International Journal of Management and Fuzzy Systems
2016; 2(6): 51-57
http://www.sciencepublishinggroup.com/j/ijmfs
doi: 10.11648/j.ijmfs.20160206.11

An Improved Firefly Algorithm Based on Local Search Method for Solving Global Optimization Problems

R. M. Rizk-Allah

Department of Basic Engineering Science, Faculty of Engineering, Minoufia University, Shebin El-Kom, Egypt

Email address:
rizk_masoud@yahoo.com

To cite this article:
R. M. Rizk-Allah. An Improved Firefly Algorithm Based on Local Search Method for Solving Global Optimization Problems. *International Journal of Management and Fuzzy Systems.* Vol. 2, No. 6, 2016, pp. 51-57. doi: 10.11648/j.ijmfs.20160206.11

Received: December 9, 2016; **Accepted**: December 20, 2016; **Published**: March 1, 2017

Abstract: This paper proposes an improved firefly algorithm (IFA)based on local search method for solving globaloptimization problems. The main feature of the proposed algorithm is to improve the solutions quality generated from the fireflies by embedding the local search method. Moreover, the new solutions are generated based on the movement formula of the fireflies that is modified by exponential formula. The exponential formula reduces the randomization parameter so that it decreases gradually as the optimum is approaching. In addition, local search method (LSM) is introduced to improve the solution quality. Finally, the proposed algorithm is tested on several benchmark problems from the usual literature and the numerical results have demonstrated the superiority of the proposed algorithm in finding the global optimal solution.

Keywords: Firefly Algorithm, Local Search Method, Global Optimization

1. Introduction

Optimization problems are of importance for the industrial as well as the scientific world in many applications. There are many optimization problems that present attributes, such as high nonlinearity and multimodality, the solution of this kind of problems is usually a complex task. Moreover, in many instances, complex optimization problems present noise and/or discontinuities which make traditional deterministic methods inefficient to find the global solutions. Meanwhile, global optimization methods based on meta-heuristics are robust alternatives to solve complex optimization problems and do not require any properties of the objective function have been developed.

Due to the computational drawbacks of existing numerical methods, researchers have to rely on meta-heuristic algorithms based on simulations to solve some complex optimization problems. A common feature in meta-heuristic algorithms is that they combine rules and randomness to imitate natural phenomena. These phenomena include the biological evolutionary process (e.g., the genetic algorithm (GA) [1] and the differential evolution (DE) [2]), animal behavior (e.g., particle swarm optimization (PSO) [3] and ant colony algorithm (ACA) [4]), and the physical annealing process (e.g., simulated annealing (SA) [5]). Over the last decades, many meta-heuristic algorithms and their improved algorithms have been successfully applied to various engineering optimization problems [6, 7, 8, 9, 10]. They have outperformed conventional numerical methods on providing better solutions for some difficult and complicated real-world optimization problems.

A promising new meta-heuristic algorithm denoted as firefly algorithm (FA) which inspired by social behavior of fireflies and the phenomenon of bioluminescent communication. There are two important issues in the firefly algorithm that are the variation of light intensity and formulation of attractiveness. Yang [21] simplified that the attractiveness of a firefly is determined by its brightness which in turn is associated with the objective function. In general, the attractiveness is proportional to their brightness. Furthermore, every member of the firefly swarm is characterized by its bright that can be directly expressed as an inverse of an objective function for a minimization problem.

In this paper we propose improved firefly algorithm based on local search method for named IFA-LSM for solving the global optimization problems. The motivation for the proposed algorithm is to improve the solutions quality for solving the global optimization problems. This methodology consists of two phases. The first phase explore new solutions based the movement formula of the fireflies that is modified by exponential formula. The exponential formula reduces the randomization parameter so that it decreases gradually as the optimum is approaching, while the other phase employs the local search method to exploit the existing solutions to improve the solution quality of optimization problems. By these phases, the proposed algorithm achieves robust results compared to the current well-known algorithms in the literature. Finally, the exploration and exploitation enhanced the superiority of the proposed approach in finding the global optimal solution.

The organization of the remaining paper is as follows. In Section 2 we describe some preliminaries on optimization problems and basics of FA. In Section 3, the proposed algorithm, named IFA-LSM, is explained in detail. Experiments anddiscussions are presented in Section 4. Finally, we conclude the paper in Section 5.

