

# Using improved firefly algorithm based on genetic algorithm crossover operator for solving optimization problems

Fazli wahid<sup>a,\*</sup>, Ahmed Khalaf zager alsaedi<sup>b</sup> and Rozaida Ghazali<sup>a,c,\*\*</sup>

<sup>a</sup> *Faculty of Computer Science and Information Technology,  
Universiti Tun Hussein Onn, Malaysia  
E-mail: wahid\_uomian@hotmail.com*

<sup>b</sup> *College of Science, Physics Department, University of Misan, Ministry of Higher Education and Scientific Research of Iraq*

*E-mail: ahmedkhalafzager@uomisan.edu.iq*

<sup>c</sup> *Soft Computing Data Mining Research Centre (SCDM),  
Universiti Tun Hussein Onn, Malaysia  
E-mail: rozaida@uthm.edu.my*

**Abstract.** Firefly algorithm (FA) is one of the most recently introduced stochastic, nature-inspired, meta-heuristic approaches that have seen countless applications in solving various types of optimization problems. The major source of inspiration leading to the development of FA is the phenomenon of light emission by fireflies that attract other fireflies for their potential mates. All the fireflies are unisexual and attract each other according to the intensities of their flash lights. Higher the flash light intensity, higher is the power of attraction and vice versa. For solving optimization problem, the brightness of flash is associated with the fitness function to be optimized. The firefly algorithm is advantageous over other optimization algorithms due to its flexibility, simplicity, robustness and easy implementation but a major drawback associated with the standard FA applied for solving different optimization problems is poor exploitation capability when the randomization factor is taken large during firefly changing position. This poor exploitation may lead to skip the most optimal solution even present in the vicinities of the current solution which results in poor local convergence rate that ultimately degrades the solution quality. To overcome this problem, the crossover operator of genetic algorithm (GA) is incorporated into firefly position changing stage that results in better exploitation capability which improves the local convergence rate resulting in better solution quality. The performance of the proposed approach has been compared with standard FA, GA, artificial bee colony (ABC) and ant colony optimization (ACO) algorithms in terms of convergence rate for various types of minimization and maximization optimization functions.

**Keywords:** Firefly algorithm, hybrid firefly algorithm, optimization functions, faster convergence rate, better solution quality, crossover operator

## 1. Background

One of the most recently artificial intelligence (AI) techniques is swarm intelligence (SI) that has attracted the attention of researchers for the last few decades.

All the techniques coming under the umbrella of SI are used for solving different types of optimization problems [1]. The major source of inspiration for the SI techniques is the collective behavior of different types of swarms e.g. bees, termites, ants, school of fish and flock of birds. For keeping a strong inter-connected and coordinated system among the swarms, the individuals making the swarm play the most vital role.

---

\*Corresponding author. E-mail: wahid\_uomian@hotmail.com

\*\*Corresponding author. E-mail: rozaida@uthm.edu.my

But, in order to achieve different types of complicated goals and perform various complex activities, a decentralized system is developed by these individuals that contribute their part in keeping the coordination and inter-communication strong. All these participating individuals establish a very strong communication system for assisting the whole system in achieving different types of sophisticated goals e.g. making nets, searching food, migrating from one location to another location and keeping themselves safe in the dangerous situations. A well-managed and mutually understandable coordinated system is maintained for achieving these goals.

In order to keep this coordination system strong, two phenomena namely exploration and exploitation are used by the individuals of the swarms when performing activities in a group-wise manner. Exploration means collecting new information whereas the exploitation means using the existing information for communication among different individuals of the swarms to better manage their coordination. These two phenomena are used by SI techniques when defining solution search space during solving optimization problems. The exploration is the discovery of new solutions by widening the solution search space whereas the exploitation means focusing on so far discovered solutions to avoid missing optimal solution present around the present solutions.

The process of exploitation is used to focus more on the local solutions whereas the exploration is used for focusing on global solutions. During solving the optimization problems, the exploration is the process of increasing the solution search space to bring variations in the values of optimization function (collecting new information) whereas the exploitation means focusing on the so far found solutions to check all the nearby solutions to enable the search space to not skip the most optimal solution present in the local solution search space (using the existing information). A trade-off is required between the exploration and exploitation during finding solution of the optimization problems [2]. The self-organized and decentralized approach in swarm intelligence was first introduced by [3] for optimization purpose in cellular robotic system and since then this idea has seen many applications in different areas e.g. load balancing, finding solutions for optimization problems, routing in mobile networks and many other areas.

There are many SI techniques that have been developed for solving various types of optimization problems. Examples of these algorithms include ant colony

optimization (ACO) [4], particles swarm optimization (PSO) [5], artificial bee colony (ABC) [6]. Some recently SI algorithms developed for solving optimization problems include clustering algorithms [7], cuckoo search algorithm; krill herd algorithm [8], bat algorithm [9] and firefly algorithm [10]. Different SI techniques are also used in combination with other AI algorithms to target specific functionalities of AI approaches. For example, [11] applied PSO to optimize parameters for support vector machine used in freight volume prediction. [12] used FA to optimize weight for multi-layer perceptron applied for wind speed prediction. Similarly, [13] applied FA to use in combination support vector machine used in forecasting of field capacity and permanent wilting point.

