

Nature-Inspired Optimization Method : Hydrozoan Algorithm for Solving Continuous Problems

Daranat Tansui

Computational Intelligence Laboratory

Faculty of Information Technology

King Mongkut's Institute of Technology Ladkrabang

Bangkok, 10520 Thailand

56606010@kmitl.ac.th

Arit Thammano

Computational Intelligence Laboratory

Faculty of Information Technology

King Mongkut's Institute of Technology Ladkrabang

Bangkok, 10520 Thailand

arit@it.kmitl.ac.th

Abstract— In this article, a new optimization algorithm that is inspired by the biology of hydrozoa (HA) is proposed. Our aim was to develop an algorithm that is based on the regeneration and transplantation processes of hydrozoa for finding the best solutions for continuous optimization problems. Basically, HA follows the same general processes of evolutionary algorithm; however, its distinctive processes mimic the life cycle of 3 basic forms of hydrozoa: motile planula, polyps, and medusa. In particular, the growth of strong buds from the polyp stage depends on levels of morphogens: activators and inhibitors. These 3 forms develop or evolve into the best solution. HA was performance tested with 20 standard benchmark functions and compared with genetic algorithm and Particle Swarm Optimization (PSO). The test results have confirmed that the proposed algorithm is computationally more efficient than both GA and PSO. It works very well on most benchmark functions.

Keywords: *nature-inspired optimization method, hydrozoan life cycle, continuous optimization problems.*

I. INTRODUCTION

In general, optimization is widely used to increase performance and efficiency or cut down cost—optimization algorithm uses mathematical method to study how to search for the optimal solution for problems that may have numerous solutions. In other words, it is a study of effective planning and solving problems by using mathematical tools. There are 2 types of optimization methods. The first is conventional optimization algorithms which find the exact solution by some mathematical analytical method while the second is approximation optimization algorithms which are more effective for big and complex problems that the methods in the first group may not even be able to solve at all or may be able so but use unsatisfactorily long time to compute [1]. Nowadays, approximation optimization methods are widely used, especially the ones that work by mimicking some natural processes. Most new algorithms are nature-inspired by many biological foundations.

Nature-Inspired algorithms have been solving difficult survival problems for a very long time, and many plants and animals have evolved with surprising efficiency in optimizing their evolutionary purposes such the purpose of reproduction. Based on the survival success of some plants and animals,

several nature-inspired algorithms have been created over the last few years [2]. For Example, genetic algorithm (GA) is a search method based on Darwinian evolution and natural selection of plants and animals [3]. Invasive weed optimization (IWO) was based on seed distribution of weeds [4] while ABC algorithm was inspired by the a few smart behaviors of honey bees. Another algorithm of this kind, a flower pollination algorithm (FPA) was based on the development processes of flowering plants, while Particle Swarm Optimization algorithm (PSO) was based on swarming behaviors of bird or fish [5]. These nature-inspired algorithms have been found to be promising for solving real world problems and thus became popular and widely used including for solving multiobjective optimization problems.

In this paper, we propose a nature-inspired optimization method that is based on hydrozoan life cycle (HA), specifically, its processes of regeneration and transplantation—formation of new animals that are genetically identical to the parent organism (asexual) and that are different from their parents (sexual) [6]. Nature-inspired optimization methods are search algorithm for finding the best solution to various problems. It is used as a means for finding the global optimum. This algorithm was performance tested with 20 standard benchmark functions and compared with genetic algorithm (GA), and Particle Swarm Optimization (PSO).

This paper is organized as follows: Section 2, introduction, describes our inspiration and the biological foundation; Section 3 is a detailed description of the proposed algorithm; Section 4 describes the chosen benchmark functions and the 3 comparing algorithms as well as experimental results and discussion; and Section 5 concludes the paper.

II. INSPIRATION AND BIOLOGICAL FOUNDATION

A. Inspiration

The proposed HA algorithm is inspired by the processes of regeneration and transplantation in hydrozoan life cycle. Hydrozoa are a group of cnidarians that are known for their complexity and diversity in life cycles. Colonial hydrozoans have great phenotypic plasticity and regenerative capabilities.