2. Preliminaries

2.1. Statement of Global Optimization Problem

The general numerical global optimization problem [22] can be defined as in (1):

Find $\mathbf{x}$ such that

$$\min F(\mathbf{x}), \mathbf{x} = (x_1, x_2, ..., x_n) \in \Re^n \tag{1}$$

where $\mathbf{x} \in \Omega \subseteq S$. The objective function F is defined on the search space $S \subseteq \Re$ and the set $\Omega \subseteq S$ defines the feasible region. Usually, the search space S is defined as an $n-$ dimensional rectangle in $\Re^n$, domains of variables defined by their lower and upper bounds as in (2):

$$x_j^L \leq x_j \leq x_j^U, \; j = 1, 2, ..., n \tag{2}$$

2.2. The Basics of FA

FA [30] is one of the most recent meta-heuristic techniques for approximate optimization. The inspiring source of FA is the social behavior of fireflies for sharing food with others or for attracting the prey. At the core of this behavior is the direct communication between the fireflies by means of bioluminescent communication, which enables them to moves toward a neighbor that glows brighter.

There are two important issues in the firefly algorithm that are the variation of light intensity and formulation of attractiveness. Yang [10] simplified that the attractiveness of a firefly is determined by its brightness which in turn is associated with the objective function. In general, the attractiveness is proportional to their brightness. Furthermore, every member of the firefly swarm is characterized by its bright that can be directly expressed as an inverse of an objective function for a minimization problem. Based on this objective function, initially, all the agents (fireflies) are randomly dispersed across the search space. The two stages of the firefly algorithm are as follows.

1) Variation of light intensity: Light intensity is related to objective values [10]. So for minimization problem a firefly with high intensity will attract another firefly with high intensity. Assume that there exists a swarm of m agents (fireflies) and $\mathbf{x}_i$ represents a solution for a firefly i, whereas $f(\mathbf{x}_i)$ denotes its fitness value. Here the brightness I of a firefly is selected to reflect its current position $\mathbf{x}$ of its fitness value $f(\mathbf{x})$, given as in (3):

$$I(\mathbf{x}_i) = 1/f(\mathbf{x}_i) \;, i = 1, 2, ..., m \tag{3}$$

2) Movement toward attractive firefly: The firefly has an attractiveness which is proportional to the light intensity seen by adjacent fireflies. Each firefly has its distinctive attractiveness β which implies how strong it attracts other members of the swarm. However, the attractiveness β is relative; it will vary with the distance. r_{ij} between two fireflies i and j at locations $\mathbf{x}_i$ and $\mathbf{x}_j$ respectively, is given as in (4):

$$r_{ij} = \left\| \mathbf{x}_i - \mathbf{x}_j \right\| \tag{4}$$

The attractiveness function $\beta(r)$ of the firefly is determined by using (5):

$$\beta(r) = \beta_0 \, e^{-\gamma r^2} \tag{5}$$

where β_0 is the attractiveness at $r = \mathbf{0}$ and γ is the light absorption coefficient.

The movement of a firefly i at location $\mathbf{x}_i$ attracted to another more attractive (brighter) firefly j at location $\mathbf{x}_j$ is determined as in (6):

$$\mathbf{x}_i(t+1) = \mathbf{x}_i + \beta_0 \, e^{-\gamma r^2} \, (\mathbf{x}_j - \mathbf{x}_i) + \alpha_0 \, (rand - 0.5) \tag{6}$$

where the second term is due to the attraction while the third term is randomization with α being the randomization parameter and *rand* is a random number generator uniformly distributed in $[0,1]$. The pseudo code of the FA can be summarized in the Table 1.

Table 1. *The pseudo code of the FA.*

```
Set values of parameters.
Create an initial population of fireflies, m, within n-dimensional search space x_i, i = 1,2,...,m .
Evaluate the fitness of the population f(x_i) which is inversely proportional to light intensity I(x_i) .
while (not termination condition) do
    for i = 1 to m
      for j = 1 to m .
        if I_i < I_j ,
            Move firefly i towards j by using Equation (6)
        end if
        Vary attractiveness with distance r via e^(-γr^2)
        Evaluate new solutions and update light intensity by using Equation (5)
      end for j
    end for i
    Rank fireflies and find the current best;
end while
```

3. The Proposed Algorithm

The motivation for the proposed algorithm is to improve the solutions quality for solving the global optimization problems compared with the state of the art. By extending the basic ideas of FA, we can develop the following IFA-LSM.

