Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 6 (2012) 126 – 135

Procedia Technology

2nd International Conference on Communication, Computing & Security (ICCCS-2012)

The Lion's Algorithm: A New Nature-Inspired Search Algorithm

B.R.Rajakumar*

Project Lead- Research, Griantek, Bengaluru, India

Abstract

Natural Computing is an efficient computing field that intends to develop search, optimization and machine learning algorithms by the inspiration of nature's behavior in problem solving circumstances. In the same way, numerous biologically or naturally inspired search and optimization algorithms have been proposed in the literature. This paper proposes a novel solution search algorithm called lion's algorithm. The natural inspiration behind the proposed algorithm is lion's social behavior that aids to keep the mammal be strong in the world. The interpretation of such social behavior to algorithmic perspective helps in searching out highly optimal solutions from a huge solution space. The algorithm solves both single variable and multi-variable cost function problems through the generation of binary structured and integer structured lion, respectively. The algorithm is implemented and tested using De-Jong's Type I function and the results are compared against the evolutionary programming. The test results show the algorithm performance under varying sizes of solution spaces.

Keywords: Lion's Algorithm; Territorial defense; Territorial Takeover; lion; pride; mating; nomadic lion.

1. Introduction

Nature-inspired computing or Natural Computing is a field of research (Rozenberg *et al*., 2011) intends to solve complex problems (Liu and Tusi, 2006) with uncertainty, partially true, imprecise and high conflicts, only based on natural inspiration (Zhang, 2009). Bio-inspired computing, a subset of nature-inspired computing (Shadbolt, 2004) starts emerging from the day of nature's behavior gets be executed for solving real life problems (De Souza and Costa, 2009).

* Corresponding author. *E-mail address:* rajakumar@ieee.org

doi:10.1016/j.protcy.2012.10.016

Nomenclature

x_i	i^{th} solution variable
$x_i^{\min}$	minimum limit of solution space of i^{th} solution variable
$x_i^{\max}$	maximum limits of solution space of i^{th} solution variable
X^{male}	male territorial lion
X^{female}	female territorial lion
L	length of the solution vector
X^{cub}	cubs of X^{male} and X^{female}
X^{m_cubs}	male group of cubs
X^{f_cubs}	male group of cubs
X^{nomad}	male group of cubs

The bio-inspired optimization algorithms have been effectively developed based on natural inspiration since 1970 (Rechenberg, 1973) (Holland, 1975) and find applications in almost all the emerging fields (Bongard, 2009) (Forbes, 2000). The bio-inspired optimization algorithms can be broadly categorized into two namely, Evolutionary algorithms (David, 1989) (Storn, 1996) and swarm intelligence. In this paper, a new search algorithm, called as Lion's algorithm, is proposed, based on the lion's social behavior. Lion's unique social behavior is being as one of the factors of exposing it as the strongest mammal in the world. Based on the behavior, the Lion's algorithm is formulated to search for optimal solutions in a huge search space.

2. Lion's Social Behavior

Lions have an interesting social behavior to keep the animal stronger in every generation, unlike other cat species. Pride is the term used to refer Lion's social system, in which resident females are attending males and give birth to offspring, sharing an area called as territory with peaceful interactions. A cub needs 2-4 years to attain sexual maturity and so the territorial lion needs to defend for the territory for the same number of years. In between these 2-4 years, nomadic lions may try to invade the pride, which we call it as territorial defense. In the territorial defense, a war is held between the territorial lions and nomadic lions. Coalition is built among the lions that belong to the pride to defeat the nomadic lion. If the nomadic lion defeats the territorial lion, the territorial lion may be either killed or driven out of the pride by the nomadic lion. The nomadic lion becomes the territorial lion by killing the cubs of lost lion. The new territorial lion can immediately force the female lion to estrus and copulate for its offspring. Once the cubs of a pride reach sexual maturity and if they seem to be stronger than the territorial lion to take over the pride, the territorial lion may be either killed or driven out of the pride. The new stronger pride lion kills the cubs of the territorial laggard lion and prepare for copulation to give birth to their own cubs (Bauer *et al*, 2003).

