

# LACAIS: Learning Automata based Cooperative Artificial Immune System for Function Optimization

Alireza Rezvanian<sup>1</sup> and Mohammad Reza Meybodi<sup>2</sup>

<sup>1</sup> Department of Computer & IT Engineering, Islamic Azad university, Qazvin branch, Iran

<sup>2</sup> Department of Computer & IT Engineering, Amirkabir University of Technology, Tehran, Iran

rezvan@qiau.ac.ir, mmeybodi@aut.ac.ir

**Abstract.** Artificial Immune System (AIS) is taken into account from evolutionary algorithms that have been inspired from defensive mechanism of complex natural immune system. For using this algorithm like other evolutionary algorithms, it should be regulated many parameters, which usually they confront researchers with difficulties. Also another weakness of AIS especially in multimodal problems is trapping in local minima. In basic method, mutation rate changes as only and most important factor results in convergence rate changes and falling in local optima. This paper presented two hybrid algorithm using learning automata to improve the performance of AIS. In the first algorithm entitled LA-AIS has been used one learning automata for tuning the hypermutation rate of AIS and also creating a balance between the process of global and local search. In the second algorithm entitled LA-CAIS has been used two learning automata for cooperative antibodies in the evolution process. Experimental results on several standard functions have shown that the two proposed method are superior to some AIS versions.

**Keywords:** Artificial Immune System, Hypermutation, Learning Automata, Cooperative, Function Optimization.

## 1 Introduction

Global optimization problems are used in continuous spaces in various problems of communication, commerce, engineering design and biological sciences. Optimization in nonlinear, non-convex and non-differential functions remained as investigative challenge for researchers on solving of optimization problems [1]. According to applications of these problems from many years ago, several methods have been developed for solving them that can be classified them in two groups: traditional and heuristics (stochastic) methods. In most traditional methods that usually including of numerical methods such as linear programming, gradient based method or analytical methods like differential calculus and lagrange multiplies [12] meantime have answer and enough time, there is limitation like derivation on function. Also, there are other methods that find all of local minimums, finally among which global minimum are selected [1]. In spite of guaranties of definite methods, because of being time

consuming and limitations that they have on function, also, stochastic methods are important. With regard to comparison of mentioned algorithms, there are difficulties, among them, all of them have not examined with similar functions. Some of methods require arrangement of many parameters, and stating about the better methods in many cases depends on input function specifications. For example, for function that they have noises, it have been developed certain methods [13]. Of course, with due attention to [11] performance average of search methods equals on all of functions. It is too easy we cannot produce a search algorithm that relative to other algorithms on all of functions has the better performance. But each of existing algorithms has area of its certain performance.

In set of function optimization methods have been suggested certain heuristic methods that these methods in the proportion of numerical and traditional methods have more flexibility and they are usable in various areas. From heuristic methods in this fields, we can mention that simulated annealing [2], tabu search [3], genetic algorithm [4], evolutionary strategy [5] [6], particle swarm optimization [7], differential evolution algorithm [8] [9], and recently artificial immune system [10]. Among optimization methods which they have been inspired from nature, genetic algorithm is very famous. In genetic algorithm there is possibility of early local optima and it applies mutation for came out from local optima. Also, set of points that they are considered as the next generation candidate is limited.

Artificial immune system algorithms are evolutionary methods and have several powerful features, such as population size dynamically adjustable, effective exploitation exploration of the search space, location of multiple optima, capability of maintaining local optima solutions, and also having the most of genetic algorithm features without difficulties of genetic algorithm [18]. Learning automaton (LA) is a general-purpose stochastic optimization tool, which has been developed as a model for learning systems. The LA tries to determine, iteratively, the optimal action to apply to the environment from a finite number of actions that are available to it. The environment returns a reinforcement signal that shows the relative quality of an action of the LA, and then LA adjusts itself by means of a learning algorithm. LA have been used to enhance the learning capability of other algorithms such as neural networks [17], genetic algorithms [32], particle swarm optimization [14] [31] and ant colony optimization [33].