The procedure starts with an appropriate definition of objective functions with associated nonlinear constraints. We first initialize a population of m fireflies so that they should distribute among the search space as uniformly as possible. This can be achieved by using sampling techniques via uniform distributions. Once the tolerance or a fixed number of iterations is defined, the iterations start with the evaluation of brightness or objective values of all the fireflies and compare each pair of fireflies. Then, firefly is attracted to another more attractive (brighter) firefly. The main steps of the IFA-LSM are summarized as follows:

Step 1. Initialization

Initialize a swarm of fireflies with assigned a random vector $\mathbf{x},(x_1,x_2,...,x_n)$, where each firefly contains n variables (i.e., the position of the i^{th} firefly in the n dimensional search space can be represented as $\mathbf{x}_i=(x_{i1},x_{i2},...,x_{in})$). Furthermore every member of the swarm is characterized by its light intensity (i.e., initialize each firefly with distinctive light intensity $(I_0(\mathbf{x}_i),i=1,2,...,m$).

Step 2. Light intensity

Calculate the light intensity $I_i(\mathbf{x}_i),i=1,2,...,m,$ for each firefly which in turn is associated with the encoded objective function, where for minimization problem the brighter firefly represents the minimum value for $I(\mathbf{x})$.

Step 3. Movement toward attractive firefly

In this step, each member of the swarm explores new solution based on the movement of a firefly. The traditional movement of a firefly is implemented by using Equation (6) where the randomization term may moves the firefly to lose its best location, so we introduce a modification on the randomization term that makes the fireflies approached from the optimum. This modification represents a further improvement on the convergence of the proposed algorithm by using Equations (7) and (8).

$$\mathbf{x}_i(t+1)=\mathbf{x}_i+\beta_0\, e^{-\gamma r^2}\,(\mathbf{x}_j-\mathbf{x}_i)+\alpha_t\,(rand-0.5) \quad (7)$$

$$\alpha_t=\alpha_0\,\theta^t\,,t=1,2,...,T \quad (8)$$

where α_0 is the initial randomness factor, T is the maximum number of generations, and $\theta\in[0,1]$ is the randomness reduction constant.

Step 4. local search method

In this step the local search scheme is carried around the found solution by fireflies in order to enhance the existing solution by the fireflies, therefore the fireflies move in new directions in search of newer regions. The pseudo code of the local search scheme is shown in Table 2.

The flowchart of the proposed IFA-LSM approach is shown in Figure 1.

Table 2. *The pseudo code of the local search method.*

Input: $\mathbf{x}=(x_1,x_2,...,x_i,...,x_n);x_i^L;x_i^U$; number of maximum iteration.
Set $t=0$
Generate $d\mathbf{x}$ ($d\mathbf{x}=.5*(U-L)*(\varepsilon)^n$
If $\exists\mathbf{x}'=\mathbf{x}+d\mathbf{x}\mid\mathbf{x}'\in\Omega$,then $\mathbf{x}_{new}=\mathbf{x}'$
Else if $\exists\mathbf{x}'=\mathbf{x}-d\mathbf{x}\mid\mathbf{x}'\in\Omega\ \mathbf{x}'\in\Omega$,then $\mathbf{x}_{new}=\mathbf{x}'$
Else $\mathbf{x}_{new}=[\,]$ then $t=t+1$
End if
output: $\mathbf{x}_{new}$

