

CONVERGENCE TENDENCY OF GENETIC ALGORITHMS AND ARTIFICIAL IMMUNE SYSTEM IN SOLVING CONTINUOUS OPTIMIZATION FUNCTIONS

Isa Maleki¹, Marjan Mahmoodi Tabrizi², Seyyed Reza Khaze¹, Seyyed Amir Reza Abedini¹

¹Department of Computer Engineering, Dehdasht Branch, Islamic Azad University, Dehdasht, Iran

²Department of Computer Engineering, Science and Research Branch, Islamic Azad University, West Azerbaijan, Iran

ABSTRACT

By the advances in the Evolution Algorithms (EAs) and the intelligent optimization methods we witness the big revolutions in solving the optimization problems. The application of the evolution algorithms are not only not limited to the combined optimization problems, but also are vast in domain to the continuous optimization problems. In this paper we analyze and study the Genetic Algorithm (GA) and the Artificial Immune System (AIS) algorithm which are capable in escaping the local optimization and also fastening reaching the global optimization and to show the efficiency of the GA and AIS the application of them in Solving Continuous Optimization Functions (SCOFs) are studied. Because of the multi variables and the multi-dimensional spaces in SCOFs the use of the classic optimization methods, is generally non-efficient and high cost. In other words the use of the classic optimization methods for SCOFs generally leads to a local optimized solution. A possible solution for SCOFs is to use the EAs which are high in probability of succeeding reaching the local optimized solution. The results in paper show that GA is more efficient than AIS in reaching the optimized solution in SCOFs.

KEYWORDS

Evolution Algorithms, Genetic Algorithm, Artificial Immune System, Solving Continuous Optimization Functions, Optimization

1. INTRODUCTION

At the late decades EAs has utilized as a searching tool according to the population in combined optimization fields. The ease of use and the accessibility to the close to optimized solution are of the success reasons of EAs [1]. In contrast to other optimization algorithms like calculation methods and the methods based on gradient, EAs operates on a set of the solutions in a searching space or uses the cooperative or comparative relations of the solutions and is able to find the optimized solution in the complex optimization problems very fast [2]. The searching process takes place in parallel form based on population in EAs and this had led to relative solution of the local optimized trap and it is able to increase the convergence rate in finding the global optimized answer.

GA [3] is one of the random searching algorithms which are inspired from the nature. GA is very applicative in solving the combined optimization problems and the NP-Hard problems [4, 5]. GA is an optimization calculation algorithm which takes into consideration a set of the points of solution space in each calculation repetition and searches the solution space affectively. In GA, the Select, Crossover and Mutation operands make the new points of solution space be searched in each calculation repetition and also lead to increase in searching capability and finding the optimized points of the searching space and it means that the capabilities of the algorithm search increases.

AIS algorithm [6, 7] is a searching algorithm based on the population among EAs and swarm intelligence for solving the optimization problems and so is very important. AIS algorithm is a EAs which is inspired from the defense mechanisms of the immune system of the living beings. This algorithm is inspired from the immune system of the body of the human beings to fight against the affecting factors and preserve the body from their harms. AIS has been used for solving different engineering problems like computer networks [8], shop scheduling [9], task scheduling [10] and many other application.

In continuous optimization problems, the searching status space of the solution is very vast and a long time is needed for the optimized searching of them. For the continuous optimization problems the fitness function, function variable or the algorithmic situations may not change for long time. So in this paper we use AIS and GA to study the SCOFs.

This paper is organized as follows: in the section 2, we have introduce the related works; in the section 3, the EAs is introduced; in section 4, the analysis of the AIS and GA for SCOFs are presented; in section 5, we have evaluated and results presented the AIS and GA algorithms; in section 6, the AIS and GA algorithms are discussed and at finally in section 7, we have presented the conclusions and the future works to be done.

2. RELATED WORKS

Taking into consideration the up growing complexity of the calculations and the presence of many affective factors in optimization problems which search for the solutions among many optimized solutions, it is generally not suitable to use the classic searching methods and test all solutions. So, the most important way to solve the optimization problems is to use EAs. EAs are inspired from the natural evolution concepts to search for the optimized answer.

Researchers [11] have used the Bacterial Foraging Optimization Algorithm (BFOA) and the Particle Swarm Optimization (PSO) for SCOFs. BFOA which is inspired from the behavior of the bacteria, works based on the random search and population and tendency to the behavior. BFOA is high in convergence speed, flexible and is very tolerant for errors which are the specifications needed for solving the continuous optimization problems. In PSO algorithm, the particles are continuously analyzing the fitness of the solutions of the problems and represent the situation which is the best in solution as the final answer. To study the efficiency of the BFOA and PSO, some continuous and multi-dimensional functions are studied. The results of the examinations show that the accurate answers for the continuous optimization functions using the BFOA is very complex. In PSO when the particles are convergent to an optimized path, there is more possibility of finding the better answer and escaping from the local optimized points and finding the accurate optimized global answer.

