

Experimental Study of Selected Parameters of the Krill Herd Algorithm

Piotr A. Kowalski^{1,2}, Szymon Łukasik^{1,2}

¹ Systems Research Institute, Polish Academy of Sciences,
ul. Newelska 6, PL-01-447 Warsaw, Poland,
{pakowal,slukasik}@ibspan.waw.pl,

² Department of Automatic Control and Information Technology,
Cracow University of Technology
ul. Warszawska 24, PL-31-155 Cracow, Poland
{pkowal,szymon1}@pk.edu.pl

Abstract. The Krill Herd Algorithm is the latest heuristic technique to be applied in deriving best solution within various optimization tasks. While there has been a few scientific papers written about this algorithm, none of these have described how its numerous basic parameters impact upon the quality of selected solutions. This paper is intended to contribute towards improving the aforementioned situation, by examining empirically the influence of two parameters of the Krill Herd Algorithm, notably, maximum induced speed and inertia weight. These parameters are related to the effect of the herd movement as induced by individual members. In this paper, the results of a study – based on certain examples obtained from the CEC13 competition – are being presented. They appear to show a relation between these selected two parameters and the convergence of the algorithm for particular benchmark problems. Finally, some concluding remarks, based on the performed numerical studies, are provided.

Key words: Krill Herd Algorithm, Biologically Inspired Algorithm, Optimization, Metaheuristic.

1 Introduction

The optimization problems [5] are encountered when deriving solutions to many engineering issues [15]. Optimization can be considered as choosing of the best possible use of limited resources (time, money, etc.) while attempting to bring about certain goals. In achieving this, the optimization problem can be written in a formal way. Let us introduce the 'cost function' K :

$$K : A \mapsto \mathbb{R}, \quad (1)$$

(in some instances, this function can be also called the 'fitness function') where $A \subset \mathbb{R}^n$. The optimization task consists of finding the value $x^* \in A$, such that for every $x \in A$ the following relationship is true:

$$K(x) \geq K(x^*). \quad (2)$$

Although the optimization problem can be easily defined and described, determining its solution is already a very difficult issue. To find solutions for this problem, certain optimization algorithms are commonly used. Generally, these are placed within two classes: deterministic and stochastic algorithms.

Deterministic algorithms operate according to strictly defined procedures, and the results obtained through employing them are repeatable. A simple and well-known example of this group is the numerical algorithm called 'Newton's Method' [5]. This can easily be used to find the minimum of the function. Assuming no change in the input data, the intermediate results and the final result obtained in subsequent runs of the algorithm will be identical.

On the other hand, we have the stochastic algorithms. These have a random factor and each execution of some algorithm belonging to this class implies that results will vary. Typically this group of algorithms allows users to obtain solution in a short time, but very often it constitutes only an approximation of optimum value. Among stochastic algorithms that can be distinguished there exists a group of procedures inspired by Nature. These are divided into three main streams: evolutionary algorithms, artificial immune systems, and swarm intelligence.

Evolutionary Algorithms [1] are based on biological evolution, and they are built upon conceptualising a population that adapts to its surrounding world by using mechanisms such as selection, reproduction, mutation and crossover. There are many implementations of the evolutionary approach. Among these we can distinguish Genetic Algorithms, Genetic Programming, and Evolutionary Strategies [9]. In each of those, a population is created which evolves in successive iterations towards the best solution.

Artificial Immune Systems [2] are adaptive systems inspired by the results of studies on the immune systems of living organisms. Their main task is to detect anomalies. Examples of such systems are the Negative Selection Procedure and the Clonal Selection Procedure.

Swarm intelligence, also known as the 'intelligent group' [4], is a decentralized self-organizing system which is formed by a population consisting of a number of individuals called 'agents'. In this approach, there is no commanding authority that determines the way the agents behave. Instead, individual 'animals' roam the space of proposed solutions, while following few simple rules. The most well-known implementations of this are: Ant Colony Optimization [3], Bee Colony Algorithm [8], Particle Swarm Optimization [14], Glowworm Swarm Optimization [13], and Firefly Algorithm [12]. In 2012, A. H. Gandomi and A. H. Alavi contributed to these by developing Krill Herd Algorithm (KHA) [6].