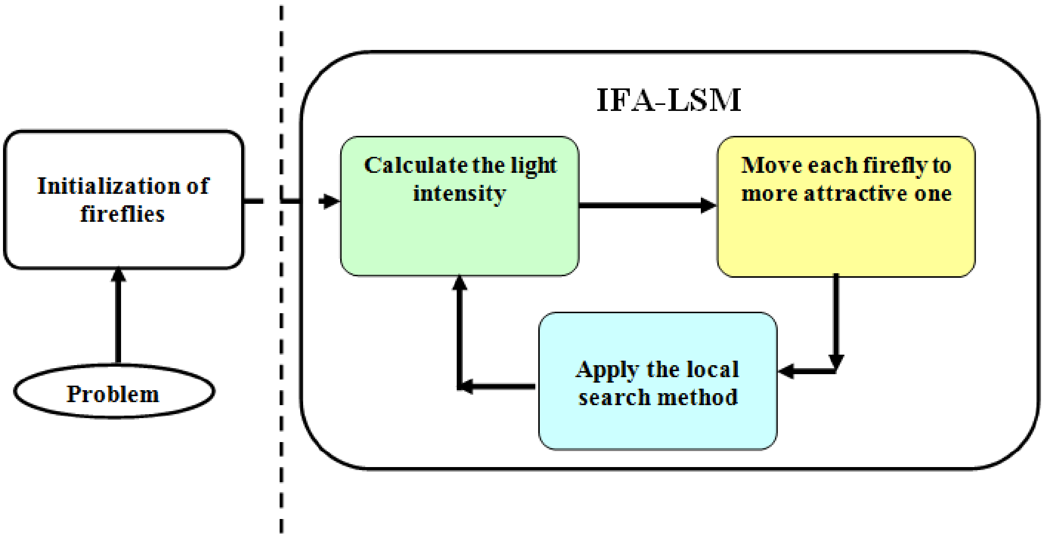

Figure 1. *The flowchart of the proposed IFA-LSM.*

4. Experiments and Discussions

An extensive set of experiments have been conducted, in order to show IFA-LSM algorithm's effectiveness for the purpose of global optimization. fifteen common test functions were used in the experiments and then the evaluated results were compared with the prominent algorithms that reported in [19, 23] which are called successive zooming genetic algorithm (SZGA), harmony search theory (HS), dynamic random search technique (DRASET) [23], chaotic particle swarm optimization(CPSO), particle swarm optimization (PSO), genetic algorithm (GA) and particle swarm ant colony optimization (PSACO) [19]. The test functions, which are benchmark from [19, 23], are listed in Table 3. Table 3 gives the details of the test functions, including their equations, dimensions, domains and the optimal values. Selected test functions are run on PC, which has Pentium 4 3.0 GHz processor and 1.0 GB RAM while testing the performance IFA-LSM. The IFA-LSM algorithm had been coded in MATLAB 7.

4.1. Parameters Setting

The proposed algorithm contains number of parameters. These parameters affect the performance of the proposed algorithm. Extensive experimental tests were conducted to see the effect of different values on the performance of the proposed algorithm. Based upon these observations, the following parameters have been set as in Table 4.

Table 3. *Test functions.*

Test functions	Dimension	Domain	optimal
$F_1 = x_1^2 + 2x_2^2 - 0.3\cos(3\pi x_1) - 0.4\cos(4\pi x_2) + 0.7$	2	[-1.281.28]	0
$F_2 = [\cos(2\pi x_1) + \cos(2.5\pi x_1) - 2.1]*[2.1 - \cos(3\pi x_2) + \cos(3.5\pi x_2)]$	2	[-11]	-16.09172
$F_3 = \left[0.002 + \sum_{j=1}^{25}(j + \sum_{i=1}^{2}(x_i - a_{ij})^6)^{-1}\right]^{-1}$	2	[-65.53665.536]	0.9980
$a = \begin{bmatrix} -32 & -16 & 0 & 16 & 32 & -32 & -16 & 0 & 16 & 32 & -32 & -16 & 0 & 16 & 32 & -32 & -16 & 0 & 16 & 32 & -32 & -16 & 0 & 16 & 32 \\ -32 & -32 & -32 & -32 & -32 & -16 & -16 & -16 & -16 & -16 & 0 & 0 & 0 & 0 & 0 & 16 & 16 & 16 & 16 & 16 & 32 & 32 & 32 & 32 & 32 \end{bmatrix}$			
$F_4 = \left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi}\right)\cos(x_1) + 10$	2	$x_1 \in [-5\ 10]$ $x_2 \in [0\ 15]$	0.3978873
$F_5 = \left(4 - 2.1x_1^2 + \frac{x_1^4}{3}\right)x_1^2 + x_1x_2 + \left(4x_2^2 - 4\right)x_2^2$	2	$x_1 \in [-3\ 3]$ $x_2 \in [-2\ 2]$	-1.0316285
$F_6 = \left(1 + (x_1 + x_2 + 1)^2\left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right)* \left(30 + (2x_1 - 3x_2)^2\left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right)$	2	$[-5\ 5]$	3
$F_7 = \left(\sum_{i=1}^{5} i\cos((i+1)x_1 + i)\right)*\left(\sum_{i=1}^{5} i\cos((i+1)x_2 + i)\right)$	2	$[-10\ 10]$	-186.73091