In this paper, in the first algorithm, it has been used the combination of artificial immune system and learning automata for improvement of standard AIS. But, presence of high learning rate in learning automata is resulted in increasing of convergence speed or falling in local optima and in the event a good accuracy but will reduce convergence speed. Therefore, for removing of this difficulty, in second algorithm we proposed using of cooperation between learning automata with difference learning rate. Also in previously has been introduced the other model of learning concept with cooperative in from of CLA-EC in [34]. But in suggested model in second algorithm has been used combination of two cooperative automaton with various, low and high learning rate, so, high learning rated automaton merely results in speed acceleration of antibodies towards optimum answer and at the same time of learning rated automaton in cooperation with high learning rated automaton

with interchanging some of its antibodies prevents falling in local optima.

In this paper, at first it has been introduced AIS in short, in third section it has been presented LA briefly. Proposed two algorithms has discussed in fourth section, and finally, results of experiments on proposed methods and other famous methods have been given in fifth section.

## **2 Artificial Immune System**

Artificial immune system (AIS) is one of the computational intelligence branches in computer sciences, which inspired from natural immune system to present several algorithms for solving computer problems. Natural immune system acts in various levels, which in the first level, immune system tries to prevent enter the external invaders as known pathogen using skins, tears and similar strategies. In the second level as innate immune system encountered with all kind of pathogens in form of public strategy, immune response in this step done for all of antigens similarly. Also, this level of immune system acts very slow and is not enough for encountering with antigens. In the next level has settled adaptive immune which in this level for each antigen is created ideal comparison method. This level of immune acts very fast and it can produce many immune cells for encounter with antigens. The Immune algorithms that have planning in AIS, have modeled adaptive immune and these algorithms have been used for solving many computer problems. AIS algorithms are classified to several groups: negative selection, clonal selection, bone marrow, Immune networks, and danger theory. Each algorithm has modeled part of natural immune system. According to the literature, these algorithms have been used for solving optimization problems, pattern recognition, classification, clustering, intrusion detection, and other computer problems and have obtained good results relative to existing algorithms [14] [16]. Generally, can be consider the AIS as an adaptive, very parallel and distributed system.

Mutation in genetic algorithm is used for preventing premature convergence, recycling and finding unseen and missed solutions, but in the AIS, mutation acts as only and most important operator that called Hypermutation, so acts in form of probable in the affinity is used between antibodies and antigens. Part of the population with high affinity value tolerate the lowest mutation rate and similarly part of antibodies with low affinity value tolerate highest mutation rate. Small amount of mutation rate in genetic algorithm is necessary and acts successfully, so it has created diversity in recombination. As a value, increasing cycles approaches the answer, so the rate will reduce until become zero and maintenance of most competent will become to the most amount. But in the artificial immune system algorithm, mutation should acts effectively as only and the most important operator. The general concept of standard AIS algorithm which is mentioned in the various references is given in form of Pseudo-code in figure 1 [15] [29] [30].

---

**Algorithm 1:** Standard Artificial Immune System Algorithm

---

**Initialize** population (randomly)  
**Individuals** (candidate solution)  
**Evaluation** (fitness function) for all antibodies  
**While** (termination criterion not satisfied)  
    **Select** (superior antibodies from parent population)  
    **Cloning** based on fitness value  
    **Variation** operators on clones (**Hypermutation**)  
    **Evaluating** new generated antibodies  
    **Selection** of superior antibodies  
    **Creation** of next generation population  
**End**

---

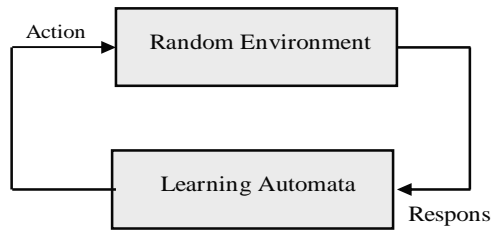
**Fig. 1.** Pseudo-code of standard AIS algorithm

Also, several applications of optimization by AIS reported for solving different problems [10] [19] such as Multi-Modal Optimization, Multi-Objective Optimization, Constrained Optimization, Combinatorial Optimization, Inventory Optimization, Time Dependent Optimization, Job Shop Scheduling, Numerical Function Optimization [20].