3. The Lion's Algorithm

The Lion's Algorithm searches optimal solution based on two unique lion's behavior, namely territorial defense and territorial takeover. The territorial defense is carried out between the resident males and nomadic males whereas the territorial takeover is carried out between the old territorial male and new territorial male.

Definition 1: Lion is a solution to be determined whereas the Cubs are solutions that are derived from existing solutions.

Definition 2: Territorial Defense is a process of evaluating the existing solution (territorial lion) and newly generated solution (nomadic lion), replacing existing solution by new solution if new solution is better than existing solution and vanishes the derived solutions of old solution.

Definition 3: Territorial Takeover is a process of keeping only derived best male and female solutions, which are competent over new solution for certain extent, and vanishing existing solutions in the pride.

In most of the biologically inspired optimization algorithms, an equivalent process for territorial takeover, mostly called as selection operation, is performed to keep better solution and to vanish undesired solution, but the process of territorial defense is new to this field. The lack of such equivalent process for territorial defense in other optimization algorithms has made the lion's algorithm better than other optimization algorithms, in fact in reality, these behaviors made lion as the strongest animal in the world in a terse evolution.

3.1. Basic Structure

A basic working model of the proposed Lion's algorithm is given in Fig. 1.

```
Step 1: Pride Generation
        Step 1.1: Generate territorial male and female subject to solution constraints
Step 2: Mating
        Step 2.1: Crossover
        Step 2.2: Mutation
        Step 2.3: Gender grouping
        Step 2.4: Kill sick/ weak cubs
        Step 2.5: Update Pride
Step 3: Territorial Defense
        Step 3.1: Keep the record of cubs' age
        Step 3.2: Do
                Step 3.2.1: Generate and Trespass
                        Nomadic lion
                Until stronger Nomadic lion trespasses
        Until Cubs get matured
Step 4: Territorial Takeover
        Step 4.1: Selection of best lion and lionesses
      Step 4.2: Go to Step 2 until termination criteria is met
```

Fig. 1. Basic Structure of the proposed Lion's Algorithm

The basic structure of Lion's Algorithm can be generally grouped into four major components based on the nature of its functions. They are, (i) Pride Generation, which is responsible for generating solutions, (ii) Mating that refers to deriving new solutions and (iii) Territorial Defense and (iv) Territorial Takeover intend to find and replace worst solution by new best solution. The repeated process enables heuristic search to converge the solution nearer to the target/ desired solution.

Definition 4: Pride is a dynamically varying solution pool with varying size, initiated with two arbitrary solutions, one represents male and the other represents female, in which updates of derived solutions and vanishing of undesired solutions happen.

Definition 5: Mating is a process of deriving new best solutions from the existing solutions that includes crossover and mutation for deriving new solutions, gender grouping to find diversification among the solutions and killing sick/ weak cubs ensures the derived solutions to be best.

Similar to GA, the Lion's algorithm can be both binary coded and integer coded so that single variable optimization problem can be solved using binary coded Lion's algorithm and multi-variable can be done by integer coded Lion's Algorithm.

3.2. The Search Procedure of Lion's Algorithm

The search procedure of Lion's Algorithm strictly follows the basic structure of the algorithm, which is given in Fig 1. It mainly focus on finding out the optimal solutions, which solves the objective i.e. minimize or maximize the objective function. Let us consider an objective function

$$\underset{x_i \in (x_i^{\min}, x_i^{\max})}{\arg\min} \quad f(x_1, x_2, \ldots x_n); \qquad n \geq 1 \tag{1}$$