## 2. Standard Firefly Algorithm (FA)

Xin-She Yang introduced the FA for the first time who was inspired from the light emission capability of fireflies which is the primary source of communication among different fireflies to achieve different goals e.g. reproduction and food search [10]. In order to solve optimization problems, Xin-She Yang associated the intensity of the light emitted by fireflies with the objective function to be optimized. According to physics rules of light, the intensity of the light emitted by firefly,  $I(r)$ , at distance  $r$  from the firefly can be observed by Eq. 1:

$$I(r) = \frac{I_o}{r^2} \quad (1)$$

Where  $I_o$  represents the light intensity generated at the light source. If  $\gamma$  is the absorption coefficient of the medium, then the light intensity  $I$ , at distance  $r$  is given by the Eq. 2:

$$I = I_o \exp(-\gamma r^2) \quad (2)$$

Where,  $r$  represents the distance between source of light and the light observation point. This light intensity can be associated with the attractiveness between the fireflies in FA which is given by Eq. 3:

$$B = \beta_o \exp(-\gamma r m) (m \geq I) \quad (3)$$

Where  $\beta_o$  represents the attractiveness at distance  $r = 0$ . The distance between two fireflies  $x_i$  and  $y_j$  called as Euclidian distance is given by Eq. 4:

$$r_{ij} = |x_i - y_j| = \sqrt{\sum_{k=1}^d (x_{i,k} - y_{j,k})^2} \quad (4)$$

In each generation, the position of firefly can be changed according to the following Eq. 5:

$$X_i = x_i + \beta_o \exp(-\gamma r_{ij}^2)(x_i - y_j) + \alpha \varepsilon \quad (5)$$

Where  $\gamma$  represents the parameter of randomization,  $\varepsilon$  represents random number generated in Gaussian distribution. The parameter of randomization is used to control the solution search space.

### 3. Literature Review

In order to improve the performance of standard FA, many modifications have been introduced by different authors and it has also been used in hybridization with other optimization algorithms. This section elaborates the recent work that has been carried out for modifying FA or hybridizing FA with other optimization algorithms. The performance of the firefly algorithm depends upon two main factors namely the attractiveness formulation and variations in the intensity of light of fireflies. These two factors are targeted by the researchers to bring improvements in the performance of the algorithm. For example, some modifications were made by [14] in the random movements of some fireflies if there is no significant improvement in the solution quality for some generations. A set of uniform random vectors is generated for improving the position of the brightest firefly and it is moved towards the best performance direction. The brightest firefly possesses the property of elitist solution because the solution is not replaced by the best solution in the current iteration if the best solution in the current iteration has lower fitness value.

For balancing the exploration and exploitation of the standard FA, [15] applied tidal force formula to bring modifications in the operation of the FA. The proposed approach is based on the gravitational effects of one massive body on the other massive body. The tidal force is a situation in which one body is dependent

on other bodies. For achieving the global minimum with a less number of iterations, the authors embedded the tidal force phenomenon in the standard FA framework. The tidal force was formulated in firefly operations when handling the objective function. Random uniform distribution in less and more dimensions was used for determining the direction of movements of fireflies. For exploration of the solution search space, the whole solution search space is searched and the minimum value is found. Similarly, [16] developed an enhanced model of FA for optimizing weights and bias values for the artificial neural network (ANN) to improve the performance of ANN. The authors introduced modifications in the tuning of attractiveness parameter of the standard FA. The mouse/gauss map was used as chaotic map for tuning the parameter for enhancing the capability of the standard FA and the performance of the standard FA was significantly improved by the introduction of mouse/gauss map for attractive parameter tuning.

A modified version of standard FA was proposed by [17] for finding optimal values of multilevel threshold for image segmentation. Three objective functions namely minimum cross entropy, between class variance method and kapur's entropy were optimized by the proposed modified FA. In the proposed model, the initial population in the standard FA was generated by using chaotic map. During firefly position changing stage, the authors embedded the global search method of PSO for further improving the efficiency of the standard FA. In the same manner, [18] modified the basic working mechanism of FA for making it more suitable for multi-objective optimization problem in big data. For introducing more diversification in the solution quality by improving the exploration of the solution search space, the authors used the crossover operation used in the differential equation. The authors used two objective functions to be optimized by the proposed modified FA model.

Similarly, [19] introduced an enhanced version of FA to improve its convergence rate. For improving the performance of the FA, the authors have embedded the attractive function of Mantegna's algorithm inside standard FA. The randomness of the standard FA during finding solution is reduced by implementing Levy flight that improves the solution quality. [20] modified the parameter tuning and the updating strategy of standard FA to improve its solution quality. The exploration and exploitation procedure of the FA was modified by introducing absorption coefficient and adaptive randomness during firefly position changing stage.

Instead of fixed randomness, the proposed approach used adaptive randomness in which the values of randomness change in each iteration. Similarly, [21] enhanced the exploration and exploitation of standard FA by introducing implicit firefly movements instead of random movements by fireflies during firefly position changing stage. During each iteration, the movement of fireflies is based on implicit backward Euler movement instead of randomly moving the fireflies.

[22] improved the optimization capability of conventional FA by incorporating various components of metaheuristic approaches in its standard operation. The various metaheuristic components embedded in standard FA included logistic and mouse/Gauss chaotic map, Levy flight and adaptive inertia weight. All these components were applied for various purposes e.g. initial solution generation, parameters updating and exploration and exploitation balancing. Likewise, [23] explored the effects of different control parameters on the performance of the standard FA and proposed an adaptive control parameter FA in which the control parameters of the standard FA possess the nature of adaptability to improve its solution quality. Although, all of these authors have tried to introduce some modifications in the operation of standard FA but failed to significantly improve the performance of the FA due to poor exploitation capability of FA during firefly position changing stage.

In addition to modified versions of FA, the solution quality of conventional FA is improved by balancing the exploration and exploitation by using FA in combination with other optimization algorithms. For example, [9] hybridized the firefly algorithm with the eagle search strategy (ES), which is a type of new metaheuristic search algorithm inspired by the foraging of eagle behaviors. The phenomenon behind the eagle search strategy is to catch the prey as efficiently as possible when it is seen. The eagle search consists of two components namely intensive local search and random search. In the proposed algorithm, the firefly algorithm was applied for local search and random search was performed using Levy flight. [24] used a hybrid approach of firefly algorithm with the differential equation for improving the searching accuracy and information sharing among different fireflies. Based on the fitness of fireflies, all the fireflies were divided into two subgroups. In the first group, the operators of the conventional firefly algorithms were applied where as in the second; the operators from the differential equations were applied.