### 3 Learning Automata

Learning Automata (LA) are adaptive decision-making devices operating on unknown random environments. An LA has a finite set of actions and each action has a certain probability (unknown for the automaton) of getting rewarded by the environment. The aim is to learn to choose the optimal action, the action with the highest probability of being rewarded, through repeated interaction of the system. If the LA is chosen properly, the process of interacting with the environment can result in selection of the optimal action. A study reported in [28] illustrates how a stochastic automaton works in feedback connection with a random environment.

Figure 2 shows relation between learning automata and environment and figure 3 pseudo code respectively [21].



**Fig. 2.** The interaction between learning automata and environment

---

**Algorithm 2: Learning Automata**

---

**Initialize**  $p$  to  $[1/s, 1/s, \dots, 1/s]$  where  $s$  is the number of actions

**While** not done

    Select an action  $i$  a sample realization of distribution  $p$

    Evaluate action and return a reinforcement signal  $\beta$

    Update probability vector according to learning algorithm

**End While**

---

**Fig. 3.** Pseudo-code of variable-structure learning automaton

Variable structure LA can be defined as  $\{\alpha, \beta, p, T\}$  which  $\alpha = \{\alpha_1, \dots, \alpha_r\}$  is set of automata actions,  $\beta = \{\beta_1, \dots, \beta_m\}$  is set of automata inputs,  $p = \{p_1, \dots, p_r\}$  is vector of each actions and  $p(n+1) = T[\alpha(n), \beta(n), p(n)]$  is the learning algorithm.

The following algorithm is a sample of linear learning algorithms. We assume that action  $\alpha_i$  is selected at time step  $n$ .

In case of desirable response from the environment:

$$\begin{aligned} p_i(n+1) &= p_i(n) + a[1 - p_i(n)] \\ p_j(n+1) &= (1-a)p_j(n) \quad \forall_j \quad j \neq i \end{aligned} \quad (1)$$

In case of undesirable response from the environment:

$$\begin{aligned} p_i(n+1) &= (1-b)p_i(n) \\ p_j(n+1) &= \left(\frac{b}{r-1}\right) + (1-b)p_j(n) + \quad \forall_j \quad j \neq i \end{aligned} \quad (2)$$

In equations (1) and (2),  $a$  is the reward parameter and  $b$  is the penalty parameter. When  $a$  and  $b$  are equal, the algorithm is called  $L_{RP}$ , when  $b$  is much smaller than  $a$ , the algorithm is  $L_{ReP}$ , and when  $b$  is zero, it is called  $L_{RI}$ .

More details and information about learning automata, obtained in [22].

## 4 Proposed Method

In this section it has been explained proposed method for improvement of AIS algorithm based on learning automata. Our suggestion to improve artificial immune system algorithm is configuration of rate of probability mutation changes. Mutation rate in standard algorithm and some of extended versions in constant and usually take places with reverse of affinity amount in form of adaptive for example in AIGA model [18] also has been used from  $\alpha$  value as common balance coefficient in mutation rate changes. In the first algorithm, it proposed to use one LA to configure mutation rate as adaptively. LA has three actions: *increasing of mutation rate*, *decreasing of mutation rate*, and *no change of mutation rate*. In the second algorithm, it use two LA for two antibodies groups, similarly the LAs have three actions: *increasing of mutation rate*, *decreasing of mutation rate*, and *no change of mutation*

rate, with different learning rate and similarly actions. In each stage, automatons have selected one of these actions and based on selected action is corrected parameter value of mutation rate, consequently become well. LA updates probability of suitable action (mutation rate change) according to feedback from environment. General structure of LA has been shown in figure 4.