Hydrozoan diversity came from evolution of developmental processes at evolutionary time scale. Hydrozoan life cycle can

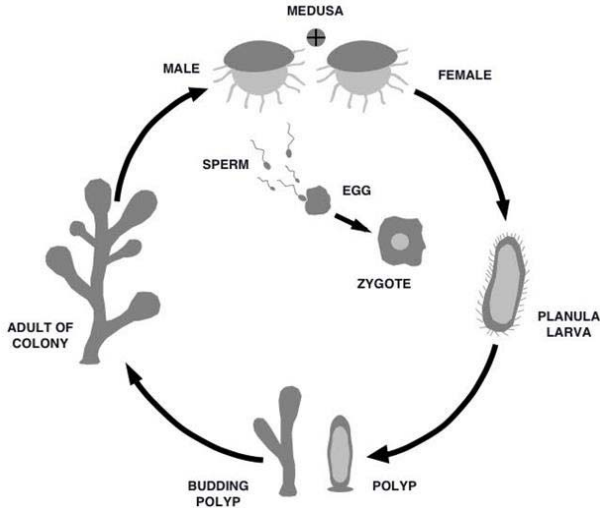


Fig. 1. Hydrozoan Life Cycle

include sexual and asexual reproduction. After a mating process, growth pulsations then started to spread over hydranth anlage area and cell mass area. A typical hydrozoan life cycle comprises three basic forms: motile planula (larva), polyps, which typically propagate to form extensive colonies by asexual reproduction, and pelagic medusa, a sexual form usually generated by strong budding from an adult polyp [7] [8].

In this paper, we were inspired by three main stages of hydrozoan colonies in our effort to make effective search optimization processes: the planula stage is about crossing over to produce next generation of chromosomes and random mutation of chromosomes in new generation, the polyps stage is about feeding and budding that require organizer regions for maintaining regular body patterning in continuous tissue dynamics during asexual reproduction and growth. The growth of strong buds from the polyp stage depends on levels of morphogens: activators and inhibitors. These morphogens interact with each cell and control the increase and decrease of growth value [9]. The growth values of all chromosomes are ranked for selection and inclusion into the next generation. The final medusa stage is gradual mutation then selection by roulette wheel method to obtain the parent chromosomes for the next generation. The proposed algorithm mimics the process of hydrozoan reproduction in order to develop or "evolve" solutions into the best solution.

III. OPTIMIZATION METHOD : HYDROZOAN ALGORITHM (HA)

In this section, we describe HA. This algorithm was inspired by the regeneration and transplantation processes of hydrozoa life cycle. Regeneration and transplantation consisted of cloning, splitting, and mutation. Cloning transfers the good fitness of better parents to offspring, splitting promotes generation of more offspring with better fitness, and mutation

results in rich diversity of offspring or solutions. In order to help produce a new and better solution, each hydrozoan uses its

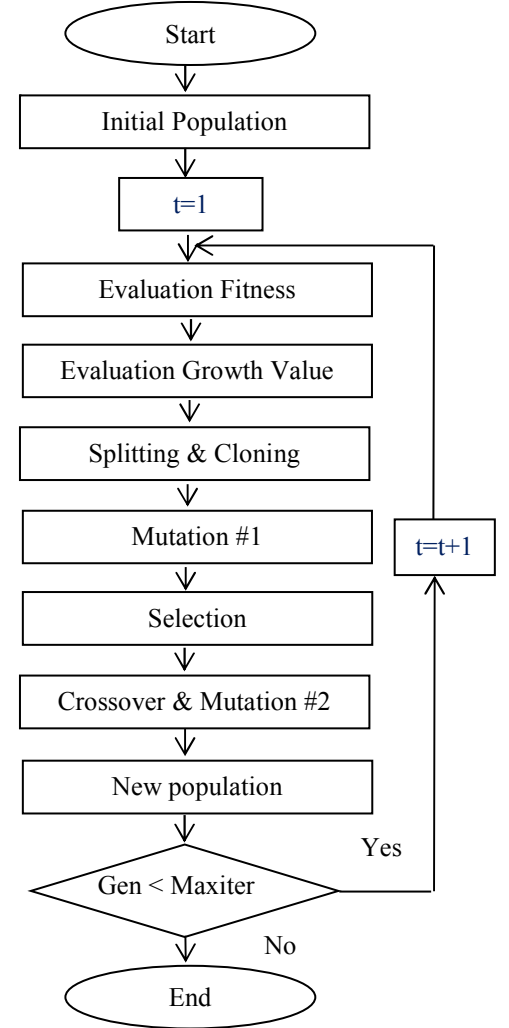


Fig. 2. Flowchart of hydrozoan algorithm

personal-best solution achieved in a previous iteration so far and the global best solution of the whole population.