[25] proposed a hybrid model of FA and pattern search for solving optimization problem of estimation of electrical circuit parameters for enhancing the performance of energy systems. The authors of the approach are in the opinion that FA is quite efficient in solving the optimization problems where more exploration of the solution search space is required for finding the most optimal solution. But, in order to improve the exploitation capability of the FA, some other algorithms have strong exploitation capability should be used in combination with standard FA. Therefore, they combined pattern search with this as the local optimizer to improve the exploitation capability that ultimately results in high quality solution.

[26] hybridized quantum evolutionary algorithm (QEA) with the standard FA for solving quality of service multicast routing problem. In their proposed model, the evolutionary equations of quantum algorithm were replaced by the evolutionary equation of FA. Based on the hybridization of QEA and FA, two approaches were proposed. The first one named FAQEA1, consisted of evolutionary equation of FA integrated into the operators of QEA. The second one named FAQEA2 consists of replacement of evolutionary equation of QEA by the evolutionary equation of FA. A hybrid model of FA and gravitational search algorithm (GSA) was proposed by [27] for solving task scheduling problem. The objective function was the cost associated with each virtual machine and there were different input parameters. The first step was to associate the input parameters with the fireflies and the cost with the light intensity of fireflies. The firefly with least cost is considered as the best firefly. After a fixed number of iterations, the best firefly is found out.

For parameters estimation in the micro grid, [28] proposed a hybrid model of FA and harmony search. For ensuring the faster convergence and endeavor the less randomization used in the standard FA, the diversification of the solution search space was increased by introducing the harmony search algorithm. For decreasing the searching delay in the local optima, exponential decay bandwidth and the linear incremental adjustment rates were considered in the hybrid model. For optimizing the parameters for the machining process, [29] developed a hybrid model of FA and PSO. The authors used FA for exploitation and the PSO for exploration of the solution search space. The results obtained were compared with standard FA and PSO in which the proposed approach outperformed both. In the same manner, [30] developed a hybrid model of FA and PSO to eliminate few drawbacks of standard

FA by using PSO. Initially, the population was divided into two sub population with one assigned to FA and the other to PSO. In the second phase, the solution search space is divided into many small sized regions which are used to help the current solution in finding the most optimal solution. Lastly, the exploitation capability of the FA is improved by introducing Quasi Newton method.

Similarly, [31] hybridized FA with ABC to improve its exploration and exploitation capability. The standard FA is incorporated in scout bee phase of the ABC algorithm to improve its solution quality. Likewise, [32] hybridized FA with PSO for solving complex numerical optimization problems. The author eliminated the drawbacks of both the SI algorithms and developed a hybrid model that can be used to solve complex optimization problems. For optimizing the dynamic quantities of four bar mechanism, [33] developed a hybrid model of Cuckoo search and FA. During searching of nests, the FA position changing is embedded into the cuckoo search algorithm to improve its searching process.

#### 4. Motivation

For quantitative and qualitative measurement of solution search space, the meta-heuristic algorithms use two terms namely exploration and exploitation. Exploration is used for bringing diversification in the solutions of the problem which in turn expands the solution search space to cover solutions with more differences. On the other hand, exploitation is used to contract the solutions of the problem by keeping less difference among different solutions in the solution search space. In its simplest form, the exploration means globalizing the solution search space whereas exploitation means localizing the solution search space [34]. For better performance of the optimization algorithms in terms of solution quality, these two properties must be balanced. If the solution search space is highly explorative and poorly exploitative, then the difference between the solution of the previous iteration and the current iteration will be high and the best optimal solution can be missed during iteration which could be obtained with less difference between the solutions. In each meta-heuristic algorithm, the process of searching the solution depends upon the balancing between exploration and exploitation [35].

FA is also meta-heuristic in nature which means that there is a trade-off between the exploration (Finding

new search space) and exploitation (Finding solution in local search space), which may lead to slow convergence rate that ultimately results in low quality solution. Different variants of FA and hybridization of FA with other optimization algorithms have also been proposed for balancing the relationship between the exploration and exploitation to solve various types of optimization problems but there is still much room for improvements.

### 5. Proposed Method

#### 5.1. Problem statement

The standard FA operates in three major stages namely initialization stage, firefly position changing stage and lastly, the termination stage. The initialization stage is used to provide the starting solution search space for the algorithm to proceed. The firefly position changing stage is used for updating the solutions provided by the initialization stage to update the solution search space in each iteration. The termination stage is used to end the process of optimization by getting the most optimal solution or the maximum number of iteration is reached. For updating the solution search space during firefly position changing stage and controlling the boundaries of the solution search space of the optimization problem, the randomization factor  $\hat{\alpha}$ , outlined in equation 5 plays the most important role. This factor plays two major roles in establishing the quality of solution in the solution search space i.e. finding new solutions (diversifying the solution space, exploration) and focusing on so far found solutions (exploitation). These two properties must be balanced for a successful search of the most optimal solution. If the solution of the problem is not improved during each iteration, then the exploration and exploitation properties are not balanced resulting in the low quality solution. If this factor is taken large, the solution search space will be highly explorative and poorly exploitative that leads to skip the most optimal solution present around the current solution.

#### 5.2. Conceptualization

In our proposed approach, this problem is resolved by introducing genetic algorithm operator named crossover operator in the firefly changing position which will improve the local search space resulting in high quality solution to the problem considered.