The goal of the paper is to reveal the impact of selected parameters in KHA, to the quality and convergence speed of an optimization procedure. Presented tests empirically provide suggested strategies of selecting two parameters within the Krill Herd Algorithm: maximum induced speed and inertia weight. These quantities are related to the part of the herd movement that is induced by other swarm members. Parameter choice is based upon numerical simulation results which indicate solution's quality: average error value with its standard

deviation, the best result obtained by the swarm, and, additionally, the speed of its convergence. Finally, to increase the quality of solutions, a procedure for decreasing the value of the inertia weight parameter has also been proposed.

2 Krill herd algorithm

Antarctic krill (*Euphausia superba*) is a species of krill found in the Southern Ocean. An adult body has a length of 6 cm, and its weight comes about 2 grams. This species feeds on phytoplankton. A characteristic feature of krill, and the inspiration of the present algorithm [6, 16], is the ability for individual krill to be moulded within a large herd that is even hundreds of meters in length.

Algorithm 1 Krill Herd Algorithm

```

1:  $k \leftarrow 1$  {initialization}
2: Initialize parameters ( $D^{max}$ ,  $N^{max}$ , etc.)
3: for  $i = 1$  to  $M$  do
4:   Generate Solution ( $x_i(k)$ )
5:   {evaluate and update best solutions}
6:    $K(x_i(0)) \leftarrow$  Evaluate quality( $x_i(0)$ )
7: end for
8:  $x^* \leftarrow$  Save best individual  $x^*(0)$ 
9: {main loop}
10: repeat
11:   sort population of krills
12:   for  $i = 1$  to  $M$  do
13:     Perform motion calculation and genetic operators:
14:      $N_i \leftarrow$  Motion induced by other individuals
15:      $F_i \leftarrow$  Foraging activity
16:      $D_i \leftarrow$  Random diffusion
17:     Crossover
18:     Mutation
19:     {update krill position}
20:     Update Solution ( $x_m(k)$ )
21:     {evaluate and update best solutions}
22:      $K(x_i(k)) \leftarrow$  Evaluate quality( $x_i(k)$ )
23:   end for
24:    $x^* \leftarrow$  Save best individual  $x^*(k)$ 
25:   stop condition  $\leftarrow$  Check stop condition ()
26:    $k \leftarrow k + 1$ 
27: until stop condition = false
28: return  $K(x^*(k))$ ,  $x^*(k)$ ,  $k$ 

```

The Krill Herd procedure is shown in Algorithm 1. Operation of the algorithm starts with the initialization of data structures, i.e. describing individuals, as well as the whole population. Initializing data structure representing a krill, means

situating it in a certain place (at the 'solution space') by giving it a set of coordinates. For this purpose, it is recommended to employ random number generation according to a uniform distribution. Like other algorithms inspired by Nature, each individual represents one possible solution of the problem under consideration.

After the initialization phase of the algorithm, it follows into a series of iterations. The first step of each is to calculate fitness function value for each individual of the population. This is equivalent to the calculation of optimized functions with arguments which are the coordinates of the krill's position. Then, for each individual of the population, the vector which indicates the displacement in the solution space is calculated. The movement of krill is described by an equation dependent on three factors:

$$\frac{dX_i}{dt} = N_i + F_i + D_i, \quad (3)$$

where N_i refers to the movement induced by the influence of other krills (subsection 2.1), F_i is a movement associated with the search for food (subsection 2.2), and D_i constitutes a random diffusion factor (subsection 2.3).

In order to improve efficiency, the algorithm includes two genetic operators: crossover and mutation (subsection 2.4). This phase is optional, implementation of these operators can be completely omitted or only one of them can be employed. As shown by certain preliminary tests [6] the Krill Herd Algorithm achieves the best results when only the crossover operator is implemented.

The next step is to update the position of krills in the solution space, in accordance with the designated movement vectors (3). In the next iteration, based on these values, the new value of the fitness function will be calculated.

A termination of the algorithm can be implemented in many ways. One possibility is to stop it when the solution achieves or is below predetermined cost function value. This approach is used where the main purpose is accuracy. The second way is to terminate the procedure upon reaching a predetermined number of iterations. This method is applicable in situations where the solution must be achieved in a fixed time. The most common method is a combination of both conditions.

