A New Hybrid Algorithm Based on Firefly Algorithm and Cellular Learning Automata

Tahereh Hassanzadeh*, Mohamad Reza Meybodi**
*Qazvin Azad University, t.hassanzadeh@qiau.ac.ir
**AmirKabir University of Technology, mmeybodi@aut.ac.ir

Abstract: *In this paper, a new evolutionary optimization model, called CLA-FA, is proposed. This new model is a combination of a model called cellular learning automata (CLA) and the Firefly Algorithm (FA). In the proposed algorithm, at first we modify the firefly algorithm to improve the efficiency of this algorithm then we use this algorithm with CLA. in the proposed algorithm, each dimension of search space is assigned to one cell of cellular learning automata and in each cell a swarm of fireflies are located which have the optimization duty of that specific dimension. The learning automata in each cell are responsible for making diversity in fireflies' swarm of that dimension and adapting the FA parameters for equivalence between global search and local search processes. In order to evaluate the proposed algorithm, we used five well known benchmark function, including: Sphere, Ackly Rastrigin, Xin-she yang and Step functions in 10, 20 and 30 dimensional spaces. The experimental results show that our proposed method can be effective to find the global optima and can improve the global search and the exploration rate of the standard firefly algorithm.*

Keywords: Firefly algorithm, Cellular learning automata, Optimization, Global search, Local search.

1. Introduction

Collective intelligence is a type of artificial intelligence that based on collective behavior of agents in decentralized and self-organized systems. These systems are usually organized from simple agents that cooperate locally with other elements and environments. However, there is not any centralized mechanism on agents' behavior. So, nature inspired metaheuristic algorithms are becoming more powerful algorithms for solving optimization problems [1-5].

There are different methods for optimization. Particle Swarm Optimization (PSO) [6], Ant Colony Optimization (ACO) [7], Artificial Fish Swarm Algorithm (AFSA) [8] and Bee Colony [9] are the most well-known algorithms have ever been proposed for optimization. By idealizing some of the flashing characteristics of the fireflies, Firefly Algorithm was recently introduced by XIN-SHE YANG at Cambridge University [10]. This swarm intelligence optimization technique is based on the assumption that solution of an optimization problem can be shown as a firefly which glows proportionally to its quality in a considered problem setting. Consequently, each brighter firefly attracts its partners, which makes the search space being explored efficiently. Yang used the FA for nonlinear design problems [11] and multimodal optimization problems [12] and showed the efficiency of the FA for finding global optima in two dimensional environments. YANG also introduced a new version of FA (Levy FA), which combined Levy flight with the search strategy via the firefly for improving the randomization of FA [13]. Discrete firefly algorithm also introduced recently by SAYADI [14].

In [15], Cellular learning automata (CLA) which a combination of the cellular automata (CA) and learning automata (LAs) is introduced. This model is superior to CA because of its ability to learn and also is superior to single LA because it is a collection of LAs, which can interact with each other toward solving a particular problem. The basic idea of CLA, which is super class of stochastic CA, is to use learning automata to adjust the state transition probability of stochastic CA.

In this paper, at first we proposed a new modify of firefly algorithm and then a new algorithm called CLA-FA is proposed in that with use of CLA the efficiency of FA was improved. In the proposed algorithm, a cell of CLA is assigned to each dimension of problem space that has a population of fireflies. Each population has the responsibility of optimizing one dimension. Available learning automata in each cell is controlling the parameter of firefly of that cell and avoiding of premature convergence. Each learning automata by using the experiences of neighbor cells determines its form of cell's fireflies' movement.

The rest of the paper is organized as follows: Section 2 describes the Firefly Algorithm and the new modify algorithm. Section 3 gives a brief detail of learning automat and cellular learning automata Section 4 gives a detailed description of the proposed method. The experimental results are discussed in Section 5, and Section 6 concludes the paper.