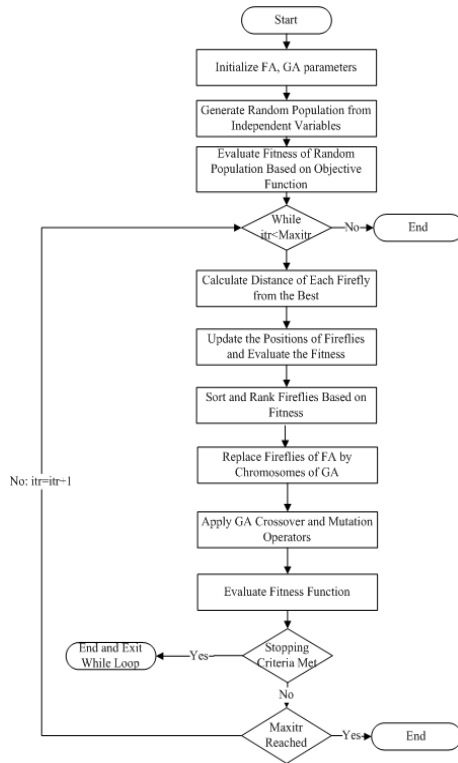


Fig. 1. Data flow diagram of proposed approach

The flow chart of the proposed approach is shown in Fig.???. The implementation of GA inside the FA involves few technical steps. The standard FA consists of fireflies whereas the standard GA has the concept of genes and chromosomes. To embed GA operators in FA, the first step is the representation of FA's fireflies by GA's chromosomes. Each firefly of FA becomes chromosome in GA and all the fireflies of FA represent the chromosomes of the population. To implement the crossover operator, the genes in the newly created chromosomes are interchanged according to ratio specified in the experimental setup. After these operations are performed on chromosomes, the fitness values of the optimization functions are evaluated. If the fitness value of the optimization functions is the required value, then the algorithm terminates otherwise the chromosomes are replaced by the fireflies in the next iteration and the process repeats until the maximum number of iteration reaches the termination criteria is met. Pseudocode for the proposed approach is shown in Algorithm .

#### Pseudocode Proposed Approach

- 1: **Begin**
- 2: Initialize the fireflies population
- 3: Evaluate the fireflies based on fitness function
- 4: **while** Stopping criteria do not meet **do**
- 5:  $i = 0$
- 6: Calculate the distance of each firefly from the best using Eq. 4
- 7: Update the fireflies positions using Eq. 5
- 8: Evaluate the fitness function
- 9: Sort and rank the fireflies according to fitness function values
- 10: Interchange the fireflies of FA and chromosomes of GA
- 11: Apply GA crossover operator
- 12:  $i + +$
- 13: **end while**

In fig. 1, *itr* means iteration whereas *Maxitr* means maximum number of iteration.

#### 5.3. Mathematical Model

In order to give an understandable and conceptual mathematical representation to the proposed approach, the flow of both FA and GA must be clear enough. The working mechanism of FA has been already presented in the previous sections and this section will first give the flow of standard GA and then the mathematical representation of the proposed method.

##### 5.3.1. GA working mechanism

GA is a search and optimization evolutionary algorithm that follows the genetics and natural selection principles described in details by [26]. Genetic algorithm is applied in the optimization problems where the search space is very large. Genetic algorithm uses an initial population of individuals for making a population of high quality individuals in which each individual represents a solution of the problem to be solved. The individuals are called chromosomes and their basic structural unit is gene. The value of fitness function (The function that needs to be optimized) associated with each chromosome determines the quality of fitness of each chromosome. There are some parameters of genetic algorithm that affect the performance of the algorithm according to different problems. These are: fitness function selection, individual representation, and the values of genetic algorithm parameters e.g. cross over, mutation rate, and population size and fitness value threshold. The process of opti-

mization using genetic algorithm takes place in steps shown in algorithm 1. In this algorithm,  $Pop$  is population size,  $CR$  is crossover rate,  $MR$  is mutation rate,  $PR$  is probability ratio,  $Maxitr$  is maximum iteration,  $FV$  represents fitness value,  $F1, F2, F3, \dots, Fn$  (the value of each function represents fitness value associated with each chromosome) represent the optimization functions and  $X1, X2, X3, \dots, Xn$  represent the variables present in the optimization functions (Each variable represents gene and all the variables collectively represent chromosomes). In our case, we have considered the following parameters and their values. The initial population selected was 100; one-point cross-over was performed with the mutation rate as 0.1% and the probability of 0.9%. When the genetic algorithm optimization process completed, the optimal values of the optimization functions were obtained by selecting the best fitted chromosome. The algorithm was run for 200 generations and stops in two cases; either the maximum generations reach or there is no significant change in the value of fitness function.  $F1, F2, F3, \dots, Fn$  (the value of each function represents light intensity value associated with each firefly) represent the optimization functions and  $X1, X2, X3, \dots, Xn$  represent the variables present in the optimization functions (all the variables collectively represent fireflies).

---

**Algorithm 1** Genetic Algorithm
 

---

- 1: **Parameters Initialization**
  - 2:  $Pop, CR, MR, PR, Maxitr$
  - 3: Generate random population of chromosomes  $Pt$  using  $X1, X2, \dots, Xn$  of Eq. 6, Eq. 7, Eq. 8, ... Eq. 19
  - 4: Put  $i = 0$
  - 5: **while**  $i < Maxitr$  **do**
  - 6: STEP1: Select parents from  $Pt$  to create offspring based on Eq. 6, Eq. 7, Eq. 8, , Eq. 19
  - 7: STEP2: Apply crossover operation using  $CR$
  - 8: STEP3: Apply mutation operation using  $MR$
  - 9: STEP4: Evaluate new  $FV$  values
  - 10: STEP5: Sort and rank  $FV$  values
  - 11: STEP6:  $i++$
  - 12: **end while**
  - 13: **return** The best solution
- 

### 5.3.2. FA working mechanism

The basic operational procedure of conventional FA is stepwise represented in Algorithm 2.

