International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:3, No:1, 2009

A New Method for Multiobjective Optimization Based on Learning Automata

M. R. Aghaebrahimi, S. H. Zahiri, and M. Amiri

***Abstract*—**The necessity of solving multi dimensional complicated scientific problems beside the necessity of several objective functions optimization are the most motive reason of born of artificial intelligence and heuristic methods.

In this paper, we introduce a new method for multiobjective optimization based on learning automata. In the proposed method, search space divides into separate hyper-cubes and each cube is considered as an action. After gathering of all objective functions with separate weights, the cumulative function is considered as the fitness function. By the application of all the cubes to the cumulative function, we calculate the amount of amplification of each action and the algorithm continues its way to find the best solutions. In this Method, a lateral memory is used to gather the significant points of each iteration of the algorithm. Finally, by considering the domination factor, pareto front is estimated. Results of several experiments show the effectiveness of this method in comparison with genetic algorithm based method.

***Keywords*—**Function optimization, Multiobjective optimization, Learning automata.

I. Introduction

BY exposure of the many complicated multidimensional engineering problems and the necessity of different objective functions simultaneously, the weakness of derivative based methods is appeared. For this reason, researchers tried to invent calculation and optimization methods to solve the problems without use of derivative. The result of these researches is new optimization methods known as bio-optimization methods. These methods are based on natural and unique phenomenon. Despite of simple structure, these methods are very powerful. Genetic algorithm (GA), particle swarm optimization (PSO) and ant colony are the main samples of these methods. ([1]-[4]) Learning automata (LA) is an intensive learning method that showed its ability in different applications. ([5]-[7])

In this research, we use LA to optimize a function and the result of this idea is introduction of a new method for multiobjective optimization that can be compared with other famous optimization methods like GA and PSO. Vast review on the literature showed that recently, LA is used to solve the single objective problems. ([8],[9])

M. R. Aghaebrahimi is with the University of Birjand, Birjand, Iran (98-561-2227044; fax: 98-561-2227795; e-mail: Aghaebrahimi@ Birjand.ac.ir).

S. H. Zahiri is with the University of Birjand, Birjand, Iran (e-mail: Hzahiri@ Birjand.ac.ir).

M. Amiri is with the University of Birjand, Birjand, Iran (e-mail: M. Amiri@ Birjand.ac.ir).

In this paper, we use LA in multiobjective optimization problems. In the proposed method, after the collection of objective functions (in different number and weights) as a cumulative function, search space is divided into separate hyper-cubes. Each hyper-cube is considered as an action and in a successful designed search process the probability density function of them is encouraged. Less important cubes are eliminated of tournament and cubes that their probability density factor are more encouraged goes under research by a higher probability. Finally, when the volume of the cubes becomes decreased less than a specified amount, the average of them is considered as an answer. This process continues by dynamic change of the weights and at the end we have a front of solutions is called pareto front.

The rest of this paper organized as follow:

Section II consists of some explanations of LA based single objective function optimization. In section III, we introduce the method of LA based multiobjective optimization. Section IV consists of the results of the vast experiences of application of our proposed method on different famous functions. Also, we compare the result of this method by the GA based method in this section. Finally, section V concludes the paper.

II. LA Based Function Optimization

Among the LA based single objective function optimization methods, the one introduced by Zegn and Lui is one of the most powerful. In order to minimize function $f($, that in which $X = (x_1, x_2, \ldots, x$ is the n-dimensional variable of the function. divides into hyper-cubes and each of them considers as an action. After that, we do randomly examination of inner points of cubes. In accordance with the ability of each point to decrease the amount of function, the amount of encourage or penalty in different action probabilities is determined. Also, the actions or cubes have amount of probability less than determined threshold (a percent of whole amount obtained until that time), are eliminated and replaced by new actions. Fig. 1 shows this method in a brief.

III. LA Based Multiobjective Function Optimization

In this section we introduce the LA based proposed method for multiobjective optimization.

Step 1: Initialization of internal parameters

r: Number of hypercubes which divide the feature space

δ : Threshold of action probabilities

s: Normalized factor of convergence

ε : Error band

Step 2: Divide the feature space into *r* hypercubes

Each hypercube is supposed as an action denoted by $\alpha_i, i \in \{1,2,\ldots,r\}$.

Corresponding to each action there are some parameters introduced as below:

$\eta_i(n)$: Total reward obtained by the action α_i until nth sampling instant.