2. Cellular Automata and Cellular Learning Automata

2.1 Cellular Automata

Cellular Automata (CA) are discrete dynamic systems whose behavior is completely based on Local relations. A cellular automaton consists of a grid of cells; each of them is in one of the finite number of states. In CA, the time is also discrete, and the state of a cell is a function of the previous states of its neighbor cells. A uniform rule is applied to each cell and its neighbours and each time it is applied, the new states of cells are generated [16].

2.2 Cellular Learning Automata

The Cellular Learning Automaton (CLA) is a mathematical model for modeling dynamics of a complex system which consists of a large number of simple components. In fact a cellular learning automaton is a CA in which every cell is equipped with at least one learning automaton. The Learning Automaton (LA) has a finite set of actions and its goal is to learn which action in this set is the optimal action. Like CA, there is a uniform local rule applied to the cells and based on this rule the selected action gets a reward or a penalty. If the learning algorithm is chosen properly, the iterative process of interacting with the environment can result in selection of the optima action in every cell [15].

3. Firefly Algorithm

3.1 Standard Firefly Algorithm

Fireflies are one of the most special creatures in nature. Most of fireflies produced short and rhythmic flashes and have different flashing behavior. Fireflies use these flashes for communication and attracting the potential prey. YANG used this behavior of fireflies and introduced Firefly Algorithm in 2008 [10].

In Firefly algorithm, there are three idealized rules: 1) All fireflies are unisex. So, one firefly will be attracted to other fireflies regardless of their sex; 2) Attractiveness is proportional to their brightness. Thus, for any two flashing fireflies, the less brighter one will move towards the brighter one. The attractiveness is proportional to the brightness and they both decrease as their distance increases. If there is no brighter one than a particular firefly, it will move randomly; 3) The brightness of a firefly is affected or determined by the landscape of the objective function. For a maximization problem, the brightness can simply be proportional to the value of the objective function [10]. The pseudo code of these three rules can be shown as Fig. 1.

In the standard firefly algorithm, each firefly in each compared with all other fireflies and the less brighter firefly moves toward the brighter one (in the maximization optimization). In the FA there are two important issues for variation of light intensity and the formulation of the attractiveness. For simplicity, it's assumed that the attractiveness of a firefly is determined by its brightness which associated with the objective function of the optimization problem. Since a firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the attractiveness of a firefly by:

$$\beta(r) = \beta_0 e^{-\gamma r^2} \quad \textbf{(1)}$$

where, β_0 is the attractiveness at $r = 0$ and γ is the light absorption coefficient at the source. It should be noted that the $r_{i,j}$ which is described by equation 2, is the Cartesian distance between any two fireflies *i* and *j* at x_i and x_j, where, x_i and x_j are the spatial coordinate of the fireflies *i* and *j*, respectively.

$$r_{i,j} = \left\| x_i - x_j \right\| = \sqrt{\sum_{k=1}^{d} \left(x_{i,k} - x_{j,k} \right)^2} \quad \textbf{(2)}$$

The movement of a Firefly *i*, which is attracted to another more attractive Firefly *j* is determined by:

$$X_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} \left(x_j - x_i \right) + \alpha \left(rand - \frac{1}{2} \right) \quad \textbf{(3)}$$

where, the second term is the attraction while the third term is randomization including randomization parameter α and the random number generator *rand* which its numbers are uniformly distributed in interval [0, 1].

For the most cases of implementations, $\beta_0 = 1$ and $\alpha \in [0, 1]$. The parameter γ characterizes the variation of the attractiveness and its value is important to determine the speed of the convergence and how the FA behaves. In the most applications, it typically varies from 0.01 to 100.

Firefly algorithm
Initialize algorithm parameters:
MaxGen: the maximum number of generations
*Objective function of f(**x**), where **x**=(x1,........,xd)T*
*Generate initial population of fireflies or **x**i (i=1, 2,..., n)*
*Define light intensity of **I**i at **x**i via f (**x**i)*
While *(t<MaxGen)*
  For *i = 1 to n (all n fireflies);*
    For *j=1 to n (all n fireflies)*
  If *(Ij > Ii), move firefly i towards j;* ***end if***
Attractiveness varies with distance r via Exp [-γr2];
Evaluate new solutions and update light intensity;
    End for j;
  End for i;
Rank the fireflies and find the current best;
End while;
Post process results and visualization;
End procedure;

Fig1. Pseudo code of the FA.

