Global optimization via evolutionary search with soft selection

Andrzej Obuchowicz, Józef Korbicz
Institute of Robotics and Software Engineering
Technical University of Zielona Góra
Podgórna 50
65-246 Zielona Góra, Poland
{A.Obuchowicz,J.Korbicz}@irio.pz.zgora.pl

Keywords: Evolutionary algorithms, global optimization, soft selection.

Abstract

The aim of this work is to draw a comparison of four variants of the Evolutionary Search with Soft Selection (ESSS) algorithms based on selected parameter optimization problems. They are tested with nine objective functions, most of them being strongly non-linear and multimodal. From the results obtained it follows that all modified ESSS algorithms are more effective and generally faster than the basic ESSS algorithm.

1 Introduction

The applicability of evolutionary inspirations in global optimization is not questionable. The main advantage of evolutionary processes is their capability of saddle crossing on multimodal surfaces. In contrast to conventional optimization methods, an evolutionary search does not get stuck (potentially) in a local optimum trap, which is a crucial characteristic of global optimization.

It is easy to prove that the Darwinian-type evolution has a cyclic nature in multimodal adaptation landscapes [4]. Each cycle consists of two phases: active and latent. In relative short active phases the population of individuals climbs on an adaptation slope to the neighbourhood of a local peak. The latent phase is a quasi-stationary state with sporadic fluctuations. If an occupied hill possesses a higher neighbour, then fluctuations can contribute to the crossing of the saddle and a new active phase starts.

In the case of global optimization problems without constrains a searching procedure has to reconcile the optimum localization with the capability of saddle crossing. Application of two specialized algorithms is a good solution to this problem. However, although there are many specialized algorithms of local optimization, the algorithms specialized in saddle crossing are scarce. Pure Darwinian evolution is, of course, a compromise method, but its aim is adaptation, not optimization. Consequently, its saddle crossing mechanisms are more interesting.

The idea of natural evolution is applied in several well-known algorithms, e.g. Evolutionary Strategies [15], Evolutionary Programming [2], Genetic Algorithms [6], and some unfamiliar Evolutionary Search with Soft Selection (ESSS) [3]. The last one is considered in this paper.

The ESSS algorithm is based on probably the simplest selection-mutation model of Darwinian evolution. The searching process is executed in a multi-dimensional real space, on which some non-negative function, called *fitness*, is defined. At the beginning the population of points is randomly chosen from the searching space and is iteratively changed by selection and mutation operators. As the selection operator the well-known proportional selection is used. The entries of selected elements are mutated by adding normally-distributed random values.

A long-life time of the latent phase results from the fact that the selection process prefers new offsprings allocated in well-exploited areas around the occupied peak. This is, of course, a disadvantage in the contest of the effectiveness of the optimization process. In order to overcome this problem, a natural idea is to exclude the neighborhood of the occupied peak in the exploration process. There are many instances of this idea in specialized literature [5, 9].

The aim of this paper is to introduce and compare three biologically-inspired techniques which accelerate the saddle-crossing ability of the ESSS algorithm. These are:

1. ESSS with Simple Variance Adaptation (ESSS-SVA) [13],
2. ESSS with Deterioration of the Objective Function (ESSS-DOF) [10], and
3. ESSS with Forced Direction of Mutation (ESSS-FDM) [11].

All above modifications are based on the idea of evolutionary trap, which is observed in nature. The population which is trapped around a local quality peak (*an ecological niche*) and whose quality growth is impossible, tries to explore the environment in two possible ways. The first one is to increase the phenotype variety in the population. This mechanism is proposed in ESSS-SVA and ESSS-FDM algorithms. The other is an erosion of the actual quality peak (a

Table 1. The outline of the ESSS algorithm.

Input data

η – population size;

$t_{\max}$ – maximum number of iterations (epochs);

σ – standard deviation of the normal distribution;

$\Phi : \mathbb{R}^n \to \mathbb{R}_+$ – non-negative fitness function (adaptation landscape), n number of features;

$\mathbf{x}_0^0$ – initial point.