Researchers [12] have studied the efficiency of the Artificial Bee Colony (ABC) algorithm and Firefly Algorithm (FA) for SCOFs. And also they have studied these algorithms from the

accuracy in reaching the optimized answer and the speed of answering and the reliability in reaching the optimized answer. They have studied the two and three dimensional Rastrigin function to evaluate the efficiency of the ABC and FA algorithms. The results of the tests show that the ABC algorithm in SCOFs with large dimensions is more efficient than FA and is more able in finding the global optimized points. So, the ABC algorithm is able to maintain the global and local search equilibrium in an optimized manner even by increase in the dimensions of the problem.

Researchers [13] have used the GA and Artificial Neural Networks (ANNs) combination to solve the one variable optimization function. They have used ANNs learning for optimization of GA. The goal of ANNs learning is to find the weights in the medium layer and minimize the learning errors. So ANNs learning can contain an optimization problem which aims the optimization of the weight coefficients and also the number of the learning times is predicted to be the maximum value. To increase the efficiency of GA, the optimized value for the variable of the function is found using ANNs. The results of the tests show that the algorithm is good in finding the minimum of the function.

Reference [14] has used GA and BFOA combination for SCOFs. In this reference the BFOA is used for increasing the efficiency of the crossover and mutation operands in reaching the optimized answer. To show the efficiency of the GA and the combination algorithm, the results are studied on a function. The results of the tests show that the combination algorithm is more efficient than GA algorithm.

Reference [15] has used GA and PSO to study the SCOFs. In this reference GA and PSO are tested on 36 functions to evaluate the efficiency. To show the GA and PSO efficiency, the functions are studied in a 30 dimensional space. The results of the tests show that these algorithms are good on the 30 functions and have reached the optimized answer.

Reference [16] has used AIS algorithm to study SCOFs. They have studied the AIS algorithm operands and have cited that the primary population and the affinity factor highly affect the efficiency of the algorithm. The results of the tests on 16 functions show that the AIS algorithm is highly efficient in finding the optimized answer. So it seems that for SCOFs which is very vast in its space and it is not possible to find the optimized answer in most situations, the answer close to the optimized could be accessed using the AIS algorithm.

Reference [17] has proposed the AIS algorithm to solve the Constrained Optimization Problems (COPs). As the status space of the COPs is very vast, and a long time is needed for searching it, many efforts have made to design and present the efficient algorithms for solving these problems. Also, AIS algorithm in this reference is used for solving the multi-purpose optimization problems. The results of the tests show that AIS is more efficient than the other algorithms.

Reference [18] has used GA algorithm to solve the multi-purpose optimization functions. In multi-purpose optimization there is not a unique answer but there are a set of the answers which are optimized for the decider and the decider can select one of them according to the existing situation. So, in this reference the MPX (multi-parent crossover with poly-nominal distribution) and MLX (multi-parent crossover with lognormal distribution) crossover operands are used for increasing the efficiency of GA. The efficiency of these operands is evaluated on 6 functions. The results show that GA has presented better solutions using these solutions.

3. EVOLUTION ALGORITHMS

The EAs include two stages of diversity and repetition. In the beginning stage the primary population is created according to the harmonized random distribution in a way that covers the total space of the problem. Then at the repetition stage the fitness of the individuals of the population according to a pre-defined fitness function is defined. At the next repetition the new generation is created using the two crossover and mutation operands. This cycle continues till the ending condition is met.

3.1. GA

GA is one of the EAs which is introduced first in 1975 by John Holland [3]. GA is one of the EAs optimization which takes into consideration the set of the points of answer space in each calculation repetition and effectively searches the different spaces. This algorithm creates a primary population to answer the problem. Any individual of the population is called chromosome which are considered as an answer to the problem and any element of chromosome is called gene meaning that the special variable of the aimed optimization problem. The new generation takes into consideration the individual fitness function and uses the GA operands (mutation and crossover) and the fitness function of the individuals get improved in any repetition. The execution of GA includes the following steps:

- Creating the initial population: The population of the chromosomes is created randomly. In creating the primary population the limits of the problems must be taken into consideration.
- Evaluation: This operand uses an evaluation function for defining the fitness of any chromosome in comparison to the others. In fact this operand shows the suitability rate of any chromosome for solving the problems.
- Selection: This operand selects the most fitted chromosome of the population for the next generation. If a chromosome does not meet the limits of the problem will not be selected.
- Crossover: The creation of the new chromosomes from the existing ones takes place using the crossover operand.
- Mutation: This operand selects the random genes from the random chromosomes and changes their value. The probability of any mutation on the genes of the chromosomes is very low and the goal of execution of it is to maintain the genetic diversity of the population in convergence to the optimized solution.