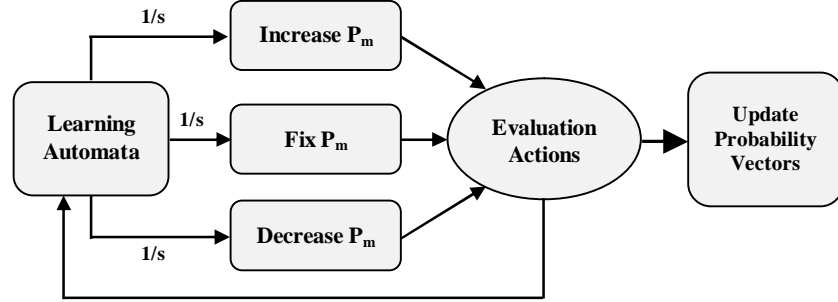


Fig. 4. Structure of proposed learning automata

Working of two algorithms is the same but in the second algorithm, cooperative automatons with different LA in each groups exchange some of antibodies randomly in certain period of iterations until to satisfy stopping conditions. Generally we can state stages of these algorithms as follows in figure 5 and figures 6.

---

**Proposed Algorithm 1: LA-AIS**

---

1. Initialize antibodies.
  2. Initialize LA.
  3. Affinity value of antibodies is calculated.
  4. According to affinity value of antibodies is accomplished process of clonal selection.
  5. Learning automatons select one of actions according to probability vectors of their actions.
  6. Based on to selected actions, quality of mutation amount of antibodies is specified and improved new value and hypermutation changes.
  7. Affinity values of antibodies are calculated again.
  8. Based on updating results by hypermutation in antibodies is studied affinity amount of antibodies and evaluated learning automata performances and update probability vector of selecting learning automata actions.
  9. Replacing antibodies is accomplished by colonies with high affinity and some parts of antibodies are eliminated with low and dense fitness.
  10. Go to 2 if termination criterion not satisfied
- 

Fig. 5. Proposed algorithm 1: Learning Automata based Artificial Immune System (LA-AIS)

In each algorithm selected action evaluated by average performance of antibodies (fitness average), that it compared in previous situation, in the event that it is being better from current situation, the selected action is evaluated positive otherwise the

action is evaluated negative.

---

**Proposed Algorithm 2: LA-CAIS**

---

1. Initialize antibodies in two groups with difference learning automata.
  2. Initialize LAs in two groups (Low and High learning rate)
  3. Affinity value of antibodies is calculated.
  4. According to affinity value of antibodies is accomplished process of clonal selection for two groups simultaneously.
  5. In each group learning automata select one of actions according to probability vectors of their actions.
  6. Based on to selected actions, quality of mutation amount of antibodies is specified and improved new value and hypermutation changes for two groups simultaneously.
  7. Affinity values of antibodies are calculated again for two groups.
  8. Based on updating results by hypermutation in antibodies is studied affinity amount of antibodies and evaluated learning automata performances and update probability vector of selecting learning automata actions for two groups simultaneously.
  9. Replacing antibodies is accomplished by colonies with high affinity and some parts of antibodies are eliminated with low and dense fitness for each group.
  10. Exchange some of their antibodies randomly between two groups when achieving to expected period of iterations.
  11. Go to 2 if termination criterion not satisfied
- 

**Fig. 6.** Proposed algorithm 2: Learning Automata based Cooperative Artificial Immune System (LA-CAIS)

The most important benefit of this method especially second algorithm is its high ability to escape from local optimum or settling in suitable convergence. Indeed with increasing mutation rate value, is grew beam of changes and has been accomplished global search and with decreasing mutation rate value, beam of changes becomes small and accomplishing a local search on the search space. In first algorithm, in spite of learning automata with constant learning rate, there were a constant convergence rate, but in second algorithm, presence of two learning automata with low and high learning rates meantime establishing fast and parallel search is created possibility of cooperation between them with exchanging intermediate solutions.