Test functions	Dimension	Domain	optimal
$F_8 = 100\left(x_2 - x_1^2\right)^2 + \left(1 - x_1\right)^2$	2	$\left[-10\ 10\right]$	0
$F_9 = \exp\left\{\frac{1}{2}\left(x_1^2 + x_2^2 - 25\right)^2\right\} + \sin^4(4x_1 - 3x_2) + \frac{1}{2}\left(x_1 + x_2 - 10\right)^2$	2	$\left[-5\ 5\right]$	1
$F_{10} = \frac{1}{10}\left(12 + x_1^2 + \frac{1 + x_2^2}{x_1^2} + \frac{x_1^2 x_2^2 + 100}{(x_1 x_2)^4}\right)^2$	2	$\left[0\ 10\right]$	1.74
$F_{11} = 100\left(x_2 - x_1^2\right)^2 + \left(1 - x_1\right)^2 + 90\left(x_4 - x_3^2\right)^2 + \left(1 - x_3\right)^2 + 10.1\left[\left(x_2 - 1\right)^2 + \left(x_4 - 1\right)^2\right] + 19.8\left(x_2 - 1\right)\left(x_4 - 1\right)$	4	$\left[-10\ 10\right]$	0
$F_{12} = \left(x_1 + 10x_2\right)^2 + 5\left(x_3 - x_4\right)^2 + \left(x_2 - 2x_3\right)^4 + 10\left(x_1 - x_4\right)^4$	4	$\left[-5\ 5\right]$	0
$F_{13} = \sum_{i=1}^{19}\left[\left(x_i^2\right)^{\left(x_{i+1}^2 + 1\right)} + \left(x_{i+1}^2\right)^{\left(x_i^2 + 1\right)}\right]$	20	$\left[-1\ 4\right]$	0
$F_{14} = (\pi / 20)\left[10\sin^2(\pi x_1) + \sum_{i=1}^{19}\left(\left(x_i - 1\right)^2\left(1 + 10\sin^2(\pi x_{i+1})\right)\right) + \left(x_{20} - 1\right)^2\right]$	20	$\left[-10\ 10\right]$	0
$F_{15} = -\sum_{i=1}^{4} c_i \exp\left[-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right]$	3	[01]	-3.86278

i	a_{ij}	c_i	p_{ij}
1	3 10 30	1	0.3689 0.1170 0.2673
2	0.1 10 35	1.2	0.4699 0.4387 0.7470
3	3 10 30	3	0.1091 0.8732 0.5547
4	0.1 10 35	3.2	0.0381 0.5743 0.8828

Table 4. *The algorithm parameters.*

Initial light intensity (I_0)	0
Initial attractiveness (β_0)	1
The light absorption coefficient(γ)	1
The randomness reduction constant(θ)	0.9
Randomization parameter (α_1)	0.2

4.2. The Comparison of Solution Quality

In order to examine the capability of IFA-LSM in global optimization problems, a comparison is made with the prominent algorithms from the literature. The test functions have been solved by IFA-LSM for 10 times. The starting values of the variables for each problem were selected randomly for all runs from the solution space. The results found by IFA-LSM such as the best and average function value, numbers of function evaluation (NFE) and solution time in seconds have been recorded in Table 5, whereas for the other algorithms only the function value and numbers of function evaluation are given because the solution times, the best and average function value for some algorisms not given. It is obtained that founded best function values by IFA-LSM are the same or the closest as average function values for all functions except from the functions F_1, F_5, F_8, F_{12}. As it deduced from Table5, the IFA-LSM is successful while finding the optimum solution of the given functions and IFA-LSM outperforms the prominent algorithms for all functions. On the other hand, IFA-LSM can find the global minimum with less iteration number than compared algorithms except for the functions F_1, F_2, F_6, F_{10}.

In this subsection, a comparative study has been carried out to assess the proposed approach concerning quality of the solution. On the first hand, evolutionary techniques suffer from the quality of solution. Therefore the proposed approach has been used to increase the solution quality by combining the two merits of two meta-heuristic algorithms. On the other hand, unlike classical techniques our approach search from a population of points, not single point. Therefore our approach can provide a globally optimal solution. In addition, our approach uses only the objective function information, not derivatives or other auxiliary knowledge. Therefore it can deal with the non-smooth, non-continuous and non-differentiable functions which are actually existed in practical optimization problems. Another advantage is that the simulation results prove superiority of the proposed approach to those reported in the literature, where it is completely better than the other approaches. So, the IFA-LSM approach is quite competitive when compared with the other existing methods. Finally, the reality of using the proposed approach to handle complex problems of realistic dimensions has been approved due to procedure simplicity.

