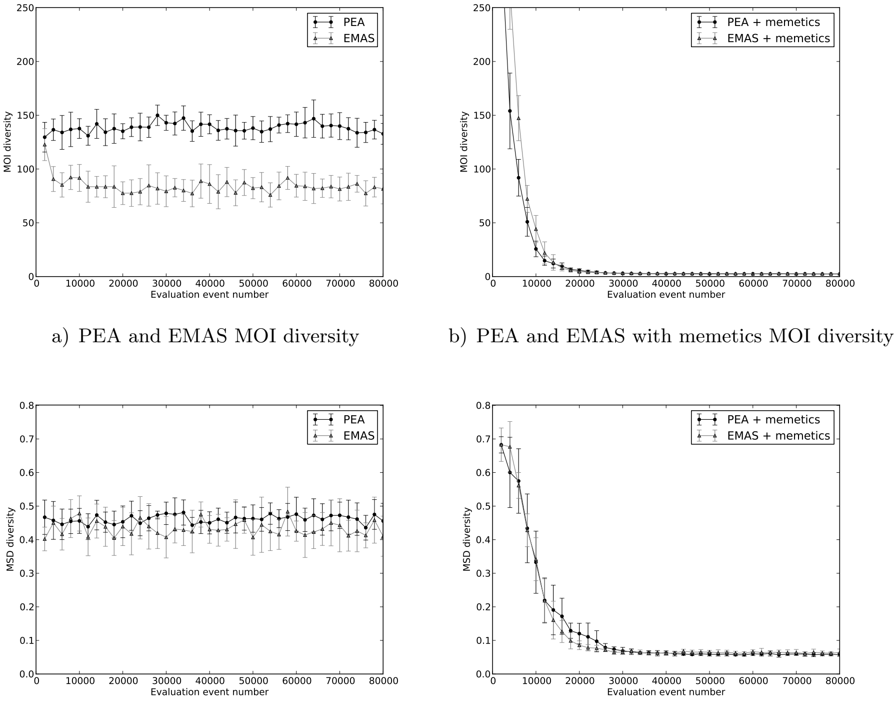
4.6 Experiments with Variable Number of Evolutionary Islands

In order to prove how parallelization affects the obtained results, we performed some preliminary experiments with the use of EMAS and a variable number of evolutionary islands. Thus we were able to check how best fitness changes in dependence on the mean number of evaluation events for each evolutionary island. Table 2 presents the results of the best fitness computed in the 15 000$^{\text{th}}$ evaluation applying 1, 3 and 6 evolutionary islands.

Number of Islands	Best Fitness	Standard Deviation
1	0.5721	0.0015
3	0.5738	0.0008
6	0.5745	0.0008

Table 2. The best fitness computed in (15 000$^{\text{th}}$) evaluation event with variable number of evolutionary islands

Although the differences are not large, it is clear that the increase of the number of evolutionary islands improves the obtained results and guarantees they are more stable (the standard deviation is lower). As these preliminary results look very



a) PEA and EMAS MOI diversity

b) PEA and EMAS with memetics MOI diversity

c) PEA and EMAS MSD diversity

d) PEA and EMAS with memetics MSD diversity

Figure 5. Diversity

promising, it would be beneficial to perform more tests focused on parallelization in our future work.

4.7 Optimal Disc Shape

Last but not least we present the shape of rotating variable-thickness disc obtained as a result of the presented different evolutionary systems. In this Section variation of results throughout the simulation process, in the successive evaluation events, is discussed. Table 3 summarizes the results obtained in the last evaluation event.

Figures 6 a), 6 b), 6 c) and 6 d) present a thickness of the optimal shape found in the simulation process. Points representing thickness values in each of the equidistant nodes have been connected with cubic spline in order to illustrate a disc shape. These curves correspond to the line $y_0, ..., y_n$ illustrated in Figure 1.