$z_i(n)$: Number of times the action α_i is chosen until nth sampling instant.

$$\xi_i(n) = \frac{\eta_i(n)}{z_i(n)},$$

$$\xi_m(n) = Max_i\{\xi_i(n)\}$$

$$\xi_l(n) = Min_i\{\xi_i(n)\}$$

$p(n)$: Action probability distribution of α_i at nth sampling instant with initial value of $\frac{1}{r}$.

Step 3: Search loop

Repeat

- Pick up an action $\alpha(n) = \alpha_i(n)$ according to $p(n)$.
- Randomly select $X = (x_1, x_2, \ldots, x_n)$ from the hypercube corresponding to α_i.
- Calculate $f(X)$.
- Update $\xi_i(n)$ as follows:

If $\alpha(n) = \alpha_i$, Then

$$\eta_i(n+1) = \eta_i(n) + \frac{M - f(X)}{M - L},$$

Where M and L are the estimated upper and lower of $f(x)$ respectively, permitting to normalize the estimates of reward probabilities to the interval [0, 1].

$$z_i(n+1) = z_i(n) + 1,$$

$$\xi_i(n+1) = \frac{\eta_i(n+1)}{z_i(n+1)},$$

For all $j \neq i$

$$\eta_j(n+1) = \eta_j(n),$$

$$z_j(n+1) = z_j(n),$$

$$\xi_j(n+1) = \xi_j(n)$$

- Update $p(n)$ as follows:

$$p(n+1) = (1 - s * \xi_m(n)) * p(n) + s * \xi_m(n) * e_m,$$

;(e_m is a r-dimensional vector with mth component unity and all others zero)

- If $p_l(n) = Min_i\{p_i(n)\} < \delta$, Then

Go to the next step,

- Else

n=n+1,

End Repeat;

Step 4: Enhancing the search process in *l*th hypercube.

Fig.1. The schedule of the function optimization based on the learning automata.

A. *The Main Contexts of Multiobjective Optimization*

Suppose that is n-dimensional desired search space and $f_i($ is the objective function, multiobjective optimization problem can be defined as bellow:

$$\text{Minimize } \hat{f}(x) = [f_1(x), f_2(x), \ldots, f_k(\quad (1)$$

If: $g_j(x) \leq 0 \;, \; j = 1,2, \ldots,$

In above formula, $\hat{f}($ is objective functions vector and $g_j($ is the jth condition of the problem. In many cases objective functions are in conflict. It means that if point is optimum solution of one of the objective functions, this point is not the optimum solution of other functions and maybe the worst solution of them.

The main aim of the algorithm of multiobjective optimization is to find pareto front. This front is composed of a set of non-dominated like . The concept of domination is defined as bellow:

If $a = (a_1, a_2, \ldots,$ and $b = (b_1, b_2, \ldots, b$ were two points of search space, we say a dominates b, if and only if: $f_i(a) \leq f_i(b) \;, \; i = 1,2, \ldots$ and also at least for a $m \in \{1,2, \ldots,$, we must have: $f_m(a) \leq f_m($.

The proposed algorithm is aimed to find the pareto front. We fully explain this front in next subsection.

B. *The Structure of Proposed Method*

One of the most famous methods of multiobjective optimization is "weighted aggregation". In this method, a linear combination of objective functions with non-negative different weights forms "aggregated function" as bellow:

$$F(x) = \sum_{i=1}^{k} w_i \cdot f_i(x) \;, \; i = 1,2, \ldots, k$$

$$\sum_{i=1}^{k} w_i = \quad (2)$$

In above formula $F($ is aggregated function and is ith non-negative weight related to ith objective function.

There are different methods for aggregation of objective function and producing aggregated function $F($. Some of these methods are: "conventional weighted aggregation" method (CWA), "big-bang weighted aggregation" method (BWA) and "dynamic weighted aggregation" method (DWA). Each of them has advantages and disadvantages.

In CWA method, weights are constant and by running the algorithm each time, the result is just one point of pareto front. Because of this, calculation size is high. Also, this method can't estimate concave pareto fronts. The ability of solving just two objective functions is the most important limitation of DWA method.

Best method of estimation of aggregation of functions is DWA, because this method shows better ability to estimate concave pareto fronts [10]. This method is defined for two objective functions as below (It can be generalized easily for more than two objective functions):

$$w_1(t) = \left|\sin\left(\frac{2\pi t}{v}\right)\right|$$

$$w_2(t) = 1 - w_1 \quad (3)$$

is the iteration index of algorithm and is the weights changes frequency.

Because of the good functionality of DWA method, we use this method in this paper.