2.1 Motion induced by other individuals

The movement stimulated by the presence of other individuals (krills) is defined as:

$$N_i = N^{max} \alpha_i + \omega N_i^{old}, \quad (4)$$

where

$$\alpha_i = \alpha_i^{local} + \alpha_i^{target}. \quad (5)$$

In the above formulas N^{max} constitutes the maximum speed induced by the presence of other krills, and it is determined experimentally (in paper [6] expressing this value as 0.01 was recommended – this recommendation rate is based on

the maximum induced speed of the krill herd [7]. Furthermore, the parameter ω describes the inertia weight of the induced motion, and takes the value from the interval $[0, 1]$. What is more, N_i^{old} is the last iteration value limiting this motion. Moreover, α_i^{local} denotes the local effect provided by neighbouring krill, and α_i^{target} constitutes a target direction effect based on the position of the best individuals. The neighbours effect, connected with attractiveness, can be described as:

$$\alpha_i^{local} = \sum_{j=1}^{M_{neighbours}} \hat{K}_{i,j} \hat{X}_{i,j}. \quad (6)$$

Where the variable $\hat{K}_{i,j}$ and vector $\hat{X}_{i,j}$ are defined as follows:

$$\hat{X}_{i,j} = \frac{X_i - X_j}{\|X_i - X_j\| + \epsilon}, \quad (7)$$

$$\hat{K}_{i,j} = \frac{K_i - K_j}{K^{worst} - K^{best}}. \quad (8)$$

Here K^{worst} and K^{best} are the worst and best values of the fitness function within a population (the wording "best" means the smallest value of $K(X_i)$, taking into account the fact that a minimum of a function (1) is to be located). In addition the variable K_i in equation (8) represents the fitness function's value of the i -th krill, and the K_j j -th neighbour for $j = 1, 2, \dots, M_{neighbours}$. Moreover a notation $M_{neighbours}$ expresses the number of krill neighbours. In order to avoid zero coming about within the equation (7), in the denominator, a factor ϵ is added, which is a small positive number. Finally, the neighbours of the krill are those individuals that are in range, i.e. in the area of a circle centred at the position of the i -th krill, and which has a radius equal to:

$$d_{s,i} = \frac{1}{5N} \sum_{j=1}^M \|X_j - X_i\|, \quad (9)$$

where M represents the total number of individuals in a population.

The movement of krill is also dependent on the best individual location. This factor determines the value α_i^{target} in following way:

$$\alpha_i^{target} = \gamma^{best} \hat{K}_{i,best} \hat{X}_{i,best}, \quad (10)$$

where γ^{best} is the ratio of the individual impact with the best fitness function value for the i -th krill:

$$\gamma^{best} = 2\left(\xi + \frac{k}{I_{max}}\right). \quad (11)$$

In the previous equation, ξ is a random number coming from the interval $[0, 1]$, while k denotes the number of the current iteration, and I_{max} indicates a maximum number of iterations.

2.2 Foraging activity

The movement associated with the search for food (foraging) depends on two components. The first one is the current food location and the second points out the previous location of food intake. For the i -th individual, this component of movement is defined as:

$$F_i = V_f \beta_i + \omega_f F_i^{old}, \quad (12)$$

where

$$\beta_i = \beta_i^{food} + \beta_i^{best}. \quad (13)$$

Furthermore, quantity V_f describes the speed of searching for food, and has been selected empirically. Its recommended value is 0.02. The notation ω_f denotes the inertia weight of the foraging motion in range $[-1, 1]$. The location of food is the quantity that for KHA is defined on the basis of the distribution of the fitness function. It is given by following equation:

$$X^{food} = \frac{\sum_{i=1}^M K_i^{-1} X_i}{\sum_{i=1}^M K_i^{-1}}. \quad (14)$$

The previous equation participates in determining a quantity which is a measure of the impact of food on the i -th krill:

$$\beta_i^{food} = C^{food} \hat{K}_{i,food} \hat{X}_{i,food}, \quad (15)$$

where C^{food} represents a coefficient determining the influence of the impact of food location on the i -th krill

$$C^{food} = 2\left(1 - \frac{k}{I_{max}}\right). \quad (16)$$

In this algorithm, the best individual and its position are also included in accordance with the following formula:

$$\beta_i^{best} = \hat{K}_{i,best} \hat{X}_{i,best}. \quad (17)$$

2.3 Physical diffusion

Physical diffusion is a random process, and this vector is defined as follows:

$$D_i = D^{max} \left(1 - \frac{k}{I_{max}}\right) \delta, \quad (18)$$

where $D^{max} \in [0.002, 0.01]$ is the maximal diffusion speed, while denoted as δ represents the random directional vector, and its elements belong within the interval $[-1, 1]$.