1. *Initiation*

(a) $P(0) = \{\mathbf{x}_1^0, \mathbf{x}_2^0, \ldots, \mathbf{x}_\eta^0\} : \quad (\mathbf{x}_k^0)_i = (\mathbf{x}_0^0)_i + N(0,\sigma);$ (1)
$i = 1, 2, \ldots, n; \quad k = 1, 2, \ldots, \eta.$

(b) $q_0^0 = \Phi(\mathbf{x}_0^0)$ (2)

2. Repeat:

(a) *Estimation*
$P(t) \to \Phi(P(t)) = \{q_1^t, q_2^t, \ldots, q_\eta^t\} : \quad q_k^N = \Phi(\mathbf{x}_k^t), \quad k = 1, 2, \ldots, \eta.$ (3)

(b) *Choice of the best element in the history*
$\{\mathbf{x}_0^t, \mathbf{x}_1^t, \mathbf{x}_2^t, \ldots, \mathbf{x}_\eta^t\} \to \mathbf{x}_0^{t+1} : \quad q_0^{t+1} = \max\{q_k^t\}, \quad k = 0, 1, \ldots, \eta.$ (4)

(c) *Selection*
$$\Phi(P(t)) \to \{h_1, h_2, \ldots, h_\eta\} : \quad h_k = \min\left\{ h : \frac{\sum_{l=1}^{h} q_l^t}{\sum_{l=1}^{\eta} q_l^t} > \zeta_k \right\}, \quad (5)$$
where $\{\zeta_k\}_{k=1}^{\eta}$ are random variables uniformly distributed on the interval $[0, 1)$.

(d) *Mutation*
$P(t) \to P(t+1);$
$$(\mathbf{x}_k^{t+1})_i = (\mathbf{x}_{h_k}^t)_i + N(0,\sigma), \quad i = 1, 2, \ldots, n; \quad k = 1, 2, \ldots, \eta, \quad (6)$$
where $N(m,\sigma)$ is a normally-distributed random variable with expectation m and standard deviation σ.

Until $t > t_{\max}$.

deterioration of the ecological niche), which is implemented in the ESSS-DOF algorithm.

2 ESSS algorithm

A basic phenotype evolution model was proposed by Galar [3]. Foundations of this algorithm are as follows:

- There exists an environment of invariant properties which have a limited capacity.

- There exists a population of reproducing elements (individuals of the same species). The elements of the population are characterized by a set of features (phenotype quantitative features). The set of all possible values of the features determines the type of each element (phenotype). Each type is assigned its fitness.

- The assumption that each element occupies one place in environment is made. The elements "live" in the environment for some length of time (generation), and then a new generation is produced out of the actual one (reproduction).

- The new generation is created by selecting parent elements from the actual generation and changing their features (asexual reproduction).

- The choice of the parents is accomplished by soft selection which is random process. Each parent element has a chance of allocating a descendant in the environment with probability proportional to the element quality.

- The descendant elements are not perfect copies of the parent elements. Type differences result from clear random mutation.

The relevant assumptions can be formalized by the numerical ESSS algorithm (Table 1). Some results of the investigations regarding the efficiency of the ESSS algorithm are presented in [4]. Numerical tests proved essential advantages of soft selection over selection which reaches only a local optimum. The ESSS algorithm is not an optimization algorithm in the sense of reaching an optimum with a desired accuracy. Evolution is not asymptotically convergent to an optimum and the interpolation efficiency of the soft selection is weak. Evolution leads next generations to an elevated response surface, rather than to maxima. Evolution which started from a high and narrow peak could terminate on a lower and wider peak. In spite of that, search advantages of the ESSS algorithm suggest that this algorithm can be used in numerical packages for global numerical optimization, especially when combined with local optimization algorithms.