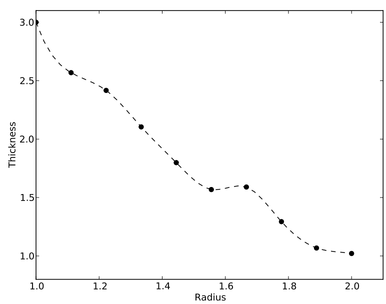
Similar disc shapes found with EMAS, PEA with memetics and EMAS with memetics correspond closely to the best results obtained in the simulations. The

	Best fitness	St. dev.	MSD div.	MSD st. dev.	MOI div.	MOI st. dev.
PEA	0.5590	0.0035	0.4556	0.0526	132.7053	9.6958
EMAS	0.5765	0.0002	0.4104	0.0598	81.6735	14.0741
PEA + memetics	0.5758	0.0004	0.0574	0.0046	2.3703	0.1538
EMAS + memetics	0.5755	0.0005	0.0643	0.0101	2.7002	0.2687

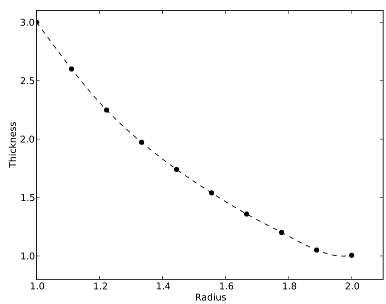
Table 3. Simulation results in the last ($80\,000^{\text{th}}$) evaluation event

worst results were obtained with the use of PEA, therefore the shape presented in Figure 6 a) remarkably differs from the rest.

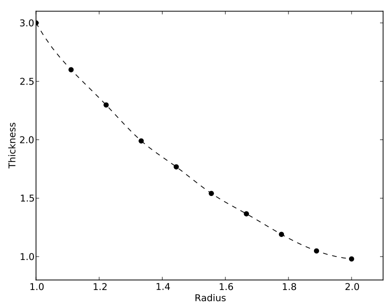
Finally, some tests with more spline nodes were performed. Their number was increased to 20 and simulation with the use of EMAS was rerun in order to examine how this modification influenced on the results. In Figure 7 a) we compared the best fitness reached by the simulation while using 10 and 20 nodes. It might be observed



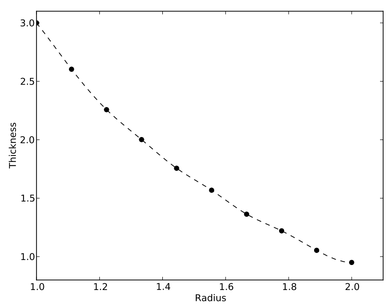
a) Optimal disc shape found with PEA



b) Optimal disc shape found with EMAS



c) Optimal disc shape found with PEA with memetics

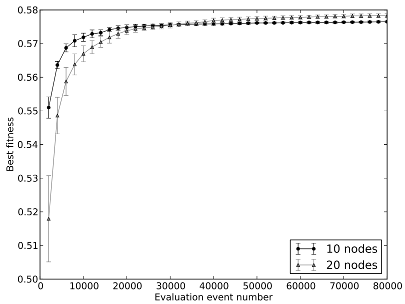


d) Optimal disc shape found with EMAS with memetics

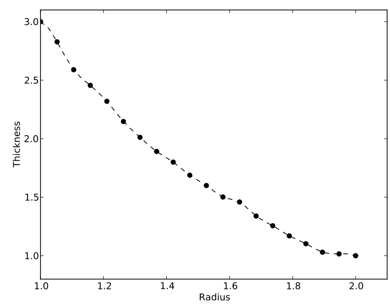
Figure 6. Optimal disc shape

that, initially, fitness improvement took place faster in the former case, nevertheless the final result was better while 20 nodes were applied.

Figure 7 b) illustrates the most optimal disc shape designed as an outcome of the experiments with the use of 20 nodes.



a) EMAS best fitness



b) EMAS optimal disc shape

Figure 7. EMAS best fitness for 10 and 20 nodes and optimal disc shape found with EMAS for 20 nodes

5 CONCLUSION

In this paper a practical engineering problem, namely shape optimization of a variable-thickness rotating annular disc was presented. Being a very complex problem without feasible deterministic solution available, metaheuristic-based approach was chosen, namely classic (as PEA) and agent-based (EMAS) were considered.