The given function in Eq. (1) is a n -variable minimization function in which every solution variable, $x_i : i = 1,2,\ldots,n$, may be subjected to certain equality and inequality constraints. When $n = 1$, the lion has to be binary structured, whereas integer structured lion is preferred when $n > 1$. As per Fig 1, the search procedure is initiated by generating a pride. The initial pride has a X^{male} and X^{female}, who have the structure as $X^{male} = [x_1^{male} \; x_2^{male} \; \ldots \; x_L^{male}]$ and $X^{female} = [x_1^{female} \; x_2^{female} \; \ldots \; x_L^{female}]$ where, L defines length of the solution vector to be determined as

$$L = \begin{cases} n & ; n > 1 \\ m & ; otherwise \end{cases} \tag{2}$$

In Eq. (2), x_l^{male} and x_l^{female}, where $l = 1,2,\ldots,L$ are arbitrary integers to be generated within the intervals $(x_l^{\min}, x_l^{\max})$ when $n > 1$, whereas at $n = 1$, x_l^{male} and x_l^{female} may be either 0 or 1 such that $g(x_l) \in (x^{\min}, x^{\max})$. The $g(x_l)$ represents both $g(x_l^{male})$ and $g(x_l^{female})$ is defined as

$$g(x_l) = d(x_1) \sum_{l=2}^{L} 2^{L-l} \, x_l \tag{3}$$

where,

$$d(x_l) = \begin{cases} 1; \, if \, x_1 = 0 \\ -1; \, otherwise \end{cases} \tag{4}$$

The generated X^{male} and X^{female} undergo mating by performing crossover and mutation process. A mating process results in the production of new four cubs X^{cub} firstly by crossover and then by mutation, here called as mating operators. They are similar to that of genetic operators (of GA); however in Lion's Algorithm, dual probabilities based crossover is introduced i.e. crossover is performed with two different probabilities. The schematic view of dual probabilities based crossover is illustrated in Fig 2.

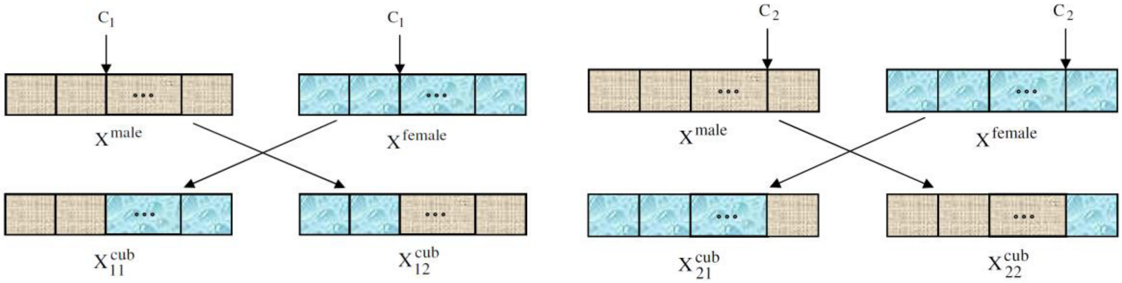


Fig. 2. Illustration of Single Point Crossover with dual probabilities used in Lion's Algorithm

Here, single point crossover operation and random mutation is enabled to generate X^{cub} from crossover and X^{new} from mutation. Once the crossover and mutation are performed, the cub pool is filled up with 4 direct cubs and 4 mutated cubs. Thus generated cub pool is subjected to gender grouping.

Definition 6: Gender grouping is a clustering process to cluster the given solution pool into two groups, one group is comprised of male cubs and the other is comprised of female cubs.