### 5.3.3. Proposed Model working mechanism

Let  $Ch1$  to  $Chi$  represent chromosomes in the population of  $GA$  and  $g1$  to  $gj$  represent genes in each chromosome of the population, then the whole population in the  $GA$  can be represented as:

$$\begin{aligned} Ch1 &= [g11, g12, g13, g14, \dots, g1j] = f(Obj1) \\ Ch2 &= [g21, g22, g23, g24, \dots, g2j] = f(Obj2) \\ Ch3 &= [g31, g32, g33, g34, \dots, g3j] = f(Obj3) \\ Ch4 &= [g41, g42, g43, g44, \dots, g4j] = f(Obj4) \\ Ch5 &= [g51, g52, g53, g54, \dots, g5j] = f(Obj5) \\ &: \\ Chi &= [gi1, gi2, gi3, gi4, \dots, gij] = f(Obji) \end{aligned}$$

In these notations,  $f(Obj1)$  to  $f(Obji)$  represents the value of objective function associated with each chromosome of the population. Now, let  $f1$  to  $fi$  represent fireflies in the  $FA$  population and  $p1$  to  $pj$  represent position of each firefly in the population, then the whole population of the  $FA$  can be represented as:

$$\begin{aligned} f1 &= [p11, p12, p13, p14, \dots, p1j] = I(1) \\ f2 &= [p21, p22, p23, p24, \dots, p2j] = I(2) \\ f3 &= [p31, p32, p33, p34, \dots, p3j] = I(3) \\ f4 &= [p41, p42, p43, p44, \dots, p4j] = I(4) \\ f5 &= [p51, p52, p53, p54, \dots, p5j] = I(5) \\ &: \\ fi &= [pi1, pi2, pi3, pi4, \dots, pij] = I(i) \end{aligned}$$

In these notations,  $I(1)$  to  $I(i)$  represent the light intensity associated with each firefly in  $FA$  population.

The first logical operation in the proposed modal is the assignment of fireflies of  $FA$  to chromosomes of  $GA$  in which each firefly of  $FA$  is represented by chromosome of  $GA$  and the value of objective function of  $GA$  is represented by light intensity of  $FA$ . This is achieved by following operation:

$$\begin{aligned} Ch1 &= f1, f(Obj1) = I(1) \\ Ch2 &= f2, f(Obj2) = I(2) \\ Ch3 &= f3, f(Obj3) = I(3) \\ Ch4 &= f4, f(Obj4) = I(4) \\ Ch5 &= f5, f(Obj5) = I(5) \\ &: \\ Chi &= fi \Rightarrow f(Obji) = I(i) \end{aligned}$$

Here, the light intensity of  $FA$  is represented by the objective function in  $GA$ . The next stage is applying crossover operator in the newly generated population. The crossover operator is logically represented as:

**Algorithm 2** Firefly Algorithm1: **Parameters Initialization**

- i Total number of fireflies in the algorithm:**  $No$
- ii Number of generations:**  $Max$
- iii The values of  $\beta_o, \gamma, \alpha, \epsilon$**
- iv Put the generation counter  $i = 0$**

- 2: Generate initial population of fireflies  $Pt$  using  $X1, X2, , Xn$  of Eq. 6, Eq. 7, Eq. 8, ... Eq. 19 representing light intensity values
- 3: Evaluate fitness values  $FV$  of Eq. 6, Eq. 7, Eq. 8, , Eq. 19 using randomly generated values of  $X1, X2, X3, , Xn$
- 4: Sort the fireflies according to the values of fitness function  $FV$
- 5: Find the best firefly from the sorted fireflies.
- 6: Put  $i = 0$
- 7: **while**  $i < Max$  **do**
- 8:   STEP1: Calculate the Euclidian distance  $r_{ij}$  of each firefly from all fireflies using eq (4)
- 9:   STEP2: Calculate the attractiveness  $\beta$  of each firefly for other fireflies using Eq. 3
- 10:   STEP3: Update position  $X_i$  of each firefly using Eq.5
- 11:   STEP4: Evaluate new  $FV$  values associated with each firefly
- 12:   STEP5: Sort and rank the updated population for values of  $FV$
- 13:   STEP6:  $i + +$
- 14: **end while**
- 15: **return** The best solution

$$Ch1 >< Ch2(Crossoveratposition1)$$

$$Ch1 = f1 = [p21, p12, p13, p14, , p1j] = f(Obj1)$$

$$Ch2 >< Ch4(Crossoveratposition3)$$

$$Ch2 = f2 = [p21, p22, p43, p24, , p2j] = f(Obj2)$$

$$Chi >< Ch2(Crossoveratpositioni)$$

$$Chi = fi = [pi1, pi2, pi3, pi4, , pi2j] = f(Obji)$$

The next stage is the assignment of chromosomes of this newly created population of GA by fireflies of FA.

#### 5.4. Model Performance Evaluation

The performance of the proposed model was measured using convergence rate, best case solutions, average case solutions, worst case solution and the standard deviation.

## 6. Experimental results and discussion

All the experiments were carried out on Intel(R) core(TM) i5-3570 CPU 3.40 GHz with MATLAB R2013a. Different parameters of all the considered algorithms are given in Table 1.