The flowchart of the hydrozoan algorithm (HA) is displayed in Fig. 2. HA minimizes a fitness function in order to find the optimum solution. HA is an iterative method and its termination condition can be specified as a maximum evolution number or a maximum running time. The algorithm works by performing the following steps:

Step 1: Specify an initial population of N hydrozoans, and randomly generate the initial position of every hydrozoan, following the equations (1) and (2).

$$h_i(0) = [h_1^i, h_2^i, \dots, h_d^i, \dots, h_D^i] \quad (1)$$

$$h_d^i(0) \in [lower, upper] \quad (2)$$

where $i = 1$ to N , within a D -dimensional continuous search space and ‘lower’ and ‘upper’ are the lower and upper bounds of the D -dimensional search space.

Step 2: Input the initial position of each hydrozoan into the objective function and evaluate it to obtain its fitness value for that position.

Step 3: In the local morphodynamic process, the growth value of each hydrozoan ($G_i(t)$) is derived from the activator value (Ac_i^β) and the inhibitor value (In_i^α) according to equations (3), (4), and (5):

$$G_i \propto Ac_i^\beta \quad (3)$$

$$G_i \propto \frac{1}{In_i^\alpha} \quad (4)$$

$$G_i(t) = \frac{Ac_i^\beta}{In_i^\alpha} \quad (5)$$

where $G_i(t)$ is the growth value of hydrozoan i at time t , Ac_i^β is the activator value of hydrozoan i at time t , β is a significance indicator of the activator, In_i^α is the inhibitor value of hydrozoan i at time t , and α is a significance factor of the inhibitor.

Step 4: input the growth value $G_i(t)$ of every hydrozoan, as expressed in equation (6), into equation (7),

$$G_i(t) = [G_1^i, G_2^i, \dots, G_D^i] \quad (6)$$

where $i = 1$ to N , $G_i(t)$ is the growth value of hydrozoan i .

Step 5: Calculate the median value which represents the growth value of the Swarm _{i} . In order to calculate the median, the growth values must first be ranked (sorted in ascending order). The median is the middle number of the ranked values. More accurately, when a set of ordered values has an odd number of values, the median is the middle number of the ordered values, but when a set has an even number of values, the median is the average of the two middle numbers of the ordered values.

Step 6: The growth value of a hydrozoan in a particular swarm is calculated by equation (7). Hydrozoans are split into swarms according to the $Split_i$ condition (equation 8).

$$Swarm_i = G_i - Median \quad (7)$$

$$Split_i = \begin{cases} =0, & \text{if } Swarm_i < 0, \\ =1, & \text{if } Swarm_i = 0, \\ =1, & \text{if } Swarm_i > 0 \& Swarm_i = \min \\ =3, & \text{if } Swarm_i > 0 \& Swarm_i = \max \\ =2, & \text{otherwise} \end{cases} \quad (8)$$

Step 7: Choose and clone the hydrozoans according to the $split_i$ value.

Step 8: Mutate the clonal hydrozoan. Mutation is defined as a small random adjustment of gene. A gene h_d^i is selected and mutated by a low probability of mutation as follows:

$$RP = (\max - \min) * \phi + \min \quad (9)$$

$$h_d^i = h_d^i + (h_d^i \times RP) \quad (10)$$

where RP is a constant derived from ‘max’ and ‘min’ which are the lower and upper bounds of mutation rate specified as percentages by user, ϕ is a random number within the range $[0, 1]$.

Step 9: Select new parents from hydrozoans that has already been mutated by using roulette wheel selection. P_j is the probability that an individual hydrozoan i is selected, which is computed as follows:

$$P_j = f_{h_i} / \sum_{q=1}^N f_{h_q} \quad (11)$$

where f_{h_j} is the fitness value of hydrozoan i at time t .