The previously described factors affect that the krill additionally changes its position to some extent randomly over time.

Finally, its location at time $t + \Delta t$ is determined as follows:

$$X_i(t + \Delta t) = X_i(t) + \Delta t \frac{dX_i}{dt}, \quad (19)$$

wherein Δt is the scaling factor for the speed of the search of the solution space, and is defined as:

$$\Delta t = C_t \sum_{j=1}^N (UB_j - LB_j). \quad (20)$$

In the above formula, the condition $C_t \in [0, 2]$ is fulfilled, while N is search space dimensionality. In addition, UB_j and LB_j are the upper and lower limitations of j -th coordinates of vector X .

2.4 Genetic operators

The final stage of the main iteration in KHA is the use of genetic operators [17]. The crossover results in a change of the m -th coordinate of i -th krill as shown below by the formula:

$$x_{i,m} = \begin{cases} x_{r,m} & \text{for } \gamma \leq Cr \\ x_{i,m} & \text{for } \gamma > Cr \end{cases} \quad (21)$$

where $Cr = 0.2\hat{K}_{i,best}$; $r \in \{1, 2, \dots, i-1, i+1, \dots, N\}$ denotes a random index, and γ is a random number from the interval $[0, 1)$ generated according to the uniform distribution. In this approach the crossover operator is acting on a single individual.

The mutation modifies the m -th coordinate of the i -th krill in accordance with the formula:

$$x_{i,m} = \begin{cases} x_{gbest,m} + \mu(x_{p,m} - x_{q,m}) & \text{for } \gamma \leq Mu \\ x_{i,m} & \text{for } \gamma > Mu \end{cases} \quad (22)$$

wherein $Mu = 0.05/\hat{K}_{i,best}$; $p, q \in \{1, 2, \dots, i-1, i+1, \dots, K\}$ and $\mu \in [0, 1)$.

Additional information about the Krill Herd Algorithm in [16, 18] could be found.

3 Numerical studies

In this section, the performance of the KHA metaheuristic for different sets of selected parameters connected with motion is verified – by running a series of experiments related to various global numerical optimization tasks. One of the main goals in presenting these numerical experiments is to show the dynamics of an optimization process based on KHA for diverse sets of the algorithm's parameters. In doing so, the minimal values of an error, as well as a set of mean values with their standard deviations were derived in order to perform such a comparison.

For numerical experiments, we applied a selected set of benchmark functions considered in the CEC'13 competition [11]. Table 1 reveals a list of functions with their formulas, dimensionality, feasible bounds and optimal values. From the set of benchmark tasks, six functions were selected, the first three having

Table 1. Benchmark functions used for experimental study

K	Name	Expression	Feasible bounds	N	$K(x^*)$
K1	Sphere	$K_1(x) = \sum_{i=1}^N z_i^2 + K_1^*$ $z = x - o$	$[-100, 100]^N$	10	-1400
K2	Rotated Bent Cigar	$K_2(x) = z_1^2 + 10^6 \sum_{i=2}^N z_i^2 + K_2^*$ $z = M_2 T_{0.5}^{0.5} (M_1(x - o))$	$[-100, 100]^N$	10	-1000
K3	Different Powers	$K_3(x) = \sqrt{\sum_{i=1}^N z_i ^{2+4\frac{i-1}{N-1}}} + K_3^*$ $z = x - o$	$[-100, 100]^N$	10	-1000
K4	Rotated Rastrigin	$K_4(x) = \sum_{i=1}^N (z_i^2 - 10 \cos(2\pi z_i) + 10) + K_4^*$ $z = M_1 A^{10} M_2 T_{0.2}^{0.2} (T_{0.8} (M_1 \frac{5.12(x-o)}{100}))$	$[-100, 100]^N$	10	-300
K5	Schwefel	$K_5(x) = 418.9829N \sum_{i=1}^N g(z_i) + K_5^*$ $z = A^{10} (\frac{1000(x-o)}{100}) + 4.209687462275036e + 002$ $g(z_i) = z_i \sin(z_i ^{1/2})$	$[-100, 100]^N$	10	-100
K6	Rotated Katsuura	$K_6(x) = \frac{10}{N^2} \prod_{i=1}^N (1 + i \sum_{j=1}^{32} \frac{ 2^j z_i - \text{round}(2^j z_i) }{2^j})^{\frac{10}{N^2}} - \frac{10}{N^2} + K_6^*$ $z = M_2 A^{100} (M_1 \frac{5(x-o)}{100})$	$[-100, 100]^N$	10	200

an unimodal character, while the others are of a multimodal type. Due to text limitation, a description of the symbols used in Table 1 can be found in [11].