EMAS-based solution method was selected because it proved effective in other optimization problems (thanks to small number of fitness evaluation events). It was especially important because in the analyzed problem the evaluation of solution candidates was simulation-based and the time cost of single simulation required the biggest computation effort for the whole computing system. Moreover, EMAS was proven as a general-purpose optimization algorithm (supported by carefully lead formal proof), making this technique a very good approach for solving so-called "black-box" scenarios [21].

The obtained results clearly showed that agent-based techniques prevailed classic evolutionary algorithm both in solution quality (the fitness was just a little better than the one obtained by PEA, but the resulting shape was smoother) and efficiency.

Future research work could concentrate on the application of EMAS-based approach to other engineering problems and on adapting the presented algorithms to highly-parallel, heterogeneous (CPU+GPU), computer systems.

Since experimental studies of the considered class of population-based computational intelligence techniques require much computing power, in the near future we also plan to extensively use the available computing infrastructure, using one of available data-farming frameworks supporting scientific computations, namely Scalarm [33, 34].

Acknowledgments

The research presented in this paper received partial financial support from AGH University of Science and Technology statutory project.

REFERENCES

[1] Berger, M.—Porat, I.: Optimal Design of a Rotating Disk for Kinetic Energy Storage. Journal of Applied Mechanics, Vol. 55, 1998, No. 1, pp. 164–170.

[2] Byrski, A.—Drezewski, R.—Siwik, L.—Kisiel-Dorohinicki, M.: Evolutionary Multi-Agent Systems. The Knowledge Engineering Review, Vol. 30, 2015, No. 2, pp. 171–186, doi: 10.1017/S0269888914000289.

[3] Byrski, A.—Kisiel-Dorohinicki, M.: Agent-Based Evolutionary and Immunological Optimization. In: Shi, Y., van Albada, G. D., Dongarra, J., Sloot, P. M. A. (Eds.): Computational Science – ICCS 2007. Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 4488, 2007, pp. 928–935.

[4] Byrski, A.—Schaefer, R.: Formal Model for Agent-Based Asynchronous Evolutionary Computation. IEEE Congress on Evolutionary Computation (CEC '09), May 2009, pp. 78–85, doi: 10.1109/cec.2009.4982933.

[5] Byrski, A.—Schaefer, R.—Smołka, M.—Cotta, C.: Asymptotic Guarantee of Success for Multi-Agent Memetic Systems. Bulletin of the Polish Academy of Sciences, Technical Sciences, Vol. 61, 2013, No. 1, 257–278, doi: 10.2478/bpasts-2013-0025.

[6] Byrski, A.: Agent-Based Metaheuristics in Search and Optimisation. AGH University of Science and Technology Press, 2013.

[7] Byrski, A.: Tuning of Agent-Based Computing. Computer Science, Vol. 14, 2013, No. 3, p. 491–512.

[8] Byrski, A.—Carvalho, M.: Agent-Based Immunological Intrusion Detection System for Mobile Ad-Hoc Networks. In: Bubak, M., van Albada, G. D., Dongarra, J., Sloot, P. M. A. (Eds.): Computational Science – ICCS 2008. Springer Berlin Heidelberg, Lecture Notes in Computer Science, Vol. 5103, 2008, pp. 584–593.

[9] Byrski, A.—Kisiel-Dorohinicki, M.: Agent-Based Model and Computing Environment Facilitating the Development of Distributed Computational Intelligence Systems. In: Allen, G., Nabrzyski, J., Seidel, E., van Albada, G. D., Dongarra, J., Sloot, P. M. A. (Eds.): Computational Science – ICCS 2009. Springer Berlin Heidelberg, Lecture Notes in Computer Science, Vol. 5545, 2009, pp. 865–874.

[10] Byrski, A.—Korczynski, W.—Kisiel-Dorohinicki, M.: Memetic Multi-Agent Computing in Difficult Continuous Optimisation. In: Barbucha, D., Le, M. T., Howlett, R. J., Jain, L. C. (Eds.): KES-AMSTA. IOS Press, Frontiers in Artificial Intelligence and Applications, Vol. 252, 2013, pp. 181–190.

[11] Byrski, A.—Schaefer, R.: Stochastic Model of Evolutionary and Immunological Multi-Agent Systems: Mutually Exclusive Actions. Fundamenta Informaticae, Vol. 95, 2009, No. 2–3, pp. 263–285.