Step 10: Cross over the genes of the selected parents with an N - point crossover operator in order to obtain a variety of better offspring for the next generation. Random crossover points are selected and the tails of the two parents are swapped between these points to get new off-springs, as shown in the following example:

	1			2		3	
P1	0.85	-2.44	1.83	1.45	4.15	-2.50	-4.20
P2	-0.05	1.10	2.54	4.30	1.05	2.70	3.20
O1	0.85	-2.44	2.54	4.30	4.15	-2.50	3.20
O2	-0.05	1.10	1.83	1.45	1.05	2.70	-4.20

Step 11: Then, a mutation operator operates on the offspring. The index of a gene to be mutated is found by randomization then replace the gene with a random value in the range between the upper and lower bounds. These processes are expressed by the equations below:

$$\begin{aligned} h_i(0) &= [h_1^i, h_2^i, \dots, h_d^i, \dots, h_D^i], \\ A_i &= h_d | h_d \in R, h_d \in [lower, upper], \\ A_i &\xrightarrow{\text{replacement}} h_d^i \end{aligned} \quad (12)$$

where $i = 1$ to N , within a D -dimensional continuous search space. This operator is applied to all offspring.

Step 12: Evaluate the fitness values of every offspring, Combine and sort all hydrozoans according to their fitness value, select a specified number of the best hydrozoans, and update the best hydrozoan with the highest fitness value I the best for that iteration.

$$I_{thebest} = \arg \min_i (f_{h_i(t)}) \quad (13)$$

Step 13: Check the stopping criterion. If the maximum number of iterations is reached, the algorithm will then be terminated.

IV. RESULTS AND DISCUSSION

The performance of the proposed model was tested with twenty standard benchmark functions [10] shown in Table I and II.

The performance of our proposed algorithm was compared with those of traditional genetic algorithm (GA) and Particle swarm optimization (PSO). They are briefly described below.

Table III and IV show the comparative performances of these 3 algorithms. For each benchmark function, The maximum cycle number used for termination of the algorithms was 10000 and the algorithms were run 20 times. Each run started with a different initial population and stopped when the maximum number of iterations was reached. The performances of the three algorithms were measured in terms of nearness to the true global minimum.

The experimental results for the proposed algorithm on two benchmark functions—Cross-in-tray (f_1) and Holder table (f_3)—in two dimensions show that it was able to find the global optimal solutions. For Booth (f_9), Griewank (f_{13}), Rasstrigin (f_{14}), Rosenbrock (f_{15}), and sum of different powers (f_{17}), step (f_{18}), Himmelblau (f_{19}), and Xin-SheYang (f_{20}) in 5 dimensions, it was able to converge nearer to the global optimal solutions than the other two algorithms.

GA was able to find the global optimal solutions for Cross-in-tray (f_1), Holder table (f_3), Sphere (f_{11}), and was able to converge nearer to the optimal solutions for Levy N.2 (f_4), Matyas (f_6), Camel (f_{10}), and Ackley (f_{12}) than the other two algorithms. However, GA had a tendency to converge to local minima rather than the global minimum of the problem for Schaffer N.2 (f_5), Griewank (f_{13}), and Rastrigin (f_{14}).

TABLE I. DETAILS OF TWO- DIMENSION FUNCTIONS

Function	Name of Function	Formula	Search domain	Global minimum
f_1	Cross-in-Tray	$f(x) = -0.0001 \left(\left \sin(x_1) \sin(x_2) \exp \left(\left 100 - \frac{\sqrt{x_1^2 + x_2^2}}{\pi} \right \right) \right + 1 \right)^{0.1}$	$-10 \leq X_i \leq 10$	-2.06261
f_2	Drop-Wave	$f(x) = -\frac{1 + \cos(12\sqrt{x_1^2 + x_2^2})}{0.5(x_1^2 + x_2^2 + 2)}$	$-5.12 \leq X_i \leq 5.12$	-1
f_3	Holder Table	$f(x) = -\left \sin(x_1) \cos(x_2) \exp \left(\left 1 - \frac{\sqrt{x_1^2 + x_2^2}}{\pi} \right \right) \right $	$-10 \leq X_i \leq 10$	-19.2085
f_4	Levy N.13	$f(x) = \sin^2(3\pi x_1) + (x_1 - 1)^2 [1 + \sin^2(3\pi x_2)] + (x_2 - 1)^2 [1 + \sin^2(2\pi x_2)]$	$-10 \leq X_i \leq 10$	0
f_5	Schaffer N.2	$f(x) = 0.5 + \frac{\sin^2(x_1^2 - x_2^2) - 0.5}{[1 + 0.001(x_1^2 + x_2^2)]^2}$	$-100 \leq X_i \leq 100$	0
f_6	Matyas	$f(x) = 0.26(x_0^2 + x_1^2) - 0.48x_0x_1$	$-10 \leq X_i \leq 10$	0
f_7	Bohachevsky	$f(x) = x_1^2 + 2x_2^2 - 0.3 \cos(3\pi x_1) \cos(4\pi x_2) + 0.3$	$-100 \leq X_i \leq 100$	0
f_8	Beale	$f(x) = (1.5 - x_1 + x_1x_2)^2 + (2.25 - x_1 + x_1x_2^2)^2 + (2.625 - x_1 + x_1x_2^3)^2$	$-4.5 \leq X_i \leq 4.5$	0
f_9	Booth	$f(x) = (x_0 + 2x_1 - 7)^2 + (2x_0 + x_1 - 5)^2$	$-10 \leq X_i \leq 10$	0
f_{10}	Camel	$f(x) = 2x_0^2 - 1.05x_0^4 + \frac{1}{6}x_0^6 + x_0x_1x_1^2$	$-5 \leq X_i \leq 5$	0