The experiments were carried out for a fixed number of iterations $I_{max} = 10000$ ($1000N$), and 30 trials were undertaken for each benchmark function. In all numerical experiments the population size $M = 40$ was used. What is more KHA was configured with the parameters given in section 2. However, this does not apply to the parameters of the algorithm described in subsection 2.1, which are the subject of this study.

As a performance measure the mean optimization error \hat{E}_{mean} was used (with $\hat{E}(k) = |K(x^*(x)) - K^*|$) along with its standard deviation $\sigma_{\hat{E}}$. In addition, as a measure of the optimization algorithm quality, a minimal error is also reported.

3.1 Selection of N^{max} parameter

The first set of tests was associated with the selection of the N^{max} parameter. This can be defined as maximum speed induced by the presence of other individuals. In order to investigate the best value of this parameter, the value of N^{max} was set in following order $\{0.005, 0.010, 0.015, 0.020, 0.025\}$. Moreover, in these tests, the decreasing character of the parameter w was proposed according to the following formula:

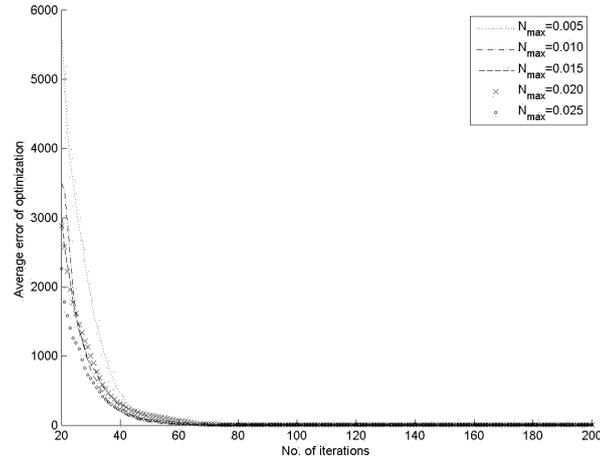
$$w = 0.1 + 0.8 * \left(1 - \frac{k}{I_{Max}} \right). \quad (23)$$

The results of this investigation can be seen in Table 2, as well as in the exemplary diagrams (Figures 1 and 2). These show the first 200 iterations of the algorithm's convergence for the unimodal (K1) and multimodal (K4) functions, respectively.

In interpreting the presented results, the following remarks can be concluded. If we optimize the function of the unimodal character, the smallest parameter value of the N^{max} should be chosen. In addition, for the K1 and K3 functions,

Table 2. Optimisation results for benchmark functions with varying N^{max}

		N^{max}				
		0.005	0.010	0.015	0.020	0.025
K1	\hat{E}_{min}	0.00000	0.00000	0.00000	0.00000	0.00000
	\hat{E}_{mean}	0.00000	0.00000	0.00000	0.00001	0.00001
	$\sigma_{\hat{E}}$	0.00000	0.00000	0.00000	0.00000	0.00000
K2	\hat{E}_{min}	4.993E+03	1.124E+03	7.949E+02	2.750E+02	1.961E+02
	\hat{E}_{mean}	1.617E+08	2.797E+07	1.246E+07	5.227E+07	2.008E+07
	$\sigma_{\hat{E}}$	5.436E+08	7.600E+07	3.279E+07	9.625E+07	3.836E+07
K3	\hat{E}_{min}	0.00005	0.00002	0.00003	0.00006	0.00004
	\hat{E}_{mean}	0.00011	0.00008	0.00009	0.00011	0.00013
	$\sigma_{\hat{E}}$	0.00004	0.00004	0.00004	0.00005	0.00005
K4	\hat{E}_{min}	14.92439	9.94961	9.94960	4.97481	8.95464
	\hat{E}_{mean}	47.19417	38.13998	31.44063	25.30509	29.94820
	$\sigma_{\hat{E}}$	17.08109	16.87140	15.33291	12.15950	13.50465
K5	\hat{E}_{min}	269.95241	309.41975	496.45608	285.41008	367.79078
	\hat{E}_{mean}	992.74120	953.26188	846.03055	786.37048	775.80619
	$\sigma_{\hat{E}}$	346.93644	288.11901	270.35735	365.50859	277.75687
K6	\hat{E}_{min}	0.02427	0.03769	0.02559	0.05169	0.03126
	\hat{E}_{mean}	0.14897	0.13162	0.12815	0.15088	0.15652
	$\sigma_{\hat{E}}$	0.10647	0.08539	0.08918	0.08900	0.09986