The most important specification of the GA is the elitism technique. In elitism technique the best chromosomes are selected for production of the new generation. The Elitism technique is able to increase the speed of GA because this method leads to increase in probability of the selection of the best chromosomes.

3.2. AIS Algorithm

AIS algorithm is inspired from the body of the living beings [6, 7]. The goal and the responsibility of the defense system of the body of living beings are to preserve the body from the illness factors and finally crating a stable status on the body. SO, the defensive system of the body of the living beings must be able to define the illness factors like bacteria and viruses which threaten the health of the body and when they are defined the series of operations to kill them are started which are called immune response. An illness factor includes different parts. One of these parts which are located at the outer surface of the factor is called anti-gene. The operation of body

system against the speedy proliferation of the defensive cells is to define the anti-genes using the anti-bodies.

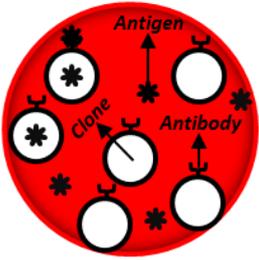


Figure 1: The presentation of the defensive cells

The efficiency of the Immune System of the body of the living beings in diagnosis of the anti-genes and removing them by producing the needed anti-bodies led researchers utilize the system in solving the complex engineering problems by simulating the operation of the system. When the anti-genes attack the organism of the body, they not only ruin the cells, but also replicate. So, one of the attracting mechanisms of the defensive system of the body of the living beings against the attack is to replicate the defensive cells which are able to identify the anti-genes and remove them. The diagnosis of an anti-gene by the anti-body takes place using the affinity factor. Anti-bodies which are less related to each other use the mutation operand to make structural changes in the illness factors and then improve their defensive operation. The designing and implementation process of the AIS includes the following stages:

- **Creating the initial population:** In this section the population of the anti-bodies in a defined number is created taking into consideration the problem.
- **Evaluation:** In this section the nearest colony of the anti-bodies to the anti-gene is evaluated. According to the colony, anti-body is grown for a cell which the anti-gene diagnoses. So, the diversity is based on the colony. The number of the produced colony for an anti-body is defined according to the equation (1) [19, 20].

$$N_c = \sum_{i=1}^n \text{round}\left(\frac{\beta \cdot N}{i}\right) \tag{1}$$

In the equation (1), N_c is the sum of the total produced colony for anti-bodies, β is the proliferation coefficient, N is the total anti-bodies number and $\text{round}()$ function is the operand which rounds the input argument to the nearest number. The value which the equation (1) returns for anti-body shows the number of the colony of the anti-body.

- **Selection:** In this part the colony which is the most dependent to the anti-gene is selected.
- **Mutation:** in this section the nearest anti-bodies to the anti-gene are replicated. The diagnosis of the anti-gene and the more dependency on it leads to the more proliferation of the anti-bodies, which is changed in structure by the mutation operand.
- **Deleting and Replacing:** The best colony of the anti-bodies is specified for the anti-genes and the non-necessary anti-bodies are randomly delete or replaced in defensive cells.

According to what is said the five concepts of affinity, selection, mutation and deleting and replacing are very effective in AIS operation. In the Figure (2) the stages of AIS algorithm operation are shown.