TABLE II. DETAILS OF N- DIMENSIONAL FUNCTIONS

Function	Name of Function	Formula	Search domain	Global minimum
f_{11}	Sphere	$f(x) = \sum_{i=1}^n x_i^2$	$-5.12 \leq X_i \leq 5.12$	0
f_{12}	Ackley	$f(x) = -20 \exp \left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2} \right) - \exp \left(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i) + a + \exp(1) \right)$	$-32.768 \leq X_i \leq 32.768$	0
f_{13}	Griewank	$f(x) = \sum_{i=1}^n \frac{x_i^2}{4000} - \prod_{i=1}^n \cos \left(\frac{x_i}{\sqrt{i}} \right) + 1$	$-600 \leq X_i \leq 600$	0
f_{14}	Rastrigin	$f(x) = 10n + \sum_{i=1}^n [x_i^2 - 10 \cos(2\pi x_i)]$	$-5.12 \leq X_i \leq 5.12$	0
f_{15}	Rosenbrock	$f(x) = \sum_{i=1}^{n-1} [(x_i - 1)^2 + 100(x_{i+1} - x_i^2)^2]$	$-5 \leq X_i \leq 5$	0
f_{16}	Levy	$f(x) = \sin^2(\pi w_1) + \sum_{i=1}^{n-1} (w_i - 1)^2 [1 + 10 \sin^2(\pi w_i + 1)] + (w_n - 1)^2 [1 + \sin^2(2\pi w_n)]$, where $w_i = 1 + \frac{x_i - 1}{4}$, for all $i = 1, \dots, n$	$-10 \leq X_i \leq 10$	0
f_{17}	Sum of Different Powers	$f(x) = \sum_{i=1}^n x_i ^{i+1}$	$-1 \leq X_i \leq 1$	0
f_{18}	Step	$f(x) = \sum_{i=1}^{n-1} (\lfloor x_i \rfloor + 0.5^2)$	$-100 \leq X_i \leq 100$	0
f_{19}	Himmelblan	$f(x) = \frac{1}{n} \sum_{i=1}^n x_i^4 - 16x_i^2 + 5x_i$	$-5 \leq X_i \leq 5$	0
f_{20}	Xin-She Yang	$f(x) = - \left(\sum_{i=1}^n x_i \right) \exp \left(- \sum_{i=1}^n x_i^2 \right)$	$-2\pi \leq X_i \leq 2\pi$	0

TABLE III. COMPARATIVE RESULTS ON TWO-DIMENSION FUNCTIONS

Function	The original model		The proposed model
	GA	PSO	HA
f_1	-2.06261	-2.06261	-2.06261
f_2	-0.93625	-1	-0.99972
f_3	-19.20805	-19.1970	-19.2085
f_4	3.84E-07	2.59E-06	1.46E-06
f_5	0.001727	2.14E-12	9.07E-08
f_6	4.66E-15	0.004563	2.66E-07
f_7	0.218313	0.227555	0.00055
f_8	2.53E-05	0.044332	2.71E-07
f_9	1.39E-05	0.494507	1.22E-05
f_{10}	5.16E-26	3.29E-15	1.07E-21