**Fig. 1.** Convergence of KHA with varying N^{max} for the benchmark function K1

the best results for N^{max} belong to the interval range $[0.005, 0.01]$. On the other hand, it can be seen that an increase in of this parameter results in a faster convergence of the optimization algorithm. For the functions of a multimodal character (K4-K6), a higher value of this parameter (even to 0.025) is recommended. This is due to the presence of numerous local minima. Thus increasing the value of this parameter allows for a stronger influence upon the solutions

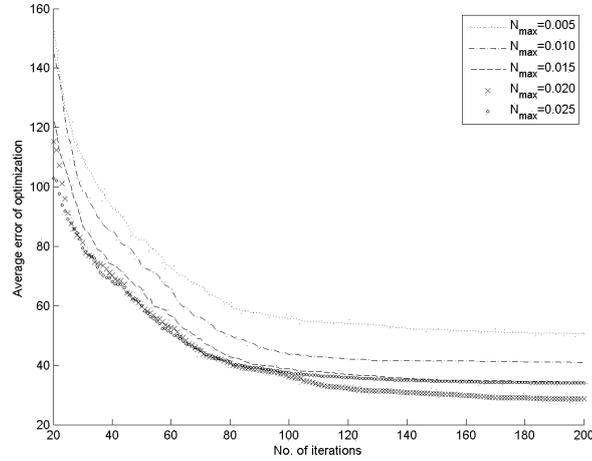


Fig. 2. Convergence of KHA with varying N^{max} for the benchmark function K4

derived from the previous iteration. This results in an easier way to escape from a local extreme.

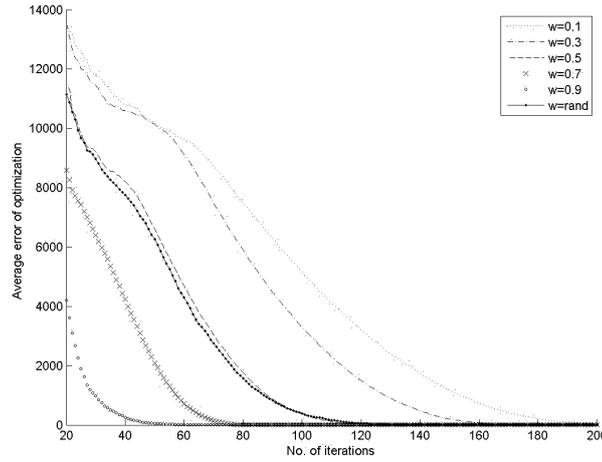
3.2 Selection of inertia weight parameter

In this subsection, the performance comparison of using KHA for varying inertia weight parameter is described. In order to perform a numerical study, a parameter w was taken as being $\{0.1, 0.3, 0.5, 0.7$ and $0.9\}$. Additionally, for comparison, a variant is described in which the added value of this parameter was generated randomly from the range $[0, 1]$ according to the uniform distribution. In this situation, a comparison also should be made to the case of decreasing the parameter that was reported in the previous subsection. To conduct the numerical tests, based on the results obtained in the last subsection, the following values of N^{max} parameter 0.01, 0.01, 0.01, 0.7 and 0.9 for the corresponding function K were set out. The results of these numerical experiments can be seen in Table 3, as well as in the exemplary diagrams (Figures 3 and 4), which show the first 200 iterations of algorithm convergence for unimodal (K1) and multimodal (K4) functions, respectively.