[12] Cantú-Paz, E.: A Summary of Research on Parallel Genetic Algorithms. IlliGAL Report No. 95007. University of Illinois, 1995.

[13] Cetnarowicz, K.—Kisiel-Dorohinicki, M.—Nawarecki, E.: The Application of Evolution Process in Multi-Agent World (MAW) to the Prediction System. In: Tokoro, M. (Ed.): Proceedings of the 2nd International Conference on Multi-Agent Systems (ICMAS '96), AAAI Press, 1996, pp. 26–32.

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

[15] Debski, R.—Drezewski, R.—Kisiel-Dorohinicki, M.: Maintaining Population Diversity in Evolution Strategy for Engineering Problems. In: Nguyen, N. T., Borzemski, L., Grzech, A., Ali, M. (Eds.): New Frontiers in Applied Artificial Intelligence (IEA/AIE 2008). Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 5027, 2008, pp. 379–387.

[16] Debski, R.—Woźniak, G.—Drezewski, R.—Kisiel-Dorohinicki, M.: Evolutionary Strategy for Shape Design of Rotating Variable-Thickness Annular Elastic Disc. AI-METH, 2002, pp. 135–138.

[17] Debski, R.—Zyczkowski, M.: On Decohesive Carrying Capacity of Variable-Thickness Annular Perfectly Plastic Disks. ZAMM-Journal of Applied Mathematics and Mechanics, Vol. 82, 2002, pp. 655–669, doi: 10.1002/1521-4001(200210)82:10<655::AID-ZAMM655>3.0.CO;2-V.

[18] Dresner, K.—Stone, P.: A Multiagent Approach to Autonomous Intersection Management. Journal of Artificial Intelligence Research, Vol. 31, 2008, pp. 591–656.

[19] Drezewski, R.—Siwik, L.: Multi-Objective Optimization Technique Based on Co-Evolutionary Interactions in Multi-Agent System. In: Giacobini, M. et al. (Eds.): Applications of Evolutionary Computing (EvoWorkshops 2007). Springer-Verlag, Berlin, Heidelberg, LNCS, Vol. 4448, 2007, pp. 179–188.

[20] Drezewski, R.—Siwik, L.: Co-Evolutionary Multi-Agent System for Portfolio Optimization. In: Brabazon, A., O'Neill, M. (Eds.): Natural Computing in Computational Finance. Springer-Verlag, Berlin, Heidelberg, Studies in Computational Intelligence, Vol. 100, 2008, pp. 271–299.

[21] Droste, S.—Jansen, T.—Wegener, I.: Upper and Lower Bounds for Randomized Search Heuristics in Black-Box Optimization. Theory of Computing Systems, Vol. 39, 2006, pp. 25–544, doi: 10.1007/s00224-004-1177-z.

[22] Eldridge, N.—Gould, S. J.: Punctuated Equilibria: An Alternative to Phyletic Gradualism. In: Schopf, T. J. M. (Ed.): Models in Paleobiology. Freeman, Cooper and Co., 1972.

[23] Faber, Ł.—Piętak, K.—Byrski, A.—Kisiel-Dorohinicki, M.: Agent-Based Simulation in Age Framework. In: Byrski, A., Oplatková, Z., Carvalho, M., Kisiel-Dorohinicki, M. (Eds.): Advances in Intelligent Modelling and Simulation. Springer Berlin Heidelberg, Studies in Computational Intelligence, Vol. 416, 2012, pp. 55–83.

[24] Franklin, S.—Graesser, A.: Is It an Agent, or Just a Program? A Taxonomy for Autonomous Agents. Proceedings of the Workshop on Intelligent Agents III, Agent Theories, Architectures, and Languages (ECAI '96). Springer-Verlag, London, UK, 1997, pp. 21–35, doi: 10.1007/bfb0013570.

[25] Genta, G.—Bassani, D.: Use of Genetic Algorithms for the Design of Rotors. Meccanica, Vol. 30, 1995, No. 6, pp. 707–717.

[26] George, J. P.—Gleizes, M. P.—Glize, P.—Regis, C.: Real-Time Simulation for Flood Forecast: An Adaptive Multi-Agent System Staff. Proceedings of the AISB '03 Symposium on Adaptive Agents and Multi-Agent Systems, University of Wales, 2003.