After definition of primary weights by (3) and aggregation of objective functions by (2), we apply the aggregated function to the proposed algorithm (figure 1) and after each iteration, five optimum points are saved in lateral memory. Finally, after running the algorithm as supposed iteration, we examine the memory and we choose non-dominated points as pareto front.

IV. Test Results

In this section, the results of implementation of proposed method on three famous problems that have different pareto fronts in monotonousness, convexity and concavity are depicted. Also, we do a comparison between the results of the proposed method and the GA-based method.

These famous problems are minimization of bellow functions:

a) Function $\vec{f_1}(x) = (f_1(x), f_2(x))$, that has a convex and monotonous pareto front.

$$f_1(x) = \frac{1}{n}\sum_{i=1}^{n} x_i^2 \quad , \quad f_2(x) = \frac{1}{n}\sum_{i=1}^{n} (x_i - 2)^2 \quad (4)$$

b) Function $\vec{f_2}(x) = (f_1(x), f_2(x))$, that has a convex and no monotonous pareto front.

$$f_1(x) = x_1 \quad , \quad f_2(x) = g(x)\left(1 - \sqrt{\frac{f_1(x)}{g(x)}}\right)$$

$$g(x) = 1 + \frac{9}{n-1}\sum_{i=1}^{n} x_i \quad (5)$$

c) Function $\vec{f_3}(x) = (f_1(x), f_2(x))$, that has a concave pareto front.

$$f_1(x) = x_1 \quad , \quad f_2(x) = g(x)\left(1 - \left(\frac{f_1(x)}{g(x)}\right)^2\right)$$

$$g(x) = 1 + \frac{9}{n-1}\sum_{i=1}^{n} x_i \quad (6)$$

Fig. 2, Fig. 3 and Fig. 4 show the result of tests on three famous functions mentioned before.

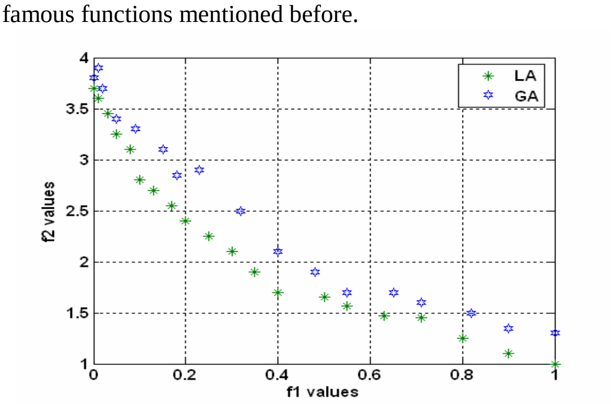

Fig. 2 The results of optimization of $\vec{f_1}$ by LA-based and GA-based methods

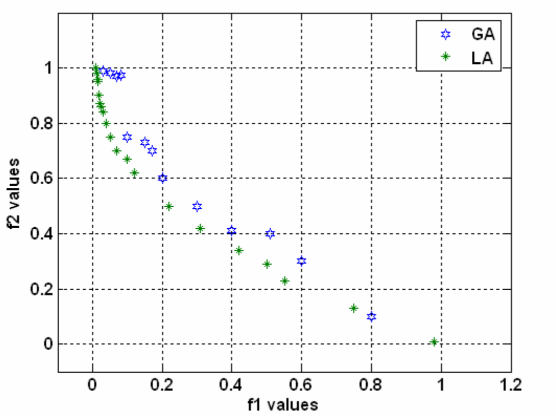

Fig. 3 The results of optimization of $\vec{f_2}$ by LA-based and GA-based methods

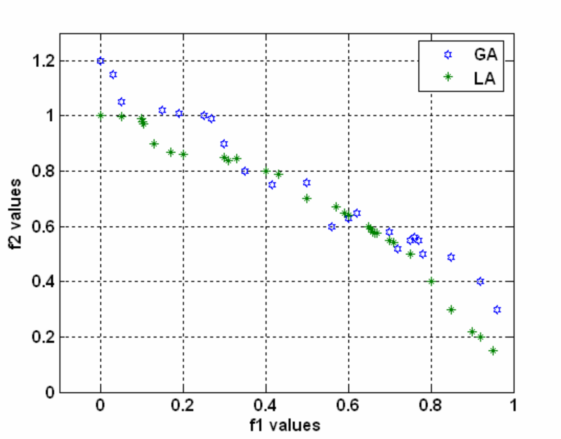

Fig. 4 The results of optimization of $\vec{f_3}$ by LA-based and GA-based methods

As it can be seen in Fig. 2, the proposed algorithm has a better functionality in comparison with GA-based method and pareto front gained by LA for two objective function $\vec{f_1}$, is dominant in comparison with LA. In the case of $\vec{f_2}$ with the exception of some points LA has better results than GA (Fig. 3).