In this quantitative comparison, both means with standard deviations and best obtained values of error were reported. The results show that here as well, two cases should be considered. An example of the former are the unimodal functions $K1 - K3$. For these instances, the smaller value of the parameter w is preferred. In this research, the best results were obtained for $w = 0.1$. The exception was the K2 function, here, the best results were obtained by way of the random inertia weight parameter. The second case incorporates examples of multimodal functions ($K4 - K6$). In this situation, we recommend a set up of

Table 3. Optimisation results for benchmark functions with varying inertia weight w

		w					
		0.100	0.300	0.500	0.700	0.900	rand
K1	\hat{E}_{min}	0.00000	0.00000	0.00000	0.00007	0.03219	0.00001
	\hat{E}_{mean}	0.00000	0.00001	0.00002	0.00019	0.07729	0.00002
	$\sigma_{\hat{E}}$	0.00000	0.00000	0.00001	0.00007	0.02606	0.00001
K2	\hat{E}_{min}	6.348E+03	6.757E+02	1.911E+02	8.015E+02	5.140E+04	2.915E+01
	\hat{E}_{mean}	1.234E+08	2.750E+07	2.133E+07	7.599E+07	3.060E+07	2.206E+07
	$\sigma_{\hat{E}}$	2.584E+08	6.249E+07	7.178E+07	1.794E+08	5.036E+07	6.098E+07
K3	\hat{E}_{min}	0.00001	0.00009	0.00026	0.00056	0.04050	0.00028
	\hat{E}_{mean}	0.00003	0.00024	0.00087	0.00175	0.09250	0.00050
	$\sigma_{\hat{E}}$	0.00001	0.00008	0.00038	0.00100	0.02360	0.00017
K4	\hat{E}_{min}	50.74270	41.78823	28.85386	42.78334	9.05741	23.87924
	\hat{E}_{mean}	123.30676	115.57938	91.40320	82.58270	33.56695	100.25815
	$\sigma_{\hat{E}}$	37.03286	46.96247	42.16632	29.73748	17.12788	38.81281
K5	\hat{E}_{min}	388.25117	283.30526	467.88315	390.25919	173.34292	374.17893
	\hat{E}_{mean}	795.00710	810.78165	808.17313	962.54859	467.21808	955.54884
	$\sigma_{\hat{E}}$	319.50109	276.70611	200.73907	274.54883	139.34147	279.48430
K6	\hat{E}_{min}	0.08638	0.02508	0.01941	0.08221	0.60140	0.05913
	\hat{E}_{mean}	0.26328	0.17722	0.16912	0.26530	0.96770	0.26897
	$\sigma_{\hat{E}}$	0.16465	0.12818	0.10473	0.13189	0.19718	0.17963


Fig. 3. Convergence of KHA with varying w for the benchmark function K1

$w = 0.9$. However, if we look at the results of the best value of cost function, then the best solution seems to be to adopt a variable parameter w derived by way of formula (23). Based on the presented Figures 3 and 4, we can easily conclude that the increase of the parameter w influences a faster convergence. Finally, it is also worth to note that KHA was found to underperform for some of examined

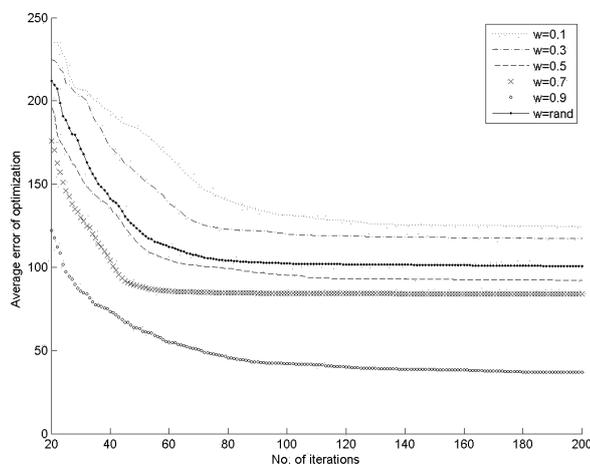


Fig. 4. Convergence of KHA with varying w for the benchmark function K4

benchmark instances (K2 in particular) - regardless of selected parameters set. This phenomenon is planned to be investigated in the next phase of our research.