3 Elements of convergence analysis

The convergence analysis of evolutionary algorithms is very difficult or, sometimes, impossible owing to their nonlinear, stochastic or heuristic nature. Standard tools of dynamic systems analysis are not effective, even though simple models are used. A first attempt at the ESSS convergence analysis was presented in [7, 8]. For simplicity let us assume an infinite size of the population. The population state can be represented by a distribution function $p_t(\mathbf{x})$ defined on the phenotype space at a discrete time moment t [7]. Let $\Phi(\mathbf{x})$ be a nonnegative fitness function and $g(\mathbf{x}-\mathbf{y})$ be a modification function which describes the transformation of an element $\mathbf{y}$ into $\mathbf{x}$:

$$g(\mathbf{x}-\mathbf{y}) = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^n \exp\left(-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}\right). \quad (7)$$

The population distribution after selection (5) can be calculated from the formula

$$p_t'(\mathbf{x}) = \frac{\Phi(\mathbf{x})p_t(\mathbf{x})}{\int\cdots\int_{\mathbb{R}^n}\Phi(\mathbf{z})p_t(\mathbf{z})d\mathbf{z}} = \frac{\Phi(\mathbf{x})p_t(\mathbf{x})}{\langle\Phi(\mathbf{z})\rangle} \quad (8)$$

and after mutation by

$$p_{t+1}(\mathbf{x}) = \int\cdots\int_{\mathbb{R}^n} p_t'(\mathbf{y})g(\mathbf{x}-\mathbf{y})d\mathbf{y} \quad (9)$$

Equations (8) and (9) do not possess, in general, closed-form solutions. Let us consider the fitness function in the Gaussian peak form with a maximum localized in the centre of the reference frame:

$$\Phi(\mathbf{x}) = \exp\left(-\frac{1}{2}\mathbf{x}^T\mathbb{T}^{-1}\mathbf{x}\right), \quad (10)$$

where $\mathbb{T}$ defines the ellipsoidal contour line of the peak. Similarly, let us assume that the population distribution at t is of the normal form

$$p_t(\mathbf{x}) = \left(\frac{1}{\sqrt{2\pi\det\mathbb{C}_t}}\right)^n \times \exp\left(-\frac{1}{2}(\mathbf{x}-\langle\mathbf{x}^t\rangle)^T\mathbb{C}_t^{-1}(\mathbf{x}-\langle\mathbf{x}^t\rangle)\right), \quad (11)$$

where $\mathbb{C}_t$ is the population covariance matrix and $\langle\mathbf{x}^t\rangle$ is the population expectation vector at t. If the initial population has isotropic symmetry, e.g. $P(0)$ is obtained by normal mutations of a given phenotype $\langle\mathbf{x}^0\rangle$, then we may expect that, after some epochs, the matrices $\mathbb{C}$ in (11) and $\mathbb{T}$ (10) have the same set of eigenvectors. Hence, after a similarity transformation, equations (10) and (11) have the forms

$$\Phi(\mathbf{x}) = \prod_{i=1}^n \exp\left(-\frac{x_i^2}{2\tau_i^2}\right) \quad (12)$$

and

$$p_t(\mathbf{x}) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi}\nu_{ti}} \exp\left(-\frac{\left(x_i-\langle x_i^t\rangle\right)^2}{2\nu_{ti}^2}\right) \quad (13)$$

If we use the above formulae in (8) and (9), then the population distribution at time $t+1$ is normal with the expectation vector

$$\langle x_i^{t+1}\rangle = \langle x_i^t\rangle\frac{\tau_i^2}{\nu_{ti}^2+\tau_i^2}, \qquad i=1,2,\ldots,n \quad (14)$$

and the variance vector

$$\nu_{t+1,i} = \sqrt{\sigma^2+\frac{\tau_i^2\nu_{ti}^2}{\nu_{ti}^2+\tau_i^2}}, \qquad i=1,2,\ldots,n. \quad (15)$$

It is easy to see that $\lim_{t\to\infty}\|\langle\mathbf{x}^t\rangle\| = 0$, so the ESSS algorithm, in the case of an infinite population size, is convergent to the optimum of the fitness function. In the latent phase, the system is stable $p_{t+1}(\mathbf{x}) = p_t(\mathbf{x})$ and has normal distribution form with the zero mean and the variance vector:

$$\nu_{\infty,i} = \sigma\sqrt{\frac{1}{2}\left(1+\sqrt{1+\left(\frac{2\tau_i}{\sigma}\right)}\right)}, \qquad i=1,2,\ldots,n, \quad (16)$$

Analyzing (16) one can deduce the following characteristics:

- If $\tau_i \ll \sigma$, then the variance of the population distribution can be approximated by the variance of the modification function $\nu_{\infty,i} \approx \sigma$. This is the proof that the optimum point is an attractor which does not allow the population to disperse.