### 6.1. Minimization Functions

Total of eight minimization benchmark functions were used. These functions are outlined in Eq. 6-13 and their convergence rates are shown in Fig. 3-10.

$$F1 = e^{-(x1-4)^2-(x2-4)^2} + e^{-(x1-4)^2-(x2-4)^2} + 2e^{-(x1)^2-(x2+4)^2} + 2e^{-(x1)^2-(x2)^2} \quad (6)$$

where  $X_i \in -5, 5$

$$F2 = (|x1| - 5) + (|x2| + 5)^2 \quad (7)$$

where  $X_i \in -10, 10$

$$F3 = 100 * (x2 - x1^2)^2 + (6.4 * (x2 - 0.5)^2 - x1 - 0.6)^2 \quad (8)$$

where  $X_i \in -5, 5$

$$F4 = 100 * (x2 - x1^2)^2 + (x1 - 1)^2 \quad (9)$$

where  $X_i \in -2.048, 2.048$



**Algorithm 3** Proposed Algorithm

- 
- 1: **Parameters Initialization**
    - i **Total number of fireflies in the algorithm:**  $No$
    - ii **Number of generations:**  $Max$
    - iii **The values of**  $\beta_o, \gamma, \alpha, \epsilon$
    - iv **Put the generation counter**  $i = 0$
  - 2: GA Parameters Initialization  
 $Pop, CR, MR, PR, Maxitr$
  - 3: Generate initial population  $Pt$  using  $X1, X2, , Xn$  of Eq. 6, Eq. 7, Eq. 8, ... Eq. 19
  - 4: Evaluate fitness values  $FV$  of Eq. 6, Eq. 7, Eq. 8, , Eq. 19 using randomly generated values of  $X1, X2, X3, , Xn$
  - 5: Sort the fireflies according to the values of fitness function  $FV$
  - 6: Find the best firefly from the sorted fireflies.
  - 7: Put  $i = 0$
  - 8: **while**  $i < Max$  **do**
    - 9: STEP1: Calculate the Euclidian distance  $r_{ij}$  of each firefly from all fireflies using Eq. 4
    - 10: STEP2: Calculate the attractiveness  $\beta$  of each firefly for other fireflies using Eq. 3
    - 11: STEP3: Update position  $X_i$  of each firefly using Eq. 5
    - 12: STEP4: Replace fireflies of FA by chromosomes of GA
    - 13: STEP5: Apply crossover operator with specific crossover rate
    - 14: STEP6: Replace the chromosomes back by fireflies
    - 15: STEP7: Evaluate new  $FV$  values associated with each firefly
    - 16: STEP8: Sort and rank the updated population for values of  $FV$
    - 17: STEP9:  $i + +$
  - 18: **end while**
  - 19: **return** The best solution
- 

$$F5 = (x2 - \frac{5.1}{4\pi^2}(x1)^2 + \frac{5}{\pi}x1 - 6)^2 + 10(1 - \frac{1}{8\pi})\cos x1 + 10 \quad (10)$$

where  $X_i \in -5, 15$

$$F6 = (x1 + 2x2 - 7)^2 - \cos(18(x1)) - \cos(18(x2)) \quad (11)$$

where  $X_i \in -1, 1$

$$F7 = (x1 + 2x2 - 7)^2 + (2(x1)(x2) - 5)^2 \quad (12)$$

where  $X_i \in -10, 10$

$$F8 = 4(x1)^2 + 2.1(x1)^4 + \frac{1}{3}x1^6 + (x1)(x2) - 4(x1)^2 + 4(x1)^4 \quad (13)$$

where  $X_i \in -5, 5$

As shown in Fig. 2 to 9 for minimization functions, the proposed approach outperforms standard FA, standard GA, ABC and ACO in terms of convergence rates. As shown in Fig. 2, there are many fluctuations in the convergence graph of FA-CROSSOVER approach before 150th iteration and after this, the convergence rate of proposed model is more qualitative than the FA, GA, ABC and ACO. In Fig.3, the same fluctuations can be observed but after iteration 200th, the solution quality of proposed model is better than all the other considered algorithms. Similar types of observations can be found Fig. 4 to 9 with the solution quality getting better after different iteration number.

Table 1  
Parameters tuning of all considered approaches

Algorithm	Parameters Names	Parameters Values
FA	Maximum Iterations	100-400 with increment of 20
	Initial Population Size	40-100 with increment of 5
	Gamma	0.5 to 2 (0.5 increment)
	Beta	1 to 4 (1 increment)
	Alpha	0.1 to 0.5 (0.1 increment)
GA	Maximum Iterations	100-400 with increment of 20
	Initial Population Size	40-100 with increment of 5
	Cross Over Type	One Point Cross Over
	Cross Over Probability	0.5 to 0.9
	Mutation Rate	0.1 to 0.3
ABC	Maximum Iterations	100-400 with increment of 20
	Colony Size	100-400 with increment of 20
	Employed Bees	50 ~ 200 (Half of colony size)
	Onlooker Bees	50 ~ 200 (Half of colony size)
ACO	Maximum Iterations	100-400 with increment of 20
	No of ants	50-150 with increment of 10
	Randomness parameter ( $q_0$ )	0.6 to 0.9 (0.1 increment)
	Evaporation rate ( $p$ )	0.1 to 0.5 (0.1 increment)
	Pheromone Control Parameter ( $\alpha$ )	1 to 5 (1 increment)
	Heuristic information Parameter ( $\beta$ )	1 to 5 (1 increment)
Proposed Method	Maximum Iterations	100-400 with increment of 20
	Initial Population Size	40-100 with increment of 5
	Gamma	0.5 to 2 (0.5 increment)
	Beta	1 to 4 (1 increment)
	Alpha	0.1 to 0.5 (0.1 increment)
	Cross Over Type	One Point Cross Over
	Cross Over Probability	0.5 to 0.9