4 Conclusions

The paper examined selected aspects of a new bio-inspired metaheuristic called the Krill Herd Algorithm. This procedure was applied herein for optimization tasks. As practical examples, a set of benchmark functions that were obtained from the CEC13 competition was utilized. The study considered two types of tests. The first one dwelt with choosing the parameter N^{max} while the second one concerned the determining of the influence of the parameter w on the quality of the obtained results, as well as on the speed of convergence of the optimization algorithm. In all comparisons, both means with standard deviations and best obtained error values were reported. During the study, two cases of the nature of the cost functions have been distinguished: unimodal and multimodal. On the basis of the carried-out tests, some clear results were obtained. We found that for optimization tasks with unimodal cost functions, the smaller value of the parameter N^{max} and w is highly recommended. Otherwise, for describing the multimodal character of cost function, we recommend higher values of these parameters or the application of the decreasing parameter w . This also results in a faster convergence of the optimization algorithm.

Finally, it should be noted that the performed experiments do not include a full analysis of the KHA parameters. The obtained results may, however, provide a starting point for further research and practical applications. In the near future, it is planned to publish research containing a full empirical analysis of the KHA procedure [10].

References

1. Ashlock, D.: *Evolutionary Computation for Modeling and Optimization*, Springer (2006)
2. Castro, L. N., Timmis, J.: *Artificial Immune Systems: A New Computational Intelligence Approach*, Springer (2002)
3. Dorigo, M., Birattari, M., Stutzle, T.: Ant colony optimization. *IEEE Computational Intelligence Magazine* 1, 28-39 (2006)
4. Engelbrecht, A.: *Fundamentals of Computational Swarm Intelligence*. John Wiley & Sons (2005)
5. Fletcher, R.: *Practical Methods of Optimization*. John Wiley & Sons (2000)
6. Gandomi, A.H., Alavi, A.H.: Krill herd: A new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation* 17, 4831-4845 (2012)
7. Hofmann, E.E, Haskell, A.G.E, Klinck, J., Lascara M.: Lagrangian modelling studies of Antarctic krill (*Euphasia superba*) swarm formation. *ICES J Mar Sci*, 61,617-631 (2004)
8. Karaboga, D., Basturk, B.: On the performance of artificial bee colony (ABC) algorithm. *Applied Soft Computing* 8, 687-697 (2008)
9. Kowalski, P.A.: Evolutionary Strategy for the Fuzzy Flip-Flop Neural Networks Supervised Learning Procedure, in: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (Eds.), *Artificial Intelligence and Soft Computing*, Lecture Notes in Computer Science vol 7894. Springer Berlin Heidelberg, pp. 294-305, (2013)
10. Kowalski, P.A, Lukasiak S.: "Properties and experimental evaluation of Krill Head Algorithm", (2014), in preparation
11. Liang, J., Qu, B., Suganthan, P., Hernandez-Diaz, A.: Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session and Competition on Real-Parameter Optimization, technical Report 201212, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore (2013)
12. Lukasiak, S., Żak, S.: Firefly Algorithm for Continuous Constrained Optimization Tasks, in: Nguyen, N.T., Kowalczyk, R., Chen, S.M. (Eds.), *Computational Collective Intelligence. Semantic Web, Social Networks and Multiagent Systems*, Lecture Notes in Computer Science vol. 5796. Springer Berlin Heidelberg, pp. 97-106 (2009)
13. Lukasiak, S., Kowalski P.A.: Fully Informed Swarm Optimization Algorithms: Basic Concepts, Variants and Experimental Evaluation, in: *2014 Federated Conference on Computer Science and Information Systems (FedCSIS)* (2014) in print
14. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. *Swarm Intell* 1, 33-57 (2007)
15. Rao, S.: *Engineering Optimization: Theory and Practice*. John Wiley & Sons (2007)
16. Wang, G.G., Gandomi, A.H., Alavi, A.H., Hao, G.S.: Hybrid krill herd algorithm with differential evolution for global numerical optimization. *Neural Comput & Applic* 1-12 (2013)
17. Wang, G., Guo, L., Wang, H., Duan, H., Liu, L., Li, J.: Incorporating mutation scheme into krill herd algorithm for global numerical optimization. *Neural Comput & Applic* 24, 853-871 (2014)
18. Wang, G.G., Gandomi, A.H., Alavi, A.H.: Stud krill herd algorithm. *Neurocomputing* 128, 363-370 (2014)