- If $\tau_i \approx \sigma$, then $\nu_{\infty,i}^2 \approx \sigma^2(1+\sqrt{5})/2 \approx 1.618\sigma^2$, so there are no quality differences between this case and that described above.

- If $\tau_i \gg \sigma$, then the variance of the population distribution can be approximated by the geometric mean of modification and fitness functions variances $\nu_{\infty,i} \approx \sqrt{\tau_i \sigma}$.

Small changes in the population variance account for the observed fact that the population is concentrated during the searching process unless the scoured area is a plateau.

4 Family of modified ESSS algorithms

4.1 ESSS-SVA

The idea is the following. When the population is trapped around a local peak then the standard deviation of mutation increases. This fact manifests itself in a larger variance of population and a worse mean fitness of population. In this way the mean fitness of population decreases to a saddle level and the possibility of saddle crossing increases.

In comparison with the ESSS, the ESSS-SVA algorithm is enriched by a additional mechanism, which consists of three new procedures:

1. *Trap test* – the objective of this procedure is to determine whether the population quality changed substantially for a given number of epochs t_t. The output is positive if the population displacement for the last t_t epochs is of the same order as the mutation variance σ_t
2. *Adaptation of the mutation variance* – This procedure is fired if an evolutionary trap is detected. The variance of the normal distribution used in mutation is multiplied by a constant $\alpha > 1$.
3. *Coming back to the initial variance* – It the evolutionary trap is not detected then the variance of normal distribution is equal to an initial, relatively low value.

The ESSS-SVA algorithm can be written in the following form (see Table 1):

1. *Initiation*
2. Repeat
   (a) *Estimation*;
   (b) *Choice of the best element in the history*;
   (c) If *Trap Test* then *Adaptation of the mutation variance* else *Coming back to the initial variance*;
   (d) *Selection*;
   (e) *Mutation*;

   Until $t > t_{\max}$.

The ESSS-SVA algorithm was successfully applied in the training process of multilayer feedforward neural networks [14]. A network was used to solve the approximation problem of a multi-variate function. Owing to the presence of many local minima in the error function to be minimized, ESSS and ESSS-SVA algorithms were explored. They were compared with the Back-Propagation algorithm, which was the most popular existing approach. Simulation experiments show that the evolutionary algorithms considered here are effective in the sense of finding global minima of the error function.

4.2 ESSS-DOF algorithm

The effectiveness of the ESSS and ESSS-SVA algorithms is limited in the case of a fitness function which consists of a group of concentrated local optimum peaks and other distant peaks. If the population starts near this local group, then it cyclically jumps from peak to peak of the group and cannot move towards a remote one. In order to overcome this problem, *the erosion mechanism* has been proposed in the ESSS-DOF. The erosion is inspired by the natural mechanism: if the population finds an ecological niche, where the existence conditions are much better than in the neighborhood, then it has troubles with running away toward other areas. The population is evolutionary trapped. The long-standing exploitation of the niche is the cause of its deterioration. The existence conditions of the population go down. A few individuals try to explore toward other unknown areas.

The ESSS-DOF influences on the topology of the objective function. If an evolutionary trap (*Trap test*) is detected, then *Erosion* is activated. This procedure transforms the objective function $\Phi(\mathbf{x})$ as follows:

$$\Phi(\mathbf{x}) = \begin{cases} \Phi(\mathbf{x}) - G(\mathbf{x}) & \text{for } \Phi(\mathbf{x}) \geq G(\mathbf{x}), \\ 0 & \text{for } \Phi(\mathbf{x}) < G(\mathbf{x}), \end{cases} \tag{17}$$

where

$$G(\mathbf{x}) = q^t_{max} \exp\left(-\frac{1}{2}(\mathbf{x} - \langle \mathbf{x}^t \rangle)^T \mathbb{E}^{-1} (\mathbf{x} - \langle \mathbf{x}^t \rangle) \right), \tag{18}$$

and $\langle \mathbf{x}^t \rangle$ is the expectation vector of $P(t)$. The covariance matrix $\mathbb{E}$ of the erosion peak has to be chosen in the form which allows for deterioration of the occupied peak as effectively as possible. If this peak can be approximated by a Gaussian peak, then its covariance matrix can be calculated from (16) using the following procedure:

1. Calculate the covariance matrix $\mathbb{C}_t$ of the actual population;
2. Find all the eigenvectors and eigenvalues of the matrix $\mathbb{C}_t$ in order to define an orthonormal matrix $\mathbb{U}$ and a diagonal matrix $\mathrm{diag}(\nu^2_{ti} | i = 1, 2, \ldots, n)$ such that
$$\mathbb{C}_t = \mathbb{U} \, \mathrm{diag}(\nu^2_{ti} | i = 1, 2, \ldots, n) \mathbb{U}^T; \tag{19}$$
3. Calculate the variances of the erosion peak (16):
$$\tau^2_i = \nu^2_{ti} \left(\frac{\nu^2_{ti}}{\sigma^2} - 1 \right); \tag{20}$$
4. Calculate the covariance matrix $\mathbb{E}$ of the erosion peak:
$$\mathbb{E} = \mathbb{U} \, \mathrm{diag}(\tau^2_i | i = 1, 2, \ldots, n) \mathbb{U}^T. \tag{21}$$

Finally, the ESSS-DOF algorithm can be written in the following form:

1. *Initiation*;
2. Repeat
   (a) *Estimation*;
   (b) *Choice of the best element in the history*;
   (c) If *Trap Test* then *Erosion*;
   (d) *Selection*;
   (e) *Mutation*;

   Until $t > t_{\max}$.

The ESSS-DOF algorithm has a much greater convergence rate than the other algorithms from the ESSS family. If the population gets stuck in an evolutionary trap, then the process of local peak erosion is started. This effect decreases the average fitness of the population. The population fitness reduces to a saddle level and running away towards other quality peak is possible. The deteriorated peak will never be attractive for the searching population. The disadvantage of the ESSS-DOF is its numerical complexity.

4.3 ESSS-FDM

Although, the ESSS-DOF algorithm seems to be the most effective algorithm from those presented above , it cannot be applied in the case of a time-varying adaptation landscape. In order to overcome this problem, the ESSS-FDM algorithm has been proposed. The idea of FDM mechanism possesses a biological inspiration. If natural conditions existing in the environment reward some direction of alteration in the phenotype space, then this direction is preferred not only by selection but also by mutation.

The ESSS-FDM algorithm differs from the standard ESSS algorithm only in the modification step. The selected elements are mutated by adding to each component i a normally-distributed random variable with expectation $m_i \neq 0$, unlike the ESSS algorithm, where $m_i = 0$ in (6). Hence (see Tab.1):

(d) *Modification*

$$\begin{aligned} &P(t) \to P(t+1); \\ &\left(\mathbf{x}_k^{t+1}\right)_i = \left(\mathbf{x}_{h_k}^{t+1}\right)_i + N(m_i^t, \sigma); \\ &\qquad i = 1, 2, \ldots, n; \quad k = 1, 2, \ldots, \eta; \\ &m_i^t = \mu\sigma \frac{E_i(P(t)) - E_i(P(t-1))}{\|E_i(P(t)) - E_i(P(t-1))\|}; \\ &E_i(P(t)) = \frac{1}{\eta} \sum_{k=1}^{\eta} \left(\mathbf{x}_k^t\right)_i. \end{aligned} \tag{22}$$

The modified expectation vector $\mathbf{m}^t$ depends on the standard deviation σ and is parallel to the latest trends of the population drift. The exogenous parameter μ, which is called momentum, determines the proportion between the standard deviation σ and the length of the vector $\mathbf{m}^t$: $\mu = \|\mathbf{m}^t\|/\sigma$. If μ is too small, then there is no difference between the ESSS-FDM and the ESSS. In the case of a very large μ $\left(\|\mathbf{m}^t\| \gg \sigma\right)$, there is no possibility of changing the population drift direction, which has been chosen in the beginning of the searching process.