3.2 Modified Firefly Algorithm

In the standard Firefly Algorithm (FA), in each iteration the brighter firefly (local optima) exerts its influence over other fireflies and attracts them towards itself in maximization optimization. In fact, in the standard FA, fireflies move regardless of the global optima and it can increase the number of iteration to find the global best and decrease exploration ability of the firefly algorithm. In this paper, to eliminate weaknesses of FA and improve the collective movement of fireflies, we propose a modify firefly algorithm (MFA). In the proposed firefly algorithm, we use global optima in firefly's movement. Global optimum is related to optimization problem and it can be a firefly that has the maximum or minimum value. And the global optima will be update in any iteration of algorithm. In the proposed algorithm , when a firefly compared with other firefly instead of the one firefly in the neighborhood being allowed to influence and to attract its neighbors, global optima (a firefly that have maximum or minimum value) in each iteration can be allowed to influence others and affect in their movement.in the MFA, when a firefly compare with correspond firefly , if the correspond firefly be brighter , the compared firefly will move toward correspond firefly, considered by global optima.

In the MFA, we use Cartesian distance to compute the distance of fireflies to global optima same as the standard FA as it shows in equation 4,

$$r_{i,best} = \sqrt{\left(x_i - x_{gbest}\right)^2 + \left(y_i - y_{gbest}\right)^2} \qquad (4)$$

But for movement of the firefly, equation (3) is replaced by:

$$x_i = x_i + \left(\beta_0 e^{-\gamma r_{ij}^2}\left(x_j - x_i\right) + \beta_0 e^{-\gamma r_{i,gbest}^2}\left(x_{gbest} - x_i\right)\right) + \alpha\left(rand - 1/2\right) \qquad (5)$$

g_{best} is global optimal and x_{gbes} is the coordinate of global optima.

4. Proposed CLA-FA Algorithm

In this section, the firefly algorithm based on cellular learning automata called CLA-FA is proposed which in that both ability of local search and ability of global search to the standard FA has been increased. In this section, we use modify firefly algorithm that discussed above. In the proposed algorithm, there are swarms of one dimensional fireflies equal to the number of search space dimensions that each swarm is placed in a cell of cellular learning automata. Each cell is equipped with a learning automaton that determines and adjusts the corresponding next movement of fireflies.

In the standard FA, for any two flashing fireflies, the less bright one will move towards the brighter, which includes a percentage of randomness. The initial percentage value and other parameter such as light absorption coefficients are crucially important in determining the speed of the convergence and how the FA algorithm behaves. In the standard FA, the parameters are fixed from beginning to end. Alpha determines random percentage in firefly's movement. Alpha is between zero to one. Light absorption coefficients or Gamma is varies between zero to extreme. If Gamma is close the zero, then $\beta_0=\beta$ and this corresponds to a special case of PSO. Besides, if Gamma is close extreme, this is the case where the fireflies fly in a very foggy region randomly.

In the standard FA, determination of initial value of these two parameters has a remarkable effect on the final result. The values of these two parameters remain constant and equal to their initial value up to the end of performing the algorithm. If we consider small values for these two parameters, algorithm can do local search with more stability and accuracy, but in this condition fireflies move slower toward goal Thus, it is necessary for improving results, a balance made between global search and local search process so the algorithm could perform both acceptably that this is done in the proposed algorithm.