## 5 Experimental Results

For examining the proposed methods have done experiments on four famous standard functions that usually they are used as criterion of evaluating methods in most of literature [14]. Functions that are used consist of Sphere, Rastrigin, Ackley, and Rosenbrock which have been defined by equations of (3) to (6) respectively [14].

$$f_1(x) = \sum_{i=1}^n x_i^2 \quad (3)$$

$$f_2(x) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10) \quad (4)$$

$$f_3(x) = 20 + e - 20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^n x_i^2}} - e^{\frac{1}{n}\sum_{i=1}^n \cos(2\pi x_i)} \quad (5)$$

$$f_4(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2) \quad (6)$$

All of these functions have global optimum with zero values. Initial population size and the number of steps have been considered 20 and 500, respectively.

For comparison, the numerical experiments were performed with six algorithms by results of average and best for Standard Artificial Immune System algorithm as *SAIS* [23], B-Cell Algorithm as *BCA* [24], Clonal Selection Algorithm as *CSA* [25], Adaptive Clonal Selection Algorithm as *ACSA* [26], Optimization Artificial Immune Network as *OAIN* [27], Standard Genetic Algorithm as *SGA* [4], Artificial Immune-Genetic Algorithm as *AIGA* [18], and finally proposed method in algorithm 1, Learning Automata-based Artificial Immune System as *AISLA* and algorithm 2, Learning Automata-based Cooperative Artificial Immune System as *CAISLA*.

In  $LA_{RP}$ , penalty and reward rate values have been considered  $a=b=0.01$  and for  $LA_{REP}$   $a=0.01$  and  $b = 0.001$ .

Also cooperation automata has been considered in form of  $LA_{RI}$  for which high learning rated automata  $a=0.01$  and for low learning rated automata  $a=0.001$ . And iterations have been considered in each exchange period equal to 10 evaluations, and experimented function dimension has been considered in form of 10 dimensional.

Results of these experiments with given results in other common methods has been compared for *Sphere*, *Rastrigin*, *Ackley*, and *Rosenbrock* in tables 1.

**Table 1.** Performance of the proposed methods and other methods for benchmark functions

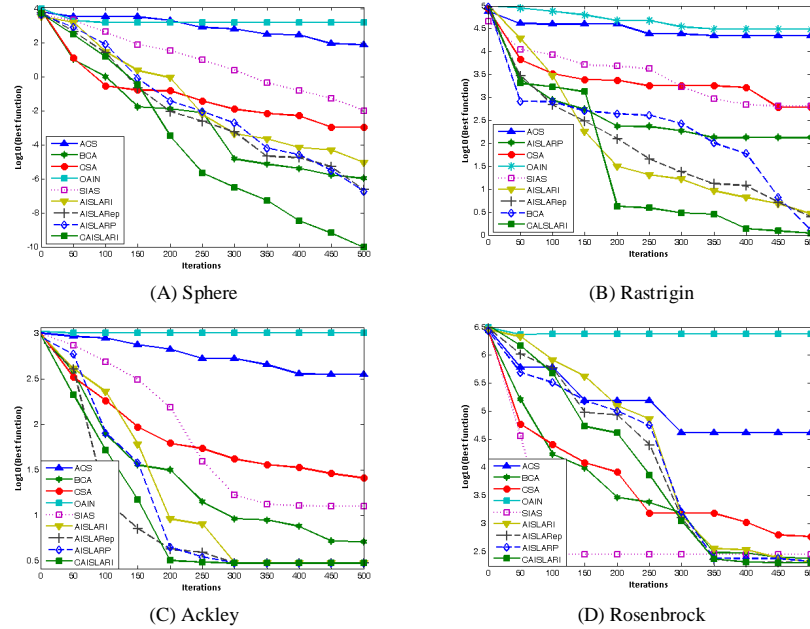
(A) Sphere			(B) Rastrigin		
Method	Best	Average	Method	Best	Average
SAIS	0.1349	0.6561	SAIS	16.6119	17.1973
BCA	0.0025	1.7112	BCA	8.3354	16.6878
CSA	0.00526	2.5832	CSA	16.1979	33.4838
ACSA	6.6806	9.3832	ACSA	77.7343	96.279
OAIN	24.3793	24.6139	OAIN	89.2454	95.2565
SGA	1.604	3.0526	SGA	13.9295	17.8693
<b>AISLA<sub>RI</sub></b>	0.0065	0.1801	<b>AISLA<sub>RI</sub></b>	1.5910	5.3821
<b>AISLA<sub>REP</sub></b>	0.0013	0.1958	<b>AISLA<sub>REP</sub></b>	1.4834	5.9271
<b>AISLA<sub>RP</sub></b>	0.0012	0.1201	<b>AISLA<sub>RP</sub></b>	1.1175	5.0602
<b>CAISLA<sub>RI</sub></b>	0.000044	0.0026	<b>CAISLA<sub>RI</sub></b>	1.0399	4.1466