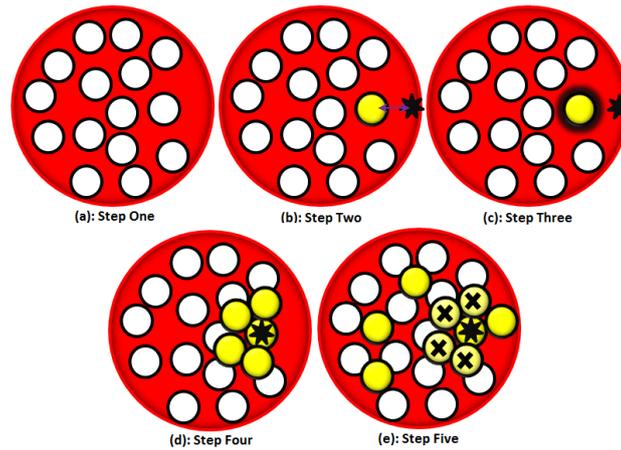


Figure 2: The stages of AIS algorithm operation

Of the very important specification of AIS algorithm it is possible to cite the selection and the colony of the anti-bodies which is the high stimulator and is suitable to the affinity rate of the anti-genes. In this algorithm anti-bodies are completed according to the colony and the best of them is selected in execution of the algorithm. So, in AIS algorithm the colonies shape the best solutions. The solutions of any repetition go to the next one to be optimized. Any colony holds a memory which saves the best solution in it and this memory is used for creating the best solution. The constructed solutions in any repetition are updated according to the fitness of the colonies' memory. If the solution is better than the one saved in memory, then it replaces that one.

4. PROPOSED SOLUTION

In this paper we have used the AIS and GA for SCOFs. The main reason for using EAs in SCOFs is the number of the variables and the limits of the continuous optimization functions and also finding the optimized solution for them using the linear impossible or long time algorithms. The speed of EAs is very high for the optimization problems and it is possible to use EAs to get the optimized solution in an accepted time.

4.1. SCOFs using the GA

Solving any optimization problem using the GA, the first stage of optimization of the problem must be taken into consideration to change the problem to a Genetic algorithm. As GA is in its nature suitable for solving the maximizing functions in non-continuous shape, so we first transform the continuous optimization function which is generally a maximizing problem to a continuous minimizing problem and then code the variables of the continuous optimization problem using the suitable numbers. So it is necessary to define the minimizing function in a way that it will be possible to maximize it and reach the optimized answer. So, it is possible to change the goal function in a way to minimize the goal function. For this reason if the f function is a continuous optimization function, the equation (2) is able to minimize it.

$$F(x_1, \dots, x_n) = \frac{1}{f(x_1, \dots, x_n)} \quad (2)$$

Using the GA operands like selection, crossover and mutation the new generation of the chromosomes are created and at last if the condition for ending the algorithm execution is met,

the execution of the program is met and the best achieved chromosome in the last generation or the best chromosome of all generations is selected as the best answer for the continuous optimization function. GA flowchart for SCOFs is showed in the Figure (3).

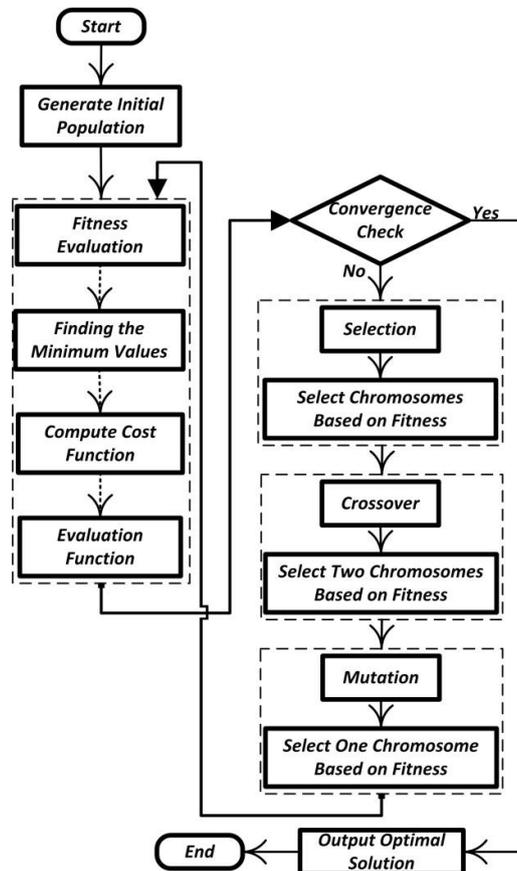


Figure 3: GA flowchart for SCOFs

In the Figure (4) the quasi code GA for the SCOFs is showed:

1. Initialize Parameters
2. Generate Initial Population
3. Do
 - Fitness Evaluation
 - Finding the Minimum Values
 - Compute Cost Function
 - Evaluation Function
 - Selection
 - Crossover
 - Mutation
4. While (a stop criteria maximum iteration)
5. Output Optimal Solution
6. End

Figure 4: SCOFs using the GA

4.2. SCOFs using the AIS Algorithm

AIS algorithm is able to combine many applicative specifications like resisting the errors, dynamic learning and flexibility and make solving the problems easier. In AIS algorithm first the initial population is produced randomly. Any of the anti-bodies include a solution for optimization problems which cover the space of the problem. In any repetition the most fitted colony of the anti-bodies is selected and the fitness is calculated. If the fitness of the primary population in some repetitions does not reach the optimized solution, the all solutions which are not efficient in an optimized manner are deleted and the new population is replaced. This operation is repeated in a defined number and finally the most fitted answer for optimization problem is defined. AIS flowchart for SCOFs is showed in Figure (5).