To perform gender grouping, K-means clustering (MacQueen, 1967) (Steinhaus, 1957) (Lloyd, 1957) is applied over the cub pool to generate such X^{m_cubs} and X^{f_cubs}. In order to update the pride, it is necessary to kill sick/ weak cubs and to maintain cub pool stability among male and female cubs. The cub pool stability is maintained by firstly testing the health of cub pools i.e. male and female cubs' health and then killing either needed number of laggard male cubs or laggard female cubs so that cub pool should have equal number of male and female cubs. It is to be noted that, testing the health is nothing but to determine the objective function value of every cub. Such stabilized cub pool is added along with the existing territorial lions and so the pride gets updated. Once the pride gets updated, the cubs' age has to be initialized to zero. At every success of territorial lion against nomadic lion in territorial defense, the cubs' age is incremented by one. The pseudo codes for territorial defense and territorial takeover operations are given in Fig 3 and 4.

Initialize age (cubs) ← 0
Do
  Generate X^{nomal} // similar to the generation of X^{male}
  if $f(X^{nomad}) < f(X^{male})$
    if $f(X^{nomad}) < f(X^{pride})$
      Kill X^{m_cubs} and X^{f_cubs}
      $X^{male} \leftarrow X^{nomad}$
        Go to
          Mating
    End if
  else
  Increment age (cubs) by 1
  End if
Until age (cubs) > Age_{mat}

Fig. 3. Pseudo code for illustration of Territorial Defense of Lion's Algorithm

In the pseudo code for territorial defense, X^{nomad} can be generated by following the similar procedure for generating X^{male}, $f(\bullet)$ is the objective function value for instance, $f(X^{male})$ and $f(X^{nomad})$ are the objective function values i.e. strength of X^{male} and X^{nomad}, respectively, whereas $f(X^{pride})$ is the strength of the entire pride that can be calculated as

$$f(X^{pride}) = \frac{1}{2(1+\|X^{m\_cubs}\|)}\left(f(X^{male}) + f(X^{female}) + \frac{Age_{mat}}{age(cub)+1} \sum_{C=1}^{\|X^{m\_cubs}\|} \frac{f(X_c^{m\_cubs}) + f(X_c^{f\_cubs})}{\|X^{m\_cubs}\|} \right) \quad (5)$$

where, $f(X_c^{m\_cubs})$ and $f(X_c^{f\_cubs})$ is the strength of male and female cubs respectively, $\|X^{m\_cubs}\|$ represents the number of male cubs in the pride and Age_{mat} is the maturity age for mating.

The strength comparison between X^{nomad} and X^{male}, and then between X^{nomad} and X^{pride} explicitly illustrates the coalition behavior of lions. This coalition (packer and pusey, 1982) (Grinnell *et al*., 1995) is mandatorily enabled in territorial defense in any lion's pride to define the strength of nomadic lion to handle a pride. The coalition behavior of the algorithm aids to introduce strong solution into the pride.

Definition 7: Coalition is a property of the algorithm that allows a new solution in to the process in such a way that the new solution should be better than the competency of existing solution and joint competency of the pride.

The territorial takeover is performed, once the cubs reach maturity level i.e. when cubs' age is greater than or equal to maturity age Age_{mat}. Once the cubs reached the level, they can be considered as lions and they start defending with the pride's old lion to prove its strength.

Initialize $gen = 0$
Select X_{best}^{male} and X_{best}^{female} : $X_{best}^{female} \neq X_{best}^{male}$
  if $X_{best}^{female} = X^{female}$
    Increment B_{count} by 1
  otherwise
    $B_{count} = 0$
  End if
  $X^{male} \leftarrow X_{best}^{male}$
  $X^{female} \leftarrow X_{best}^{female}$
  if $B_{count} > B_{strength}$
  Do
  Generate X_{new}^{female} : $X_{new}^{female} \neq X^{male}$
  Until $f(X_{new}^{female}) < f(X^{female})$
    $X^{female} \leftarrow X_{new}^{female}$
  End if
    $gen \leftarrow gen+1$
  Go to
    Mating
Until $gen > gen_{\max}$