The CLA-FA performance is as follow: instead of using of one swarm with n D-dimensional fireflies, we use D swarm of n one-dimensional fireflies. The objective of each swarm is to optimize a component of optimization vector which corresponds to that swarm. To achieve the desirable result, all these swarms should cooperate with each other. In order to cooperate, the best firefly of each swarm would be chosen. The vector that is totally being optimized for all swarms is called "main vector". Each swarm introduces it's the best firefly as representative of that dimension. In fact, main vector which is D-dimensional vector consists of the values of the best firefly of each of D swarms. In each iteration of algorithm performance, for fireflies of swarm *i*, just the element of that *i*-th dimension of main vector is changed. Indeed, for cooperation among swarms, for calculating the fitness of a firefly in *j*-th swarm, vector value for that firefly is replaced in *j*-th element of main vector to obtain a D-dimensional vector, and then its fitness is calculated in D-dimensional space.

In CLA-FA, every cell is assigned to one of components of main vector, in fact, the structure of cellular learning automata is one dimensionally (linear) with Moor neighborhood with periodic boundary (D-th cell is the neighbor of first cell). Therefore *i*-th cell is corresponding to *i*-th component of main vector and includes *i*-th swarm of fireflies. Available learning automata is equal in all cells and all are updated together (cellular learning automata is homogenous). An available learning automaton in each cell includes two action α_1 and α_2 and has variable structure with linear learning algorithm $p(n+1) = T[\alpha(n), \beta(n), p(n)]$ Actions of learning automata are as follows:

1- Decrease in Alpha and Gamma parameters with multiplying their values by smaller value than one.

2- Performing Reset action for defined percentage of available fireflies in swarm.

First action of learning automata is for balancing the global search ability and local search ability of proposed algorithm. For this purpose, first we consider large parameters value of Alpha and Gamma so the global search ability of algorithm to be high. Thus, the swarm moves faster toward global optimum and passes local optimums with more ability. With performing first action of learning automata, with fireflies movement toward global optimum, gradually parameters value of Alpha and Gamma are decreased so finally the algorithm could perform acceptable local search near global optimum.

In the proposed algorithm, each of the fireflies has corresponding Alpha and Gamma for itself, which is the values of these parameters, can be different in two different fireflies. Second action of learning automata is considered for avoiding premature convergence in order to increase the capability of escaping from local optimums. For this purpose, by doing this action, a defined percentage of fireflies of swarm randomly disperse in problem space and their Alpha and Gamma parameters values are become equal to their initial values (Reset action). Thus, fireflies that have leaved the swarm return to swarm by FA search strategy; they search spaces of problem which were remote to find better values of fitness that it can cause to discover positions with better fitness.

In the cellular learning automata that used in proposed algorithm, each cell is neighbor with two previous and next cells. Dominated local rule on cellular learning automata determines whether the action learning automata has done should receive reward or penalty. After that learning automata in cells do an action, each cell will evaluate the amount of its neighbor's improvement in previous iteration. If the ratio of at least one of neighbors is higher than defined percentage of swarm improvement corresponding to considered cell, performed action by learning automata will be fined, otherwise it'll be rewarded. In fact, if neighbor have considerable improvement to considered swarm. In corresponding dimension of this swarm, the algorithm is trapped in local optimum. Thus in proposed CLA-FA algorithm, there is a balance between global search and local search ability as well as the capability of avoiding preceding convergence is increased. Experimental results are given in next section.

5. Experimental Results

In this section, five functions are utilized for evaluation of the proposed CLA-FA algorithm. All of them are standard test functions. Applied functions together with their search space range are presented in Table 1. It should be mentioned that the global optimum solution of above functions is 0. In this paper, for evaluation of the proposed algorithm, we execute the standard FA and CLA-FA in 30 times and the largest iteration numbers set as 20 for the five functions. We test the mentioned functions in 10, 20 and 30-Dimensional spaces. The number of swarm is equal the number of dimension, and each swarm population size is 30. In the standard FA the Alpha set as 0.2 and Gamma set as 1.in the proposed CLA-FA, the maximum values for Alpha and Gamma set as 1 and the minimum values set as 0.01. Reward and punishment (a and b) coefficients for learning automata in CLA-FA assumed 0.01. For first action of learning automat, the value multiplied by Alpha and Gamma for reducing them is considered 0.9. Performing second action of learning automata resets 20 percent of swarm population. In local rule, if neighbors improve more than 10 times, performed action by automata is punished. Experiments repeated 30 times and the best, mean, variance, standard deviation and the worst case which is obtained from the standard run in 10, 20 and 30-dimensional spaces on Sphere, Ackley, Rastrigin, Xin-she yang and Step, functions have been shown in Table 2. As it shown, the results of the CLA-FA in all cases are better than FA in 10, 20 and 30 dimensional spaces, and it shows the capability of the proposed algorithm in finding optimal solution.