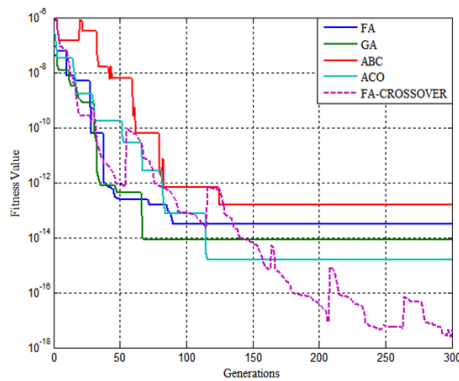


Fig. 2. F1 minimization function convergence rate

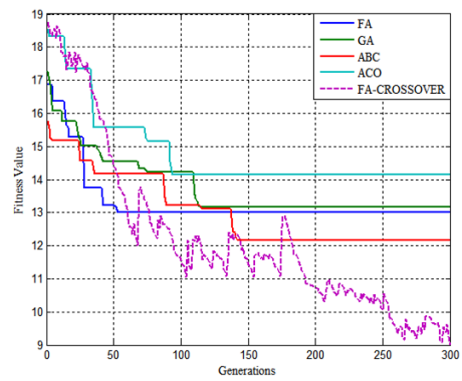


Fig. 3. F2 minimization function convergence rate

As shown in Table 2, the best case solution, average case solution and the worst case solution of proposed model for all the minimization functions are better than FA, GA, ABC and ACO whereas in standard deviation, the value of proposed model is worse than other

considered models. The variations in results are due to various parameters resulting in the conversion rates of all algorithms. The parameter tuning has followed the trail and error mechanism, as there are no fixed rules for setting these parameters.

Table 2  
Performance Evaluation of Algorithms for Minimization Functions

Function	Technique	Worst Case Solution	Average Case Solution	Best Case Solution	Standard Deviation
F1	FA	6.00E-11	3.41E-12	7.20E-14	1.33234E-11
	GA	9.76E-10	1.77E-10	8.43E-15	3.48657E-10
	ABC	6.78E-07	7.08E-08	6.67E-13	1.55205E-07
	ACO	6.68E-10	7.89E-11	7.67E-15	1.86056E-10
	FA-CROSS	8.56E-12	4.78E-13	6.45E-18	1.91059E-12
F2	FA	1.40E+01	1.31E+01	1.30E+01	2.78E-01
	GA	1.47E+01	1.37E+01	1.32E+01	4.54E-01
	ABC	1.42E+01	1.32E+01	1.42E+01	5.56E-01
	ACO	1.52E+01	1.48E+01	1.42E+01	2.75E-01
	FA-CROSS	1.17E+01	1.02E+01	9.02E+00	6.60E-01
F3	FA	9.75E-02	8.77E-02	7.17E-02	8.01E-03
	GA	9.66E-01	3.17E-01	7.54E-02	3.41E-01
	ABC	9.99E-01	1.81E-01	3.54E-02	2.61E-01
	ACO	8.34E-01	2.07E-01	3.54E-02	2.61E-01
	FA-CROSS	6.90E-03	5.91E-04	6.57E-05	1.52E-03
F4	FA	4.35E+00	1.83E+00	1.67E-03	1.27E+00
	GA	4.35E+00	1.94E+00	2.45E-03	1.48E+00
	ABC	5.35E+00	2.51E+00	6.89E-03	1.04E+00
	ACO	4.55E+00	2.83E+00	6.89E-03	1.21E+00
	FA-CROSS	8.56E-02	6.31E-03	6.56E-05	1.95E-02
F5	FA	1.34E+01	6.38E+00	1.68E-02	4.24E+00
	GA	9.57E+00	3.52E+00	1.75E-03	3.56E+00
	ABC	2.24E+01	8.91E+00	2.23E-01	7.08E+00
	ACO	7.45E+00	3.16E+00	2.23E-01	2.27E+00
	FA-CROSS	3.24E+00	8.79E-01	2.68E-04	1.07E+00
F6	FA	-1.86E+00	-1.92E+00	-1.98E+00	3.85E-02
	GA	-1.80E+00	-2.09E+00	-2.33E+00	1.60E-01
	ABC	-1.86E+00	-2.12E+00	-2.15E+00	9.42E-02
	ACO	-1.83E+00	-1.94E+00	-2.15E+00	8.63E-02
	FA-CROSS	2.29E+00	-2.38E+00	-2.88E+00	1.10E+00
F7	FA	2.46E+01	1.30E+01	8.57E+00	3.58E+00
	GA	1.24E+01	8.94E+00	5.79E+00	2.02E+00
	ABC	1.05E+01	7.46E+00	7.22E+00	1.85E+00
	ACO	1.67E+01	1.23E+01	7.22E+00	2.18E+00
	FA-CROSS	7.57E+00	4.91E+00	1.66E+00	1.55E+00
F8	FA	-8.74E-01	-9.74E-01	-1.30E+00	1.01E-01
	GA	-1.23E+00	-1.28E+00	-1.38E+00	3.26E-02
	ABC	-8.44E-01	-9.16E-01	-1.28E+00	7.55E-02
	ACO	1.18E+00	-1.02E+00	-1.28E+00	5.27E-01
	FA-CROSS	-1.44E+00	-1.57E+00	-1.62E+00	4.52E-02