(C) Ackley		
Method	Best	Average
SAIS	3.0271	3.3249
BCA	2.0442	6.0789
CSA	4.1067	5.7433
ACSA	12.8586	13.9453
OAIN	20.2124	20.2534
SGA	0.8392	1.3469
AISLA <sub>RI</sub>	1.6166	1.6503
AISLA <sub>Rep</sub>	1.6143	1.6393
AISLA <sub>RP</sub>	1.6136	1.6240
CAISLA <sub>RI</sub>	1.6131	1.6203

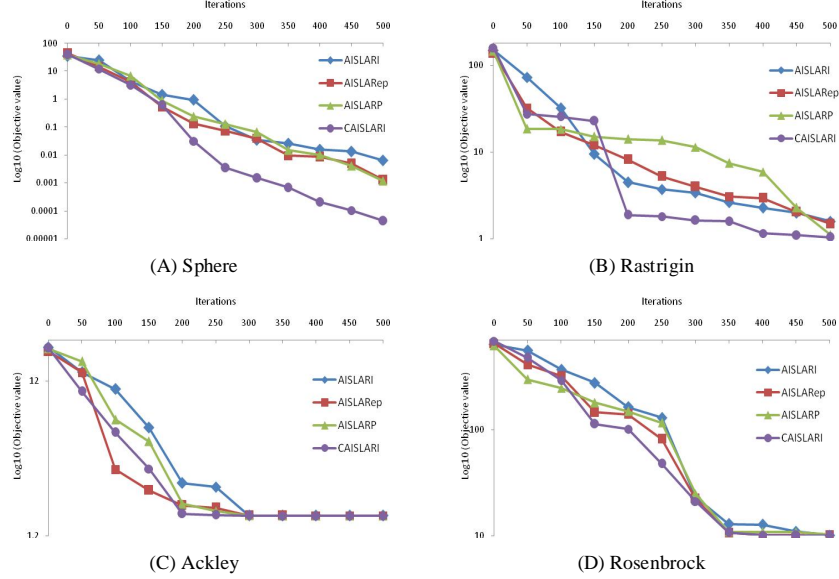
(D) Rosenbrock		
Method	Best	Average
SAIS	11.6153	11.6351
BCA	10.8634	73.0078
CSA	15.9718	83.2074
ACSA	100.9198	113.9453
OAIN	591.5480	591.5525
SGA	16.9974	24.3716
AISLA <sub>RI</sub>	10.1321	12.3744
AISLA <sub>Rep</sub>	10.0530	12.9905
AISLA <sub>RP</sub>	10.2448	10.2527
CAISLA <sub>RI</sub>	10.0117	11.2650

For better comparison of proposed method and other methods, comparison graph of converge rate in form of algorithm have been shown in figure 7 for Sphere, Rastrigin, Ackley, and Rosenbrock function.



**Fig. 7.** Comparison of proposed method and other methods for benchmark functions

Also, for better evaluation between different states of proposed learning automatons, another comparison has been given for proposed methods in form of algorithm in figure of 8 for Sphere, Rastrigin, Ackley, and Rosenbrock.



**Fig. 8.** Comparison of experimental results on learning automata types for proposed algorithms

As shown results, proposed algorithm 2 as learning automata-based cooperative artificial immune system, meantime it have the algorithm 1 features by which feedback that it gets from environment, gives more suitable behavior in search space and it has desirable efficiency, with addition of cooperation concept between automaton as create a good balance among convergence. In fact, in first proposed method Hypermutation as move important and only operator is balanced effectively in artificial immune system algorithm until improves convergence behavior and having better adaptive method to standard state, and with second propose has lower sensitivity to parameter arrangement state and it has better adaption.