Fig. 4. Pseudo code for illustration of Territorial Defense of Lion's Algorithm

In the first step of territorial takeover, construct X^{male}_{pride} and X^{female}_{pride} by appending the X^{male} and X^{m_cubs} in X^{male}_{pride} and X^{female} and X^{f_cubs} in X^{female}_{pride} . X^{male}_{best} and X^{female}_{best} is selected in such a way that they should follow the criteria

$$f(X^{male}_{best}) < f(X^{male}_{pride}(p)); X^{male}_{pride}(p) \neq X^{male}_{best} \tag{6.a}$$

$$f(X^{female}_{best}) < f(X^{female}_{pride}(p)); X^{female}_{pride}(p) \neq X^{female}_{best} \tag{6.b}$$

Once the X^{male}_{best} and X^{female}_{best} are selected, the mating strength of X^{female}_{best} is validated to decide whether to keep the X^{female}_{best} in pride or not. In the pseudo code, the B_{count} is the number of breeding by X^{female}_{best} and $B_{strength}$ represents female's maximum breeding strength, generally set to 5. The B_{count} has to be initialized at the time of initial pride generation and it has to be incremented, when the corresponding lioness undergoes mating with lion. If the old female territory lion is found to be laggard than the new female or cub, then the laggard is replaced by the new ones and again the B_{count} has to be started from zero to make the new ones into mating. On the other hand, if new lioness is found as laggard, B_{count} is updated and old female is put into mating until the $B_{strength}$ reaches maximum. This entire process is iteratively repeated by $gen_{\max}$ number of generations are obtained. Once the process reaches $gen_{\max}$, a best lion from the pride is selected as the optimal solution.

4. Experimental Results

The proposed Lion's Algorithm is implemented and tested in MATLAB 7.12 (R2011a) by evaluating the performance using a simplest benchmark function called as De-Jong's Type I function (Molga and Smutnicki, 2005), which is given below. The algorithmic parameters used in experimental validation are tabulated in Table 1.

$$f(x) = \sum_{i-1}^{n} x_i^2 \tag{7}$$

Table 1: Algorithmic parameters set for experimental validation of Integer structured and Binary structured Lion's Algorithm

Sl. No	Algorithmic Parameters	Integer Structured Lion's Algorithm	Binary Structured Lion's Algorithm
1	n	5	1
2	Age_{mat}	3	3
3	$B_{strength}$	5	5
4	gen_{max}	100	50
5	Crossover probabilities	[0.2 0.6]	[0.2 0.6]
6	Mutation probability	0.5	0.5

In order to validate the performance of the algorithm, the size of solution space is highly varied and the convergence graphs are plotted and compared against a general evolutionary programming. Fig 5 and 6 illustrates the convergence performance of binary structured and integer structured of Lion's Algorithm, respectively under various sizes of solution space over evolutionary programming.