[27] Goldberg, D. E.: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1st edition, 1989.

[28] Hinton, G.—Nolan, S.: How Learning Can Guide Evolution. Complex Systems, Vol. 1, 1987, pp. 495–502.

[29] Jennings, N. R.—Faratin, P.—Johnson, M. J.—Norman, T. J.—OBrien, P.—Wiegand, M. E.: Agent-Based Business Process Management. International Journal of Cooperative Information Systems, Vol. 5, 1996, No. 2–3, pp. 105–130, doi: 10.1142/s0218843096000051.

[30] Kaziród, M.—Korczynski, W.—Byrski, A.: Agent-Oriented Computing Platform in Python. Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2014 IEEE/WIC/ACM International Joint Conferences, IEEE, Vol. 3, 2014, pp. 365–372.

[31] Kisiel-Dorohinicki, M.: Agent-Oriented Model of Simulated Evolution. In: Grosky, W. I., Plasil, F. (Eds.): SofSem 2002: Theory and Practice of Informatics. Springer, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 2540, 2002, pp. 253–261, doi: 10.1007/3-540-36137-5_19.

[32] Kosinski, J.—Nawrocki, P.—Radziszowski, D.—Zielinski, K.—Zielinski, S.—Przybylski, G.—Wnek, P.: SLA Monitoring and Management Framework for Telecommunication Services. Fourth International Conference on Networking and Services (ICNS 2008), March 2008, pp. 170–175, doi: 10.1109/icns.2008.31.

[33] Krol, D.—Slota, R.—Kitowski, J.—Dutka, Ł.—Liput, J.: Data Farming on Heterogeneous Clouds. In: Kesselman, C. (Ed.): Cloud 2014: 2014 IEEE Seventh International Conference on Cloud Computing, June 27–July 2, 2014, Anchorage, Alaska, Proceedings. IEEE, 2014, pp. 873–880, doi: 10.1109/CLOUD.2014.120.

[34] Krol, D.—Wrzeszcz, M.—Kryza, B.—Dutka, Ł.—Kitowski, J.: Massively Scalable Platform for Data Farming Supporting Heterogeneous Infrastructure. Proceedings of Cloud Computing 2013, The Fourth International Conference on Cloud Computing, GRIDs, and Virtualization, May 27–June 1, 2013, Valencia, Spain. IARIA, 2013, pp. 144–149.

[35] Lobel, B. I.—Ozdaglar, A.—Feijer, D.: Distributed Multi-Agent Optimization with State-Dependent Communication. Mathematical Programming, Vol. 129, 2011, No. 2, pp. 255–284.

[36] Luban, M.—Staunton, L. P.: An Efficient Method for Generating a Uniform Distribution of Points Within a Hypersphere. Computers in Physics, Vol. 2, 1988, pp. 55–60, doi: 10.1063/1.168311.

[37] McArthur, S. D. J.—Catterson, V. M.—Hatziargyriou, N. D.: Multi-Agent Systems for Power Engineering Applications, Part I: Concepts, Approaches, and Technical Challenges. IEEE Transactions on Power Systems, Vol. 22, 2007, No. 4.

[38] Michalewicz, Z.: Genetic Algorithms Plus Data Structures Equals Evolution Programs. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1994, doi: 10.1007/978-3-662-07418-3.

[39] Morrison, R. W.—De Jong, K. A.: Measurement of Population Diversity. In: Collet, P., Fonlupt, C., Hao, J.-K., Lutton, E., Schoenauer, M. (Eds.): Artificial Evolution. Springer, Lecture Notes in Computer Science, Vol. 2310, 2002, pp. 31–41.

[40] Moscato, P.: Memetic Algorithms: A Short Introduction. New Ideas in Optimization. McGraw-Hill Ltd., UK, 1999, pp. 219–234.

[41] Penninger, C.—Swift, R.: Disc Brake Lining Shape Optimization by Multibody Dynamic Analysis. SAE International, 2004, doi: 10.4271/2004-01-0725.