## 6 Conclusion

In this paper, a new two methods have been presented for improvement of AIS algorithm using learning automata and cooperation concept between automaton in optimization. In all versions of AIS, rate of mutation changes as only and most important evolutionary operator is constant and it is with due attention to contrary of antibodies distances. But in proposed method using learning automata with due attention to feedback of environment changes rate is updated and addition of cooperation concept between two automaton with different learning rate is establish on equilibrium between local and global search. We never can give a method which it can cover successful on all functions but at the same time it has been shown in experiment results that proposed methods have had relative improvement to some of the other version of artificial immune system. Moreover improvements in proposed

methods with combination of soft computing concepts and emphasizing on the other applications is considered in the future works.

## References

1. Wang, Y.J., Zhang, J.S., Zhang, G.Y.: A Dynamic Clustering based Differential Evolution Algorithm for Global Optimization. *European Journal of Operational Research*, vol. 183, pp. 56--73 (2007)
2. Kirkpatrick, S., Gelatt C.D., Vecchi, M.P.: Optimization by Simulated Annealing. *Science*, vol. 220, no. 4598, pp. 671--680 (1983)
3. Hedar, A.R., Fulushima, M.: Tabu Search Directed by Direct Search Methods for Nonlinear Global Optimization. *European Journal of Operational Research*, vol. 170, pp. 329--349 (2006)
4. Fogel, D.B., Michalwicz, Z.: *Evolutionary Computation 1 - Basic Algorithms and Operators*. Bristol, UK, Institute of Physics (IoP) Publishing (2000)
5. Herrera, F., Lozano, M., Molina, D.: Continuous Scatter Search: An Analysis of the Integration of Some Combination Methods and Improvement Strategies. *European Journal of Operational Research*, vol. 169, no. 2, pp. 450--476 (2006)
6. Hedar, A., Fukushima, M.: Evolution Strategies Learned with Automatic Termination Criteria. In: *SCIS&ISIS 2006*, Tokyo, Japan (2006)
7. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: *IEEE International Conference on Neural Networks*, Piscataway, NJ, pp. 1942--1948, IEEE Press (1995)
8. Price, K., Storn, R., Lampinen, J.: *Differential Evolution - A Practical Approach to Global Optimization*. Springer Verlag (2005)
9. Qin, A.K., Huang, V.L., Suganthan, P.N.: Differential Evolution Algorithm with Strategy Adaptation for Global Numerical Optimization. *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 2, pp. 398--417 (2009)
10. Gong, M., Jiao, L., Zhang, X.: A Population-based Artificial Immune System for Numerical Optimization. *Neurocomputing*, vol. 72, no. 1-3, pp. 149--161 (2008)
11. Bozorgzadeh, M.A., Rahimi, A., Shiry, S.: A Novel Approach for Global Optimization in High Dimensions. In: *12th Annual CSI Computer Conference of Iran*, pp. 1--8, Tehran, Iran (2007)
12. Vanderplaats, G.N.: *Numerical Optimization Techniques for Engineering Design with Applications*. New York, McGraw Hill (1984)
13. Huyer, W., Neumaier, A.: SNOBFIT--Stable Noisy Optimization by Branch and Fit. *ACM Transactions on Mathematical Software*, vol. 35, no. 2, pp. 9 (2008)
14. Hashemi, A.B., Meybodi, M.R.: A Note on the Learning Automata based Algorithms for Adaptive Parameter Selection in PSO: *Journal of Applied Soft Computing*, (2010) to appear.
15. Timmis, J., Hone, A., Stibor, T., Clark, E.: Theoretical advances in artificial immune systems. *Theoretical Computer Science*, vol. 403, no. 1, pp. 11--32 (2008)
16. Dasgupta, D.: *Artificial Immune Systems and their Applications*. Springer-Verlag New York (1998)
17. Meybodi, M. R., Beigy, H.: A Note on Learning Automata Based Schemes for Adaptation of BP Parameters. *Journal of Neurocomputing*, vol. 48, no. 4, pp. 957--974 (2002)
18. Yongshou, D., Yuanyuan, L., Lei, W., Junling, W., Deling, Z.: Adaptive Immune-Genetic Algorithm for Global Optimization to Multivariable Function. *Journal of Systems Engineering and Electronics*, vol. 18, no. 3, pp.655--660 (2007)