TABLE IV. COMPARATIVE RESULTS ON N-DIMENSION FUNCTIONS

Function	The original model		The proposed model
	GA	PSO	HA
f_{11}	0	0.001000	4.89E-09
f_{12}	1.51E-14	0.297168	0.005900
f_{13}	0.064900	0.168234	0.002550
f_{14}	0.502533	0.674054	2.47E-05
f_{15}	0.033172	0.873357	0.030470
f_{16}	3.12E-08	0.037943	2.72E-05
f_{17}	8.58E-16	2.3E-05	4.77E-18
f_{18}	0.250000	1.824572	0.000130
f_{19}	1.28E-07	1.57E-05	9.31E-08
f_{20}	0.041782	0.043962	0.041780

As can be seen in Table III and IV, PSO was able to find the global minimum for Cross-in-tray (f_1), Drop-Wave (f_2), and Schaffer N.2 (f_3), in two dimensions, it was able to converge nearer to the global optimal solutions than the other two algorithm, but for $f_4, f_6, f_7, f_8, f_9, f_{10}, f_{11}, f_{12}, f_{13}, f_{14}, f_{15}$,

$f_{16}, f_{17}, f_{18}, f_{19}$, and f_{20} , it was not able to converge nearer to the global minimum than the other two algorithms.

The reasons why the proposed algorithm was more successful were because the cloning feature retained the solution with the best fitness value, the splitting feature increased the number of good solutions, and mutation at a specified rate created several hydrozoans with better fitness values. Moreover, selection and crossover, and mutation resulted in a rich diversity of more effective offspring or solutions for the new generation.

V. CONCLUSION

In this paper, we propose a novel nature-inspired algorithm named hydrozoan algorithm (HA) that mimics the regeneration and transplantation processes of hydrozoan life cycle. HA is a very simple and efficient search technique. It is a meta-heuristic optimization method that is population-based.

The performances of the proposed algorithm were examined with several benchmark functions and compared to two other algorithms: GA and PSO. The test results show that the proposed algorithm performed very well on these functions with more success at converging to global optima. Our future work will attempt to make it move toward better and better areas in the solution search space and boost the efficiency of this algorithm by having it automatically determine the optimal control parameters.

- [1] E.Nabil, "A Modified Flower Pollination Algorithm for Global Optimization," *Expert Systems with Application*, 57,192-203, 2016.
- [2] I. Fister Jr, X.S. Yang, I. Fister, J. Brest and D Fister, "A brief review of nature-inspired algorithms for optimization," *elektrotehniski vestnik* 80(3): 1–7, 2013.
- [3] R. Kumar, and Jyotishree, "Blending Roulette Wheel Selection & Rank Selection in," *International Journal of Machine Learning and Computing*, Vol. 2, No. 4, August 2012.
- [4] D.Kundu, K. Suresh, S. Ghosh, S. Das, B.K.Panigrahi and S. Das, "Multi-objective optimization with artificial weed colonies," *Information Sciences*, vol 181, pp. 2441–2454, 2011.
- [5] O.Hegazy, O.S.Soliman and M. A. Salam, "Comparative Study between FPA, BA, MCS, ABC, and PSO Algorithms in Training and Optimizing of LS-SVM for Stock Market Prediction," *International Journal of Advanced Computer Research*. ISSN (Print): 2249-7277 ISSN (Online): 2277-7970 Volume-5 Issue, 2015.
- [6] L.Leclerc, R.R Copley, T. Momose and E.Houliston, "Hydrozoan insights in animal development and evolution," *Current Opinion in Genetics & Development*, 39:157–167, 2016.
- [7] P.Cartwright and A. M. Nawrocki, "Character Evolution in Hydrozoa (phylum Cnidaria)," *Integrative and Comparative Biology*, volume 50, number 3, pp. 456–472, 2010.
- [8] S. Berking, "A model for budding in hydra: pattern formation in concentric rings," *Journal of Theoretical Biology* 222, 37–52, 2003.
- [9] S. Berking, "Principles of branch formation and branch patterning in Hydrozoa," *Zoological Institute, University of Cologne, Köln, Germany Int. J. Dev. Biol.* 50: 123-134, 2006.
- [10] Ernesto P.Adorio, "MVF - Multivariate Test Functions Library in C for Unconstrained Global Optimization," Department of Mathematics, Revised January 14, 2005.