5. Conclusions

In this paper, we propose IFA-LSM algorithm which hybridizes the solution construction mechanism of FA with the LSM. In order to overcome the drawback of classical ant colony algorithm this not suitable for solving global optimizations, the solutions obtained by the ants are evolve by roaming the fireflies through the search space. Therefore the fireflies refine the positions found by the ants by producing number of solutions equal the number of solutions generated by the ants. On the other hand, the performance of FA is improved by reducing the randomization parameter so that it decreases gradually as the optima are approaching. The comparisons of numerical results show that there is as cope of research in hybridizing swarm intelligence methods to solve difficult continuous optimization problems and the IFA-LSM is a promising and valuable tool to solve global nonlinear optimization problems. A careful observation will reveal the following benefits of the proposed optimization technique.

a) It competitive when compared with the other existing algorithm.
b) The proposed algorithm is capable of capturing the global minimum for the problems very efficiently.
c) The carried out results verified the validity and the advantages of the proposed approach.
d) It can accelerate the convergence and boost the performance through elapsed low computational time.

The future work will be focused on two directions: (i) the application of IFA-LSM to constrained optimization problems; and (ii) the extension of the method to solve the multi-objective problems.

Table 5. *The comparison of solution quality.*

functions	**IFA-LSM**							**Compared algorithms**	
	Function value		**NFE**		**Time(s)**		**Name**	**Function value**	**NFE**
	Best	**Average**	**Best**	**Average**	**Best**	**Average**			
F_1	0	1.792475E-20	3150	3099	0.2810	0.3388	SZGA	0.298002E-7	4000
							DRASET	0	957
F_2	-16.09172	-16.09172	2100	1780	0.2500	0.2266	SZGA	-16.09172	4000
							DRASET	-16.09172	722
F_3	0.9980	0.9980	1600	1600	0.2970	0.3108	SZGA	0.9980	2000
							DRASET	0.9980	1823
F_4	0.3978873	0.3978873	200	200	0.6400	0.641	SZGA	0.39789	4000
							DRASET	0.39788737	219
							PSACO	0.3979	209
							CPSO	0.3979	NA*
							PSO	0.4960	NA
							GA	0.4021	NA
F_5	-1.03162845	-1.0316284	880	916	0.0940	0.0958	SZGA	-1.03163	3000
							DRASET	-1.0316284	1738
F_6	3	3	1370	1566	0.1720	0.2359	SZGA	3	4000
							DRASET	3	2550
							HS	3	400000
							PSACO	3	240
							CPSO	3	NA
							PSO	4.62602	NA
							GA	3.1471	NA
F_7	-186.73091	-186.73091	1600	1650	0.2030	0.2031	SZGA	-186.73091	3000
							DRASET	-186.73091	1665
							PSACO	-186.7309	534
							CPSO	-186.7274	NA
							PSO	-180.3265	NA
							GA	-182.1840	NA
F_8	0	3.311002E-11	10250	11400	3.8750	4.6172	DRASET	3.9053E-15	11623
							HS	5.684341886E-15	50000
F_9	1	1	4250	4337	1.0310	1.0608	DRASET	1	29663
							HS	1	45000
F_{10}	1.744151	1.744151	1100	1100	0.0790	0.0790	DRASET	1.74415200796	251
							HS	1.74415	800
F_{11}	8.9518E-15	1.3022714E-15	32135	36750	19.5310	15.0280	SZGA	0.13074E-5	175438
							DRASET	3.72E-12	49855
							HS	4.8515E-9	70000
F_{12}	0	7.00841E-16	4800	4815	2.8900	4.8530	DRASET	8.17E-9	79990
							HS	0.1254032468E-11	100000
F_{13}	3.5906E-18	3.3480E-18	16500	5835	5	1.6668	SZGA	0.25422E-7	320000
							DRASET	2.45E-16	49325
F_{14}	4.1651E-17	3.34037E-17	10200	9780	2.8440	2.8312	SZGA	0.230033E-3	239521
							DRASET	5.93E-12	19994
F_{15}	-3.8628	-3.8628	1500	1860	2.907	2.900	PSACO	-3.8628	2000
							CPSO	-3.8610	NA
							PSO	-3.8572	NA
							GA	-3.8571	NA

References

[1] J. H. Holland, Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor, MI, 1975.