[42] Pietak, K.—Woś, A.—Byrski, A.—Kisiel-Dorohinicki, M.: Functional Integrity of Multi-Agent Computational System Supported by Component-Based Implementation. Proceedings of the 4th International Conference on Industrial Applications of Holonic and Multi-Agent Systems, 2009, doi: 10.1007/978-3-642-03668-2_8.

[43] Pisarski, S.—Rugała, A.—Byrski, A.—Kisiel-Dorohinicki, M.: Evolutionary Multi-Agent System in Hard Benchmark Continuous Optimisation. Proceedings of the 16th European Conference on Applications of Evolutionary Computation (EvoApplications '13). Springer-Verlag, Berlin, Heidelberg, Lecture Notes in Computer Science, Vol. 7835, 2013, pp. 132–141, doi: 10.1007/978-3-642-37192-9_14.

[44] Robinson, S.: Simulation – The Practice of Model Development and Use. Wiley, 2004.

[45] Russel, S. J.—Norvig, P.: Artificial Intelligence: A Modern Approach. Prentice Hall, 2003.

[46] Sarker, R.—Ray, T.: Agent-Based Evolutionary Search. Vol. 5 of Adaptation, Learning and Optimization, Springer, 1st edition, 2010, doi: 10.1007/978-3-642-13425-8.

[47] Schaefer, R.—Byrski, A.—Kołodziej, J.—Smołka, M.: An Agent-Based Model of Hierarchic Genetic Search. Computers and Mathematics with Applications, Vol. 64, 2012, No. 12, pp. 3763–3776, doi: 10.1016/j.camwa.2012.02.052.

[48] Schaefer, R.—Byrski, A.—Smołka, M.: Stochastic Model of Evolutionary and Immunological Multi-Agent Systems: Parallel Execution of Local Actions. Fundamenta Informaticae, Vol. 95, 2009, No. 2-3, pp. 325–348.

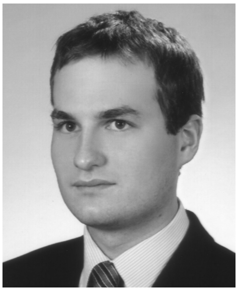
[49] Schwefel, H.-P.: Numerical Optimization of Computer Models. John Wiley & Sons, Inc., New York, NY, USA, 1981.

[50] Uhruski, P.—Grochowski, M.—Schaefer, R.: Multi-Agent Computing System in a Heterogeneous Network. Proceedings of the International Conference on Parallel Computing in Electrical Engineering (PARELEC 2002), Warsaw, Poland,

September 22–25, 2002. IEEE Computer Society Press, 2002, pp. 233–238, doi: 10.1109/pcee.2002.1115252.

[51] Vose, M.: The Simple Genetic Algorithm: Foundations and Theory. MIT Press, Cambridge, MA, USA, 1998.

[52] Wróbel, K.—Torba, P.—Paszyński, M.—Byrski, A.: Evolutionary Multi-Agent Computing in Inverse Problems. Computer Science, AGH University of Science and Technology Press, Vol. 14, 2013, No. 3.



Wojciech Korczyński received his M.Sc. in 2013 and is currently a Ph.D. student at AGH University of Science and Technology. His main scientific interests are multi-agent systems and natural language processing.

Aleksander Byrski received his Ph.D. in 2007 and D.Sc. (habilitation) in 2013 at AGH University of Science and Technology in Cracow. He works as Assistant Professor at the Department of Computer Science of AGH-UST. His research focuses on multi-agent systems, biologically-inspired computing and other soft computing methods.

Roman Dębski works as Assistant Professor at the Department of Computer Science at AGH University of Science and Technology. He received his M.Sc. in mechanics (1997) and computer science (2002) and Ph.D. specializing in computational mechanics (2002). Before joining the university he worked in the IT industry for over 15 years. His current interests include mathematical modeling and computer simulations, parallel processing, heterogeneous computing and trajectory optimization.

Marek Kisiel-Dorohinicki received his Ph.D. in 2001 and D.Sc. (habilitation) in 2013 at AGH University of Science and Technology in Cracow. He works as Assistant Professor at the Department of Computer Science of AGH-UST. His research focuses on intelligent software systems, particularly utilizing agent technology and evolutionary algorithms, but also other soft computing techniques.