5 Experiment

Many simulations (about 1600) with nine two-variable objective functions have been carried out. Test functions used during simulations are listed below:

- function f_1 (sum of two Gaussian peaks):

$$f_1(x_1, x_2) = \exp(-x_1^2 - x_2^2) + \frac{1}{2}\exp(-(x_1 - 2.3)^2 - x_2^2), \tag{23}$$

- function f_2 (De Jong's function F2):

$$f_2(x_1, x_2) = 3500 - 100(x_1^2 - x_2)^2 - (1 - x_1)^2, \tag{24}$$

- function f_3 (De Jong's function F5):

$$f_3(x_1, x_2) = 500 - \left\{0.002 + \sum_{j=1}^{25} \left[j + \sum_{i=1}^{2} (x_i - a_{ij})^6\right]^{-1}\right\}^{-1}, \tag{25}$$

$$(a_{ij}) = \begin{pmatrix} -32 & -32 \\ -16 & -32 \\ 0 & -32 \\ 16 & -32 \\ 32 & -32 \\ -32 & -16 \\ \vdots & \vdots \\ -32 & 32 \\ -16 & 32 \\ 0 & 32 \\ 16 & 32 \\ 32 & 32 \end{pmatrix}^T \tag{26}$$

- function f_4:

$$f_4(x_1, x_2) = \frac{1 + \cos\left(12\sqrt{x_1^2 + x_2^2}\right)}{\frac{1}{2}\left(x_1^2 + x_2^2\right) + 2}, \tag{27}$$

- function f_5 (Michalewicz's function):

$$f_5(x_1, x_2) = \sin(x_1)\left(\sin(x_1^2/\pi)\right)^{20} + \sin(x_2)\left(\sin(x_2^2/\pi)\right)^{20}, \tag{28}$$

Algorithm	Parameters	Values
all	$t_{\max}$	1000
	η	20
	σ	0.05
ESSS-SVA	α	1.1
ESSS-SVA and ESSS-DOF	t_t	10
ESSS-FDM	μ	0.3

Table 2: Parameter values used in the simulations.

- function f_6 (Shubert's function):

$$f_6(x_1, x_2) = 200 + \sum_{i=1}^{5} i\cos\big((i+1)x_1 + 1\big) \times \sum_{i=1}^{5} i\cos\big((i+1)x_2 + 1\big), \tag{29}$$

- function f_7 (Rastring's function):

$$f_7(x_1, x_2) = 100 - \left(x_1^2 + x_2^2\right) - 10\left(\cos(2\pi x_1) + \cos(2\pi x_2)\right), \tag{30}$$

- function f_8 (Acley's function):

$$f_8(x_1, x_2) = 5 + 20\exp\left[-\frac{1}{2}\sqrt{\frac{1}{2}\left(x_1^2 + x_2^2\right)}\right] - \exp\left\{\frac{1}{2}\left[\cos(2\pi x_1) + \cos(2\pi x_2)\right]\right\}. \tag{31}$$

All those functions are strongly non-linear and multimodal. The fitness function has been chosen in the form :

$$\Phi\left(\mathbf{x}_k^t\right) = f\left(\mathbf{x}_k^t\right) - f_{\min}^t + \left(\frac{1}{\eta}\right)^2 \tag{32}$$

where $f_{\min}^t = \min\left(f\left(\mathbf{x}_k^t\right)|k = 1, \ldots, \eta\right)$ is the minimal value of f taken over all the elements in the actual population, and f is a given objective function which has to be maximized. Such a fitness function is non-negative and its relative values in the actual population make the proportional selection (5) effective.

At first, simulations were carried out several times for different sets of input parameters. When the best set of parameters was allocated for each algorithm (see Table 2), several starting points were tested. The results are compared in Table 3.