[2] R. Storn, Differential evolution design of an IIR-filter, in: IEEE International Conference on Evolutionary Computation, Nagoya, 1996, pp. 268–273.

[3] J. Kennedy, R. C. Eberhart, Particles warm optimization, in: Proceedings of IEEE International Conference on Neural Networks, 1995, pp. 1942–1948.

[4] M. Dorigo, T. Stützle, Ant Colony Optimization, MIT Press, London, 2004.

[5] S. Kirkpatrick, C. Gelatt, M. Vecchi, Optimization by simulated annealing, Science 220, 1983, pp. 671–680.

[6] M. H. Mashinchi, A. O. Mehmet, P. Witold, Hybrid optimization with improved tabu search, Applied Soft Computing 11, 2011, pp. 1993–2006.

[7] A. Y. Qing, Dynamic differential evolution strategy and applications in electromagnetic inverses catering problems, IEEE Transactions on Geo science and Remote Sensing 44 (1), 2006, pp. 116–125.

[8] J. H. V. Sickel, K. Y. Lee, J. S. Heo, Differential Evolution and its Applications to Power Plant Control, in: International Conference on Intelligent Systems Applications to Power Systems, ISAP, 2007, pp. 1–6.

[9] Y. Marinakis, M. Marinaki, G. Dounias, Particles warm optimization for pap-smear diagnosis, Expert Systems with Applications 35 (4), 2008, pp. 1645–1656.

[10] M. Serrurier, H. Prade, Improving inductive logic programming by using simulated annealing, Information Sciences 178 (6), 2008, pp. 1423–1441.

[11] M. Dorigo, Learning and Nature Algorithm (in Italian). Ph. D. Dissertation, Dipartimento di Electtonica, Politecnico di Milano, Italy, 1992.

[12] L. Chen, J. Shen, L. Qin, H J. Chen, An improved ant colony algorithm in continuous optimization. Journal of Systems Science and Systems Engineering, 12 (2), 2003, pp. 224-235.

[13] J. Dreo, P. Siarry, An ant colony algorithm aimed at dynamic continuous optimization. Appl. Math. Comput., 181, 2006, pp. 457-467.

[14] A. A. Mousa, Waiel F. Abd El-Wahed, R. M. Rizk-Allah, A hybrid ant colony optimization approach based local search scheme for multiobjective design optimizations, Electric Power Systems Research, 81, 2011, pp. 1014–1023.

[15] C. Blum, Beam-ACO-Hybridizing ant colony optimization with beam search: An application to open shop scheduling. Computers & Operations Res32 (6), 2005, pp. 1565–1591.

[16] Z. Xiaoxia, T. Lixin, A new hybrid ant colony optimization algorithm for the vehicle routing problem, Pattern Recognition Letters 30, 2009, pp. 848–855.

[17] P. Hadi, T. M. Reza, Solving a multi-objective open shop scheduling problem by a novel hybrid ant colony optimization, Expert Systems with Applications 38, 2011, pp. 2817–2822.

[18] Y. Jingan, Z. Yanbin, An improved ant colony optimization algorithm for solving a complex combinatorial optimization problem, Applied Soft Computing 10, 2010, pp. 653–660.

[19] P. S. Shelokar, P. Siarry, V. K. Jayaraman, B. D. Kulkarni, Particle swarm and ant colony algorithms hybridized for improved continuous optimization. Applied Mathematics and Computation 188, 2007, pp. 129-142.

[20] C. Shyi-Ming, C. Chih-Yao, Parallelized genetic ant colony systems for solving the traveling salesman problem, Expert Systems with Applications 38, 2011, pp. 3873–3883.

[21] X. S. Yang.: Nature-Inspired Metaheuristic Algorithms. Luniver Press (2008).

[22] M. S. Bazaraa, C. M. Shetly, Nonlinear Programming Theory and Algorithms, Wiley, NewYork, 1979.

[23] K. Hamzacebi, F. Kutay, continuous functions minimization by dynamic randomsearch technique, Applied Mathematical Modelling 31, 2007, pp. 2189-2198.