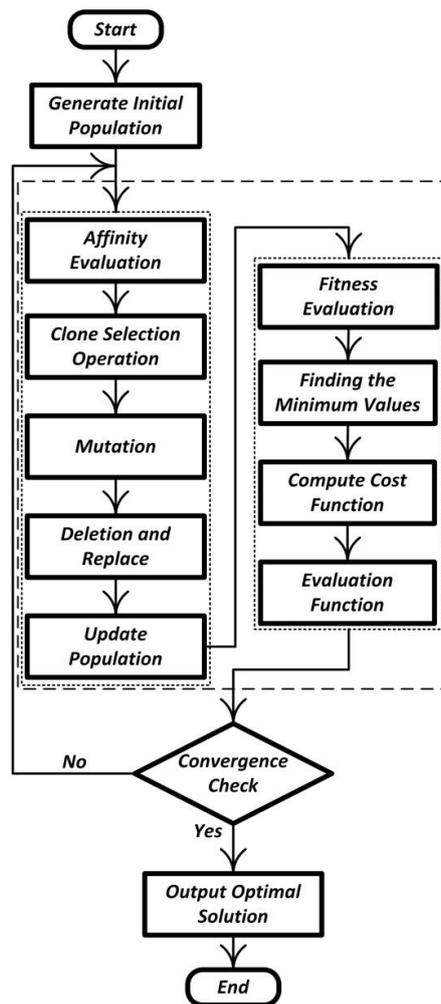


Figure 5: AIS flowchart for SCOFs

In Figure (6) the quasi code AIS for SCOFs is showed:

<ol style="list-style-type: none"> 1. <i>Initialize Parameters</i> 2. <i>Generate Initial Population</i> 3. <i>Repeat</i> 4. <i>Affinity Evaluation</i> 5. <i>Clonal Selection Operation</i> 6. <i>Mutation</i> 7. <i>Deletion and Replace</i> 8. <i>Update Population</i> <p style="margin-left: 20px;"> <i>Fitness Evaluation</i> <i>Finding the Minimum Values</i> <i>Compute Cost Function</i> <i>Evaluation Function</i> </p>

Figure 6: SCOFs using the AIS

5. EVALUATION AND RESULTS

To evaluate the efficiency of the GA and AIS algorithms, we have analyzed the two and three dimensional Rastrigin function [21]. This function holds many minimum and maximum points which lead to this fact that this function be used as a test function for evaluation of the evolution algorithms. So to compare the efficiency of these algorithms the Rastrigin function is used. The goal of utilizing the GA and AIS algorithms in solving this function is to reach the minimum value.

Two dimensional Rastrigin function

$$f_1(x) = 20 + \sum_{i=1}^2 [x_i^2 - 10 \cos(2\pi x_i)]$$

Three dimensional Rastrigin function

$$f_2(x) = 30 + \sum_{i=1}^3 [x_i^2 - 10 \cos(2\pi x_i)]$$

In EAs it is very important to set the parameters to reach the optimized answer. And this can affect the operation and the efficiency of the optimized solutions. So, EAs are very delicate in the parameters. Setting the value of the parameters leads EAs to be high in speed of reaching the answer. So, in SCOFs setting the parameters is the important factor in reaching the optimized solutions. The parameters values are shown in Table (1).

Table 1: The value of the parameters

Algorithms	Parameters	Value
GA	Population Size	100
	Crossover Rate	0.80
	Mutation Rate	0.01
	Generations	100
	Elitism	0.20
AIS	Population Size	100
	No. Clone	50
	Selection	0.05
	Mutation Rate	80%
	Delete Rate	0.2
	Replace Rate	0.5

In EAs the size of the population is an affective factor in the final solution which is very important and when the population grows, the optimized solutions are reached and then the execution time of the algorithms grows and this is more clear in continuous optimization problems. So, in GA and AIS the body of the population size is defined according to the problems. In both algorithms the size of the population is considered 100.

The results of the simulations are reached after 100 repetitions. To limit the parameters change domain in a defined limitation, and to avoid entrance of the searching neighbors out of the changes domain of the parameters, the limits of the search space is considered constant in execution of the algorithm. Identification of the searching limit leads algorithms to be better in capability of finding the situation of the more accurate answer. In Table (2) the efficiency of the GA and AIS algorithms is compared to the PSO [11] and BFOA [11] algorithms.