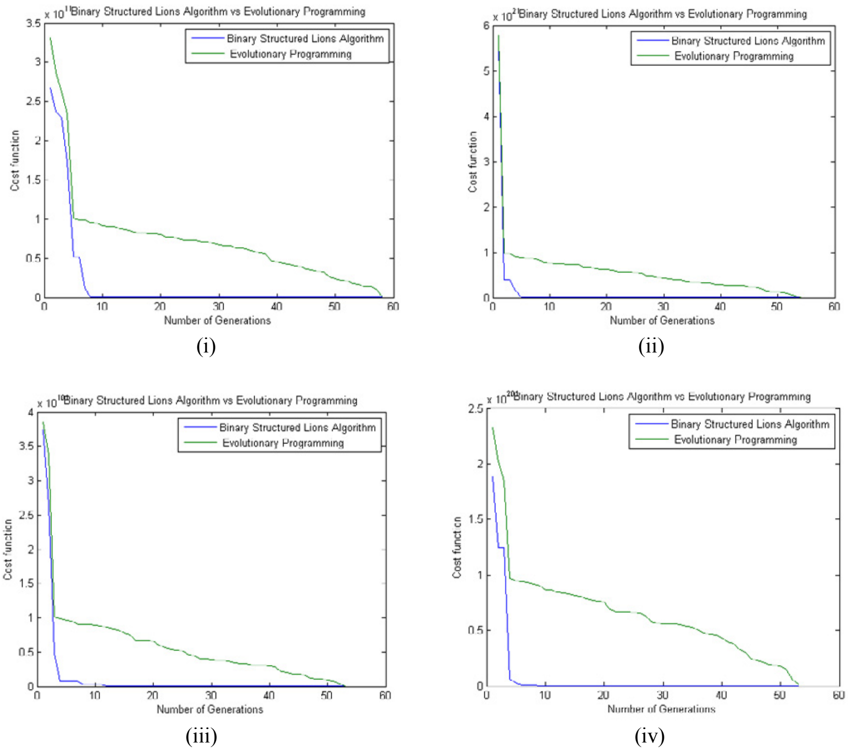
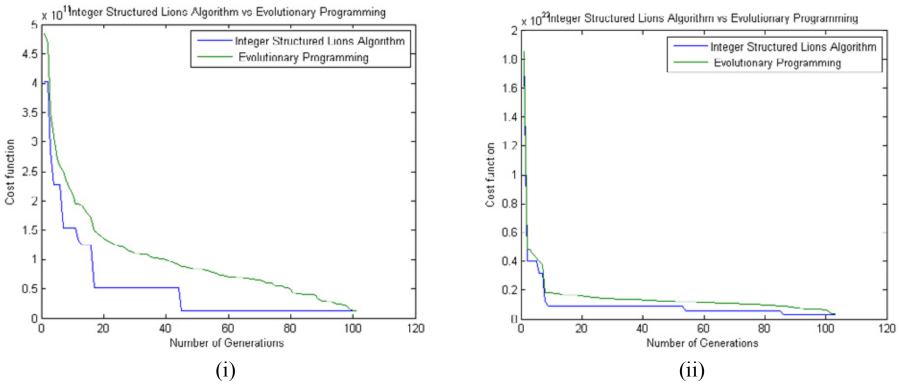
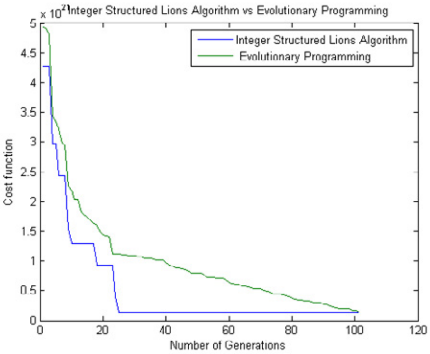
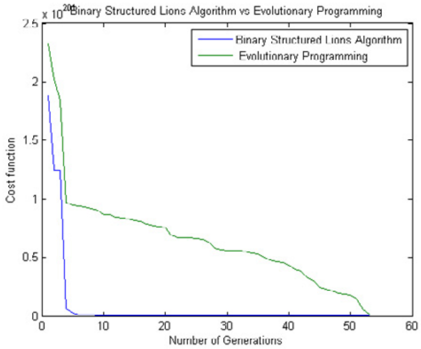


Fig. 5. Convergence of binary structured Lion's Algorithm vs. Evolutionary programming under various solution spaces, (x^{min}, x^{max}) are (i)($-10x10^5$, $10x10^5$), (ii) ($-10x10^{10}$, $10x10^{10}$), (iii) ($-10x10^{50}$, $10x10^{50}$) and (iv) ($-10x10^{100}$, $10x10^{100}$)





(iii)



(iv)

Fig. 6. Convergence graph of integer structured Lion's Algorithm vs. Evolutionary Programming under various solution spaces, (x^{min}, x^{max}) are (i)(-10x10^5, 10x10^5), (ii) (-10x10^{10}, 10x10^{10}), (iii) (-10x10^{50}, 10x10^{50}) and (iv) (-10x10^{100}, 10x10^{100})