Analysis of results shows that all mechanisms (SVA, FDM and DOF) applied to the standard ESSS algorithm accelerate the crossing of the objective function saddles and increase the effectiveness of the global optimum finding. Two algorithms, ESSS-SVA and ESSS-DOF, compete which is the best. The ESSS-SVA seems to be the most effective algorithm. But the ESSS-DOF wins in the case of a fitness function which consists of a group of concentrated local optimum peaks and other distant peaks. If the population in the ESSS-SVA starts in the area of this local group, then it cyclically moves from peak to peak of the group and cannot achieve a remote one. The ESSS-DOF erodes peaks in turn and slowly, but consequently, leads toward the global optimum.

function	ESSS	ESSS -SVA	ESSS -FDM	ESSS -DOF
f_1	38	100	87	100
f_2	53	88	79	100
f_3	0	42	58	0
f_4	0	37	0	0
f_5	12	27	13	41
f_6	22	98	81	39
f_7	26	59	74	23
f_8	0	69	0	13

Table 3: Percentages of runs, in which the global optimum has been found.

6 Conclusions

The aim of this work has been the effectiveness analysis of the Evolutionary Search with Soft Selection in the global parameter optimization. Apart from the ESSS algorithm, its three biologically inspired variants were considered: ESSS-SVA, ESSS-DOF and ESSS-FDM. All of them are more effective than the standard ESSS. Especially the ESSS-SVA and ESSS-DOF seem to be useful in technical applications.

All the objective functions used in work are two-dimensional. It would be interesting to analyze whether the effectiveness of the SVA, DOF and FDM mechanisms is kept when increasing the searching landscape dimension. This problem determines our plans for further research.

References

[1] Arabas J., Michalewicz Z. and Mulawka J. *GAVaPS - a genetic algorithm with varying population size* Proc. 1st IEEE Int. Conf. *Evolutionary Computation*, Orlando, USA, Vol.2, pp.73–78, 1994.

[2] Fogel L.J., Owens A.J. and Walsh M.J. *Artificial Intelligence through Simulated Evolution*, Wiley, New York, 1966.

[3] Galar R. *Handicapped individua in evolutionary processes*, Biol. Cybern., Vol.51, pp.1–9, 1985.

[4] Galar R. *Soft selection in random global adaptation in R^n A biocybernetic model of development*, Technical University Press, Wrocław, 1990, (in Polish).

[5] Goldberg D.E. *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc., Reading, MA, 1989.

[6] Holland J.H. *Adaptation in natural and artificial systems*, The University of Michigan Press, Ann Arbor, MI, 1975.

[7] Karcz-Dulęba I. *Simulating Evolutionary Processes as a Tool of Global Optimization in* $\mathbb{R}^n$, PhD thesis, Technical University Press, Wrocław, 1992, (in Polish).

[8] Karcz-Dulęba I. *Some convergence aspects of evolutionary search with soft selection method*, Proc. 2nd Conf. *Evolutionary Algorithms and Global Optimization*, Rytro, Poland, pp.113–120, 1997, (in polish).

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

[10] Obuchowicz A. *The evolutionary search with soft selection and deterioration of the objective fuction*, Proc. 6st Int. Symp. *Intelligent Information Systems*, Zakopane, Poland, pp.288–295, 1997.

[11] Obuchowicz A. and Korbicz J. *Evolutionary search with soft selection and forced direction of mutation*, Proc. 7st Int. Symp. *Intelligent Information Systems*, Malbork, Poland, pp.300–309, 1998.

[12] Obuchowicz A. and Korbicz J. *Evolutionary search with soft selection algorithms in parameter optimization* Proc. 3rd Int. Conf. *Parallel Processing and Applied Mathematics*, Kazimierz Dolny, Poland, pp.578–586, 1999.

[13] Obuchowicz A. and Patan K. *About some modification of evolutionary search with soft selection algorithm*, Proc. 2nd Conf. *Evolutionary Algorithms and Global Optimization*, Rytro, Poland, pp.193–200, 1997, (in Polish).

[14] Obuchowicz A. and Patan K. *An algorithm of evolutionary search with soft selection for training multilayer feedforward neural networks.* Proc. 3rd Conf. *Neural Networks and Their Applications*, Kule, Poland, pp.123–128, 1997.

[15] Rechenberg I. *Cybernetic solution path of an experimental problem*, Roy. Aircr. Establ., libr. Transl. 1122, Farnborough, Hants., UK, 1965.