Table 2: Comparison of the Results of the Algorithms

Functions	Range of Search Space	Algorithms			
		BFOA[11]	PSO[11]	GA	AIS
f_1	$[-20, 20]^D$	0.0001	0.0000	0.3326	0.5021
f_2		6.1968	0.0000	0.6257	0.6676

As it is seen in the Table (2), the GA in SCOFs is more efficient than AIS algorithm and is more capable in finding the optimized points. Also GA is more able to find the optimized solution in comparison to BFOA. In Figure (7) the comparison of the GA and AIS algorithms for f_1 function in getting the optimized solution is shown.

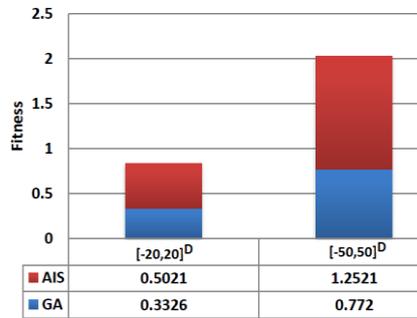


Figure 7: Comparison of GA and AIS for f_1 function

In Figure (8) the comparison of the GA and AIS algorithms for f_2 function is shown.

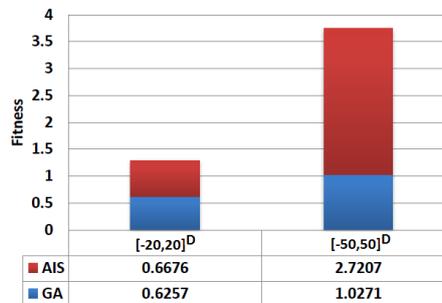


Figure 8: Comparison of the GA and AIS for f_2 Functions.

To show the efficiency of GA and AIS algorithms better, we have considered the search space bigger in Table (3). The reason is that the continuous functions have many optimized local points and also the values of the cost functions are very close to each other in these points.

Table 3: Comparison of the Results of Algorithms

Functions	Range of Search Space	Algorithms			
		BFOA[11]	PSO[11]	GA	AIS
f_1	[-50 , 50] ^D	0.0051	0.0000	0.7720	1.2521
f_2		15.1075	0.0000	1.0271	2.7207

In the Figure (9) the comparison of the GA, AIS and BFOA algorithms in reaching the optimized solution is shown. As it is clear, GA is more efficient in AIS and BFOA algorithms.

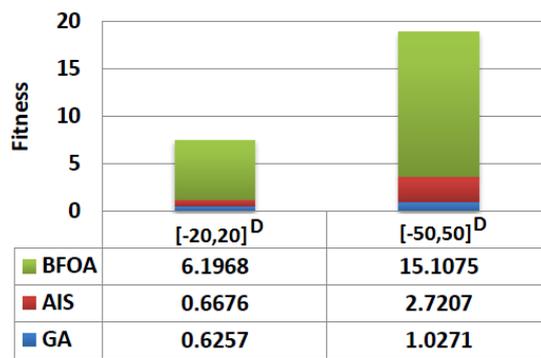


Figure 9: Comparison of the GA, AIS and BFOA Algorithms for f_2 Functions.

As it is seen, GA algorithm is more efficient than AIS in SCOFs of larger dimensions and is more able in finding the better optimized points. SO, GA is able to maintain the equilibrium of the search operands in an optimized manner even by increase in the dimensions of the problem.

6. DISCUSSION

Two exploration and exploitation concepts are very important in GA. In fact the three Crossover, Mutation and Selection operands are very important in searching of GA. If exploration is considered as searching the new spaces of searching space using the Crossover and Mutation operands, and the exploitation as the selection operand, we must know that exploration will more lead to the random search and lack of convergence and the exploitation will more lead to local search and convergence to the optimized local answer. So, these operands are very affective in GA.

The most important subject which in EAs causes the slowing of finding the optimized solutions is the repetitions and it is in fact the main loop of generation production for the new population. It must be added to this that any repetition of the ring is very time consuming (because of crossover, mutation and sorting the chromosomes). The value of these algorithms is hidden in this fact that they do not use these operands in many times. In other words, AIS algorithm is not based on the crossover operand and this is why AIS is faster than GA but not more efficient.

Each stage of GA includes a population of the chromosomes which is more fitted than the previous stage. Each population or a generation of the chromosomes is called the number of the

population. The population number shows the number of the chromosomes existing in a population or generation. If the number of the chromosomes is very low, the possibility of reaching the optimized answer is very low and just a little space will be searched. Then the number of the population affects highly the execution time of the algorithm and reaching the optimized answer.