Table 1: The Name of Function We Use In This Paper.

Test Function	**Dim**	**Func-name**
$f_0(x)=\sum_{i=1}^{D} x_i^2$	$[-100, 100]^D$	Sphere
$f_5(x)=20+e-20e^{-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{D} x_i^2}} - e^{\frac{1}{2}\sum_{i=1}^{D}\cos(2\pi x_i)}$	$[-32, 32]^D$	Ackly
$f_1(x)=\sum_{i=1}^{D}\left(x_i^2-10\cos(2\pi x_i)+10\right)$	$[-5.12, 5.12]^D$	Rastrigin
$f_6(x)=\left(\sum_{i=1}^{D}\lvert x_i\rvert \exp\left[-\sin\left(x_i^2\right)\right]\right)$	$[-2\pi, 2\pi]^D$	Xin-s Yang
$f_3(x)=\sum_{i=1}^{D}\left(\lfloor x_i+o.5\rfloor\right)^2$	$[-100, 100]^D$	Step

6. Conclusion

In this paper, we introduced a modify firefly algorithm and use this algorithm with cellular learning automata (CLA). The proposed algorithm, allocated one cell of CLA to each dimension of problem space that each of them consisted of a firefly swarm. Available learning automata in each cell has the responsibility of making balanced between global search and local search, preventing from preceding convergence and making variety in group. In the proposed algorithm, since every swarm optimizes one dimension and cooperative property exists among swarms, ability and convergence rate of the algorithm is acceptable and improved the FA results.

Table 2. Comparison of FA and CLA-FA on Sphere, Ackly, Rastrigin, Xin-she yang and Step functions.

Functions	Dim	Criteria	FA	FFA
Sphere	10	Best	8.4464e+003	**9.0163e-006**
		Ave-std	1.3816e+004±2.7545e+003	**2.0183e-004±1.9972e-004**
		Worst	1.8274e+004	**7.5604e-004**
	20	Best	2.7351e+004	**8.7387e-005**
		Ave-std	3.5835e+004±5.4730e+003	**3.6209e-004±2.5969e-004**
		Worst	4.5138e+004	**0.0013**
	30	Best	4.1884e+004	**1.8423e-004**
		Ave-std	6.0124e+004±6.4849e+003	**5.3701e-004±2.4979e-004**
		worst	7.2690e+004	**0.0013**
Ackly	10	Best	18.1092	**0.0063**
		Ave-std	19.5479±0.6590	**0.0187±00118**
		Worst	20.2489	**0.0575**
	20	Best	19.6377	**0.0069**
		Ave-std	20.2197±0.4046	**0.0169±0.0064**
		Worst	20.4192	**0.0371**
	30	Best	19.8259	**0.0076**
		Ave-std	20.3880±0.1032	**0.0171±0.0037**
		Worst	20.6653	**0.0247**
Rastrigin	10	Best	82.684	**0.0028**
		Ave-std	96.7432±12.2943	**0.0487±0.0381**
		Worst	112.7496	**0.1384**
	20	Best	199.8921	**0.0064**
		Ave-std	231.2942±10.9620	**0.0988±0.0797**
		Worst	252.3455	**0.4027**
	30	Best	310.3268	**0.0286**
		Ave-std	348.1204±18.7868	**0.1521±0.0681**
		Worst	377.6572	**0.3368**
Xin- she Yang	10	Best	0.0063	**5.6752e-004**
		Ave-std	0.0115±0.0478	**6.1336e-004±3.3381e-005**
		Worst	0.0207	**6.7460e-004**
	20	Best	9.1157e-006	**5.1678e-008**
		Ave-std	1.0497e-004±6.3480e-005	**5.8820e-008±5.2323e-009**
		Worst	2.7371e-004	**7.1866e-008**
	30	Best	2.5782e-008	**3.8317e-012**
		Ave-std	4.8277e-007±6.7683e-007	**4.5330e-012±4.9745e-013**
		Worst	1.5961e-006	**5.8086e-012**
Step	10	Best	6359	**0**
		Ave-std	1.3533e+004± 2.6312e+003	**0**
		Worst	19475	**0**
	20	Best	23253	**0**
		Ave-std	3.6567e+004± 4.4869e+003	**0**
		Worst	44467	**0**
	30	Best	51173	**0**
		Ave-std	6.1107e+004±5.9982e+003	**0**
		Worst	71368	**0**