19. Wang, X., Gao, X.Z., Ovaska, S.J.: Artificial Immune Optimization Methods and Applications - A Survey. In: IEEE International Conference on Systems, Man and Cybernetics, vol.4, pp. 3415--3420, IEEE Press (2004)
20. Campelo, F., Guimaraes, F. G., Igarashi, H.: Overview of Artificial Immune Systems for Multi-objective Optimization. EMO 2006. LNCS, vol. 4403, pp. 937--951. Springer, Heidelberg (2006)
21. Sheybani, M., Meybodi, M. R.: PSO-LA: A New Model for Optimization. In: 12th Annual CSI Computer Conference of Iran, pp. 1162--1169, Tehran, Iran (2007)
22. Meybodi, M. R., Kharazmi, M. R.: Application of Cellular Learning Automata to Image Processing. J. Aut. vol. 14, no. 56A, pp. 1101--1126 (2004)
23. Cutello, V., Nicosia, G.: The Clonal Selection Principle for In Silico and In Vivo Computing. Recent developments in biologically inspired computing, pp. 104--146, Idea Group Publishing (2005)
24. Timmis, J., Edmonds, C., Kelsey, J.: Assessing the Performance of Two Immune Inspired Algorithms and a Hybrid Genetic Algorithm for Function Optimisation. In: IEEE Congress on Evolutionary Computation, Portland, Oregon, USA, vol.1, pp. 1044--1051, IEEE Press (2004)
25. De Castro, L.N., Von Zuben, F.J.: Learning and optimization using the Clonal Selection Principle. IEEE Transactions on Evolutionary Computation, vol. 6, no. 3, pp. 239--251 (2002)
26. Garrett, S.M.: Parameter-free, adaptive clonal selection. In: IEEE Congress on Evolutionary Computation, Portland, Oregon, USA, vol.1, pp. 1052--1058 (2004)
27. Cutello, V. and Nicosia, G. and Pavone, M.: Real Coded Clonal Selection Algorithm for Unconstrained Global Numerical Optimization using a Hybrid Inversely Proportional Hypermutation Operator. In: 21st Annual ACM Symposium on Applied Computing, Dijon, France, pp. 950--954 (2006)
28. Narendra, K.S., M.A. L Thathachar.: Learning Automata: An Introduction. Prentice-Hall Inc (1989)
29. Khilwani, N., Prakash, A., Shankar, R., Tiwari, MK.: Fast clonal algorithm. Engineering Applications of Artificial Intelligence, vol. 21, no. 1, pp. 106--128 (2008)
30. De Castro, L.N., Von Zuben, F.J.: Recent developments in biologically inspired computing. Igi Global (2004)
31. Sheybani, M., Meybodi, M.R.: CLA-PSO: A New Model for Optimization. In: 15th Conference on Electrical Engineering, Volume on Computer, Telecommunication Research Center, Tehran, Iran (2007)
32. Abtahi, F., Meybodi, M.R., Ebadzadeh, M. M., Maani, R.: Learning Automata-Based Co-Evolutionary Genetic Algorithms for Function Optimization. In: IEEE 6th International Symposium on Intelligent Systems, Subotica, Serbia, pp. 1--5, IEEE Press (2008)
33. Ebdali, F., Meybodi, M.R.: Adaptation of Ants colony Parameters Using Learning Automata. In: 10th Annual CSI Computer Conference of Iran, pp. 972--980 (2005)
34. Masoodifar, B., Meybodi, M.R., Hashemi, M.: Cooperative CLA-EC. In: 12th Annual CSI Computer Conference of Iran, pp. 558--559 (2007)