Different factors are effective in convergence speed of GA to the global optimized answer. The primary population, mutation and combination probability values, the combination and mutation method, fitness function and the selection of the next population are some of these factors. The effect of these factors in convergence of the GA depends on the problem and is reached by testing.

One of the advantages of AIS algorithm in comparison to GA is the variability of mutation probability in AIS algorithm and also it being constant in GA. In AIS the probability of mutation in defensive cells with lower dependency is more than the cells with more dependency. So, the probability of changes in structure in defensive cells with suitable operation is lower than the probability of changing the defensive cells of non-suitable operation. In GA the mutation rate is considered constant for all individuals of a population.

AIS algorithm is similar to the GA from this point that they both produce a population of the answers in a random way and go through the problem searching for the optimized answer. In AIS any optimized answer of one factor is called affinity in contrast to GA. In any repetition each anti-body mutates according to the affinity of the answers in problem space. So, in AIS algorithm the mutation is more effective in number accuracy of the final solution and makes AIS algorithm find the global optimized points with high accuracy.

AIS algorithm includes the spaces of problem which contain the local and global optimized points like GA at the first steps but continuing the convergence path to the global optimization and the optimization procedure, it is stopped. In other words the decrease in genetic diversity in evolution process causes the searching process of the problem be interrupted. The second general problem of AIS algorithm is the non-stability of these algorithms in different execution processes. The random nature of the algorithm causes diversity of the quality of the responses in different execution processes. This problem causes lack of reliability in the responses of the AIS algorithm.

7. CONCLUSION AND FUTURE WORKS

Searching the space of the continuous optimization functions using the EAs leads to this that the most optimized solutions are selected among all possible solutions. EAs in SCOFs are very efficient in getting close to the optimized solutions. These algorithms take into consideration the structure and the complexity type of the problems, and study the continuous optimization problems using the testing method and searching the optimized points and lead to accessing the close to the optimized answers. In this paper, GA and AIS algorithms are analyzed for solving the continuous optimization problems which are utilized in close to optimized answers. To evaluate the efficiency of the algorithms, two types of comparison from response accuracy and convergence to the different spaces points of view are done and then the results show that the probability of convergence in solving the continuous optimization problems and also reaching the optimized answer is more in GA from AIS. GA is more flexible for solving such problems and is faster in response in getting convergent in comparison to AIS. This result is not generalizable to other problems and such judgment is not right for all optimization problems. So in SCOFs, GA is more accurate reaching the global optimized points than AIS. In this paper we hope that using the

other EAs will be able to analyze the continuous optimization problems of higher dimensions and more optimized points.