5. Conclusion and Future Work

In this paper, a new search algorithm called as the Lion's Algorithm was proposed and experimentally compared with general evolutionary programming. The experiments were conducted under various solution space sizes at pre-defined algorithmic parameters. Under such an experimental environment, the algorithm was closely monitored for its performance, when executing at every solution space. From the analysis, it can be said that the algorithm maintained a stable and reliable performance over convergence of problem to the optimal solution, when compared to the evolutionary programming. In other words, the algorithm minimized the cost function i.e. found out the solution that minimizes the cost function in a consistent manner despite the size of the solution spaces. As the algorithm is inspired by the merits of lion's social behavior in strengthening their generation, the convergence is very less time consuming and reliable. As encouraging results are obtained, the Lion's algorithm will be tested for solving the unimodal, multimodal and real-time search problems and its performance will be studied.

Acknowledgements

The author is thankful to Dr. Aloysius George and the best colleagues of Griantek, for their useful feedbacks and discussions on the algorithm and for their motivation and support.

References

Bauer, H., de Iongh, H.H.,Silvestre, I., 2003. "Lion social behaviour in the West and Central African Savanna belt.Mammalian Biology," 68(1), P. 239-243.
Bongard, J., 2009. "Biologically Inspired Computing," IEEE Computer Journal 42(4), p. 95-98.
David, E.G., 1989. "Genetic Algorithms in Search Optimization and Machine Learning," Addison Wesley, p. 41.
Forbes, N., 2000. "Biologically inspired computing," Computing in Science & Engineering 2(6), p. 83-87.
Grinnell, J., Packer, C., Pusey, A.E., 1995. "Cooperation in male lions: kinship, reciprocity or mutualism?," Animal Behaviour 49(1), P.95–105.
Holland, J.H., 1975. "Adaptation in natural and artificial systems," University of Michigan Press, Ann Arbor.
Liu, J., Tsui, K.C., 2006. "Toward nature-inspired computing," Communications of the ACM 49(10).
Lloyd, S. P., 1957. "Least square quantization in PCM," IEEE Transactions on Information Theory 28 (2): 129–137.
MacQueen, J. B., 1967. "Some Methods for classification and Analysis of Multivariate Observations," 1. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press, P.281–297.
Molga, M., Smutnicki, C., 2005. "Test functions for optimization needs," available at http://www.zsd.ict.pwr.wroc.pl/files/docs/functions.pdf (accessed on 8 January 2012).
Packer, C., Pusey, A.E., 1982. "Cooperation and competition within coalitions of male lions: kin selection or game theory?," Nature, 296(5859), P. 740 – 742.

Rechenberg, I., 1973. “Evolutionstrategie: Optimierung Technisher Systeme nach Prinzipien des Biologischen Evolution”, Fromman-Hozlboog Verlag, Stuttgart, (German)

Rozenberg, G., Bck, T., Kok, J., 2011. “Handbook of Natural Computing,” Handbook of Natural Computing, Springer Publishing Company 1st Ed.

Shadbolt, N., 2004. “Nature-inspired computing,” IEEE journal on Intelligent Systems 19(1), p. 2-3.

Souza, J.G., Costa, J.A.F., 2009. “Unsupervised data clustering and image segmentation using natural computing techniques,” IEEE International Conference on Systems, Man and Cybernetics, p. 5045 – 5050.

Steinhaus, H., 1957. "Sur la division des corps matériels en parties," (in French). Bull. Acad. Polon. Sci. 4 (12): 801–804. MR0090073.

Storn, R. 1996. "On the usage of differential evolution for function optimization," Biennial Conference of the North American Fuzzy Information Processing Society (NAFIPS), p. 519–523.

Zhang, J., 2009. “Advances in Fuzzy Method for Natural Computing,” IEEE Ninth International Conference on Hybrid Intelligent Systems 2, p. 18 – 23.