References

[1] Baeck, T., Fogel, D. B., Michalewicz, Z.: Handbook of Evolutionary Computation. Taylor & Francis (1997)

[2] Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press (1999)

[3] Kennedy, J., Eberhart, R., Shi, Y.: Swarm intelligence. Academic Press (2001)

[4] Yang, X. S.: Biology-derived algorithms in engineering optimization. (Chapter 32), in Handbook of Bioinspired Algorithms and Applications (eds Olarius & Zomaya), Chapman & Hall / CRC (2005)

[5] Liu, Y., Passino, K. M.: Swarm Intelligence: A Survey. In Proc. of ٤th International Conference of Swarm Intelligence (2005)

[6] Kennedy, J., Eberhart, R. C.: Particle Swarm Optimization. In: IEEE International Conference on Neural network، pp. 1942--194 (1995)

[7] Darigo, M., Birattari, M., Stutzle, T.: Ant Colony Optimization. In: IEEE Computational Intelligent Magazine، Vol. 1، pp. 28--39 (2006)

[8] Li, L. X., Shao, Z. J., Qian, J. X.: An Optimizing Method Based on Autonomous Animate: Fish Swarm Algorithm. In: Proceeding of System Engineering Theory and Practice، Vol. 11، pp. 32--38 (2002)

[9] Pham, D.T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., Zaidi, M.: The Bees Algorithm - A Novel Tool for Complex Optimisation Problems. Proceedings of IPROMS 2006 Conference, pp. 454--461(2006)

[10] Yang, X. S.: Nature-Inspired etaheuristic Algorithms. Luniver Press (2008)

[11] Yang, X. S.: Firefly algorithm، stochastic Test Functions and Design ptimization.Int. J. bio-inspired computation (2010)

[12] Yang, X. S.: Firefly algorithm for multimodal optimization. in: tochasticAlgorithms: foundations and applications ،SAGA ،lecture notes in computer sciences, pp. 169--178 (2009)

[13] Yang, X. S.: Firefly algorithm، levy flight and global optimization. in:research and development in intelligent system XXVI(Eds M . bramer، R. Ellis، M. Petridis)، Springer London، pp. 209--218 (2010)

[14] Sayadi، M. M., Ramezanian، R., GhaFAari-Nasb، N.:A discrete firefly meta_heuristic with local search for makespan minimization in permutation flow shop scheduling problem. ،Imternational Journal of Industrial Engineering Computations،1،pp.1--10 (2010

[15] M. R. Meybodi, H. Beigy and M.Taherkhani, "Cellular Learning Automata",Proceedings of 6th Annual CSI Computer Conference, Computer Engineering Department, University of Isfahan, pp. 153- 163, 2001.

[16] S. Wolfram, “Theory and Applications of Cellular Automata”,Advanced Series on Complex Systems, Singapore: World Scientific Publication, 1986.