REFERENCES

- [1] K. Deb, J. Sundar, N.U.B. Roa, S. Chaudhuri, “*Reference Point based Multi-Objective Optimization using Evolutionary Algorithms*”, International Journal of computational Intelligence Research, Vol. 2, No. 3, pp. 273-286, 2006.
- [2] N.D. Lagaros, M. Papadrakakis, G. Kokossalakis, “*Structural Optimization using Evolutionary Algorithms*”, Computers and Structures, Vol. 80, pp. 571-589, 2002.
- [3] J. Holland, “*Adaptation in Natural and Artificial Systems*”, University of Michigan, Michigan, USA, 1975.
- [4] F.S. Gharehchopogh, I. Maleki, M. Farahmandian, “*New Approach for Solving Dynamic Traveling Salesman Problem with Hybrid Genetic Algorithms and Ant Colony Optimization*”, International Journal of Computer Applications (IJCA), Vol. 53, No.1, pp. 39-44, September 2012.
- [5] F.S. Gharehchopogh, I. Maleki, B. Zebardast, “*A New Approach for Solving N-Queens Problem with Combination of PMX and OX Crossover Operators*”, Elixir International Journal Computer Science and Engineering (Elixir Comp. Sci. & Engg.), Vol. 61, pp. 16650-16654, July 2013.
- [6] J. Farmer, N. Packard, A. Perelson, “*The Immune System, Adaptation and Machine Learning*”, Physica D, Vol. 2, pp. 187-204, 1986.
- [7] L. de Castro, J. Timmis, “*Artificial Immune Systems: A New Computational Approach*”, Springer-Verlag, London, UK, September 2002.
- [8] J. Shen, J. Wang, H. Ai, “*An Improved Artificial Immune System based Network Intrusion Detection by Using Rough Set*”, Communications and Network, Vol. 4, pp. 41-47, 2012.
- [9] M. Guezouri, A. Houacine, “*Hybrid Flow Shop Scheduling Problem Using Artificial Immune System*”, International Journal Intelligent Systems and Applications, Vol. 10, pp.82-88, 2012.
- [10] M. Sanei, N.M. Charkari, “*Hybrid Heuristic-Based Artificial Immune System for Task Scheduling*”, International Journal of Distributed and Parallel Systems (IJDPS), Vol.2, No. 6, pp. 1-12, November 2011.
- [11] F.S. Gharehchopogh, I. Maleki, B. Zebardast, “*A New Solutions for Continuous Optimization Functions by using Bacterial Foraging Optimization and Particle Swarm Optimization Algorithms*”, Elixir International Journal Computer Science and Engineering (Elixir Comp. Sci. & Engg.), Vol. 61, pp. 16655-16661, July 2013.
- [12] S.R. Khaze, I. Maleki, S. Hojjatkah, A. Bagherinia, “*Evaluation the Efficiency of Artificial Bee Colony and the Firefly Algorithm in Solving the Continuous Optimization Problem*”, International Journal on Computational Science & Applications (IJCSA), Vol. 3, No. 4, pp. 23-35, August 2013.
- [13] N. Gupta, G.S. Gaba, H.S. Gill, “*A New Approach for Function Optimization using Hybrid GA-ANN Algorithm*”, Engineering Research and Applications (IJERA), Vol. 2, Issue 2, pp.386-389, 2012.
- [14] N. Kushwaha, V.S. Bisht, G. Shah, “*Genetic Algorithm based Bacterial Foraging Approach for Optimization*”, National Conference on Future Aspects of Artificial intelligence in Industrial Automation (NCFAAIIA), pp. 11-14, 2012.
- [15] X. Lai, “*Ensembles of GA and PSO for Real Function Optimization*”, Journal of Information & Computational Science, Vol. 7, No. 3, pp. 667-675, 2010.
- [16] K.A. Al-Sheshtawi, H.M. Abdul-Kader, N.A. Ismail, “*Artificial Immune Clonal Selection Algorithms: A Comparative Study of CLONALG, opt-IA, and BCA with Numerical Optimization Problems*”, International Journal of Computer Science and Network Security, Vol. 10, No. 4, pp. 24-30, April 2010.
- [17] L. Zhou, J. Zheng, “*A New Immune Clone Algorithm to Solve the Constrained Optimization Problems*”, WSEAS TRANSACTIONS on COMPUTERS, Vol. 10, Issue 4, pp. 105-114, April 2011.
- [18] R. Patel, M.M. Raghuwanshi, “*Multi-Objective Optimization Using Multi Parent Crossover Operators*”, Journal of Emerging Trends in Computing and Information Sciences, Vol. 2 No. 2, pp. 99-105, 2010.
- [19] L.N. de Castro, F.J.V. Zuben, “*Learning and Optimization using the Clonal Selection Principle*”, IEEE Transactions on Evolutionary Computation, Vol. 6, No. 3, pp. 239-251, 2002.
- [20] L.N. de Castro, J. Timmis, “*An Artificial Immune Network for Multimodal Function Optimization*”, In Proceedings of the 2002 Congress on Evolutionary Computation (CEC'2002), Vol. 1, pp. 669-674, Honolulu, Hawaii, May 2002.

Authors

Isa Maleki is a Lecturer and Member of The Research Committee of The Department of Computer Engineering, Dehdasht Branch, Islamic Azad University, Dehdasht, Iran. He Also Has Research Collaboration with Dehdasht Universities Research Association NGO. He is a Member of Review Board in Several National Conferences. His Interested Research Areas Are in the Software Cost Estimation, Machine Learning, Data Mining, Optimization and Artificial Intelligence.



Marjan Mahmoodi Tabrizi is a M.Sc. student in Department of Computer Engineering, Science and Research Branch, Islamic Azad University, West Azerbaijan, Iran. Her interested research areas are Wireless Sensor Networks, Data Mining, Optimization and Machine Learning.



Seyyed Reza Khaze is a Lecturer and Member of the Research Committee of the Department of Computer Engineering, Dehdasht Branch, Islamic Azad University, Dehdasht, Iran. He is a Member of Editorial Board and Review Board in Several International Journals and National Conferences. His interested research areas are in the Software Cost Estimation, Machine learning, Data Mining, Optimization and Artificial Intelligence.



Seyyed Amir Reza Abedini is a lecturer and member of the Research Committee of the Department of Computer Engineering, Dehdasht Branch, Islamic Azad University, Dehdasht, Iran. He Also Has Research Collaboration With Dehdasht Universities Research Association NGO. His Interested Research Areas Are in the Wireless Sensor Networks, Data Mining, Optimization and Artificial Intelligence.