By comparison between two algorithms in Fig. 3, it can be seen that the result points of LA has a more monotonous distribution in comparison with GA. This characteristic is one of the noticeable characteristics of a multiobjective optimization method. By reviewing Fig. 3 better functionality and monotonous distribution of points of LA in comparison with GA is obvious.

V. Conclusion

In this paper, a new method for multiobjective optimization based on LA is proposed. In this method, the entire objective functions with dynamic weights form an aggregated function. The best point of each iteration of algorithm collects in a lateral memory and after gathering enough solutions, by examining dominance factor, best solutions form pareto front.

Results of implementation of proposed method on three famous multiobjective optimization problems show the ability of the method. In some cases the functionality of proposed method is better than GA-based algorithm. Although better results this method is more complicated in comparison with GA-based method and this topic can be one of the future researches in this field.

REFERENCES

[1] C.A.C. Coello, "A comprehensive survey of evolutionary-based multiobjective optimization techniques," Knowledge and Information Science 1 (1999) 269-308.
[2] C.M. Fonseca, P.J. Fleming, "An overview of evolutionary algorithms in multiobjective optimization," Evolutionary Computation 3 (1995) 1-16.
[3] E. Zitzler, "Evolutionray algorithms for mulriobjective optimization: methods and applications," Ph.D. Tesis, Swiss Federal Institute of Technology, Zurich, 1999.
[4] M. Reyes-Sierra, Coello C.A.C., "Multi-objective Particle Swarm Optimizers: A Survey of State-of-the-Art," Intl. J. of Copmut. Intell. Res. (3) (2006) 287-308.
[5] B.J. Oommen, E.V. de St. Criox, "Graph partitioning using learning automata," IEEE Trans. Comput. 45 (1996) 195-208.
[6] X. Zeng, J. Zhou, C. Vasseur, "A strategy for controlling non-linear systems using a learning automaton," Automatica 36 (2000) 1517-1524.
[7] C.K.K. Tang P. Mars, "Games of stochastic learning automata and adaptive signal processing," IEEE Trans. Syst., Man, Cyber. 23 (1993) 851-856.
[8] X. Zeng, Z. Liu, "A learning automaton based algorithm for optimization of continuous complex function," Inform. Sci. 174 (2005) 165-175.
[9] H. Beygi, M.R. Meybodi, "A new action-set learning automaton for function optimization," Int. J. of the Franklin Inst. 343 (2006) 27-47.
[10] Y. Jin, M. Olhofer, and B. Sendho,"Dynamic weighted aggregation for evolutionrary multiobjective optimization: Why dose it work and how?" In Proc. GECCO 200.
[11] E. Zitzler, K. Deb, and L. Thiele,"Comparison of multiobjective evolution algorithms: empirical results," Evolutionary Computation, 8(2): 173-195, 2000.

M. R. Aghaebrahimi was born in Iran. He is an Assistant Professor with the Department of Power Engineering, University of Birjand, Iran. He received his B.Sc. in electrical engineering from Mashhad Ferdowsi University, Iran, in 1989. He also received his M.Sc. and Ph.D. in electrical engineering from University of Manitoba, Canada, in 1993 and 1997, respectively. His areas of interest include HVDC Transmission Systems and Electric Machinery. He is currently involved in research in the area of Renewable Energy Resources, especially Wind Energy.

S. H. Zahiri was born in Iran. He is an Assistant Professor with the Department of Power Engineering, University of Birjand, Iran. He received the B.Sc., M.Sc. and Ph. D. degrees in electronics engineering from the Sharif University of Technology, Tehran, Iran, Tarbiat Modarres University, Tehran, and Ferdowsi University of Mashhad, Mashhad, Iran, in 1993, 1995and 2005, respectively. His research interests include pattern recognition, evolutionary algorithms, and swarm intelligence algorithms.

M. Amiri was born in Iran. He received his B.Eng. from Power and Water University of Technology (PWUT), Tehran, Iran, in 2003. He is currently a student of M.Sc. at department of power engineering, University of Birjand, Iran. His areas of interest include: voltage stability, distributed generation (DG), Reactive power control and Artificial Intelligence.