## 6.2. Maximization Functions

Eq. 14 to 19 show the mathematical formulas of the maximization functions used in the experimentation and their convergence rates are shown in fig. 10 to 15

$$F9 = -(20 + (x_1^2 - 10 * \cos(2\pi x_1)) + (x_2^2 - 10 * \cos(2\pi x_2))) \quad (14)$$

where:  $x_i \in [-2.048, 2.048]$

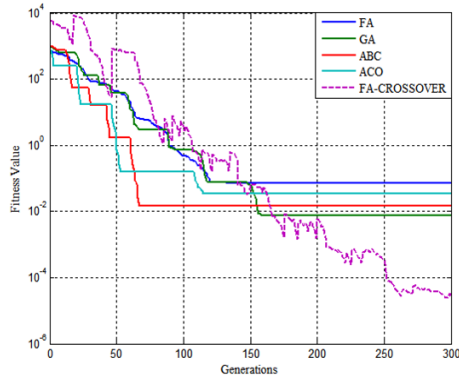


Fig. 4. F3 minimization function convergence rate

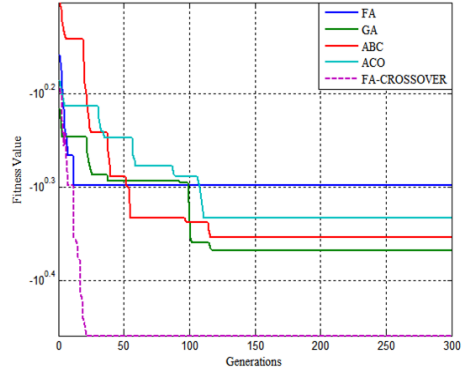


Fig. 7. F6 minimization function convergence rate

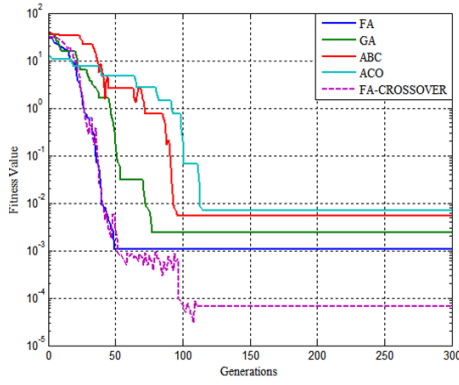


Fig. 5. F4 minimization function convergence rate

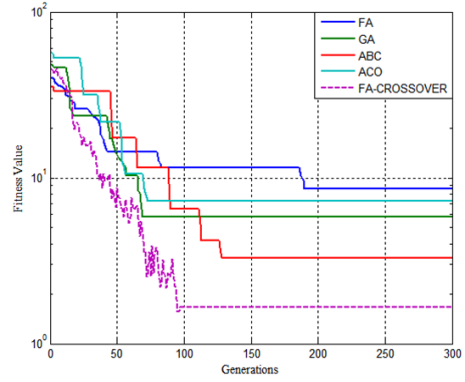


Fig. 8. F7 minimization function convergence rate

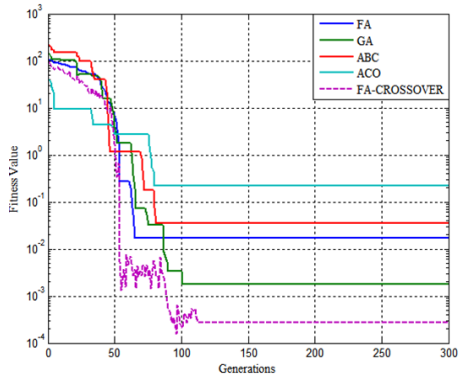


Fig. 6. F5 minimization function convergence rate

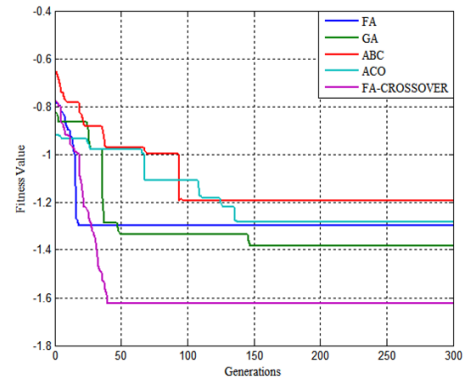


Fig. 9. F8 minimization function convergence rate

$$F10 = \cos x_1 * \cos x_2 e^{(x_1 - \Pi)^2 - (x_2 - \Pi)^2} \quad (15)$$

where:  $x_i \in [-20, 20]$

$$F11 = 280 - \frac{(x_1^4 - 16x_1^2 - 5x_1)}{2} - \frac{(x_2^4 - 16x_2^2 - 5x_2)}{2} \quad (16)$$

where:  $x_i \in -5, 5$

$$F12 = \ln[(\sin(\cos x1 + \cos x2)^2)^2 - (\cos(\sin x1 + \sin x2)^2 + x1)^2 - 0.1((x1 - 1)^2 + (x2 - 1)^2) \quad (17)$$

where:  $x_i \in -10, 10$

$$F13 = \left[ \sum_{i=1}^5 i \cos(i+1)x1 + i \right] \left[ \sum_{i=1}^5 i \cos(i+1)x2 + i \right] \quad (18)$$

where:  $x_i \in -10, 10$

$$F14 = 660 - ((x1^2) + x2 - 11)^2 - (x1 + (x2)^2 - 7)^2 \quad (19)$$

where:  $x_i \in -6, 6$

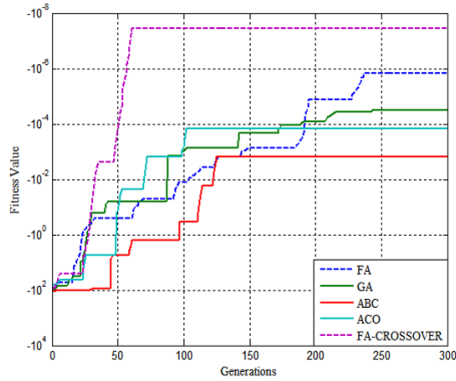


Fig. 10. F9 maximization function convergence rate

As shown in Fig.10 to 15, there are fluctuations in the solution quality of the proposed model before some iterations but later the convergence rate of the proposed model is better than all the other algorithms.

As shown in Table 3, the best case solution, average case solution and the worst case solution of proposed model for all the maximization functions are better than FA, GA, ABC and ACO whereas in standard deviation, the value of proposed model is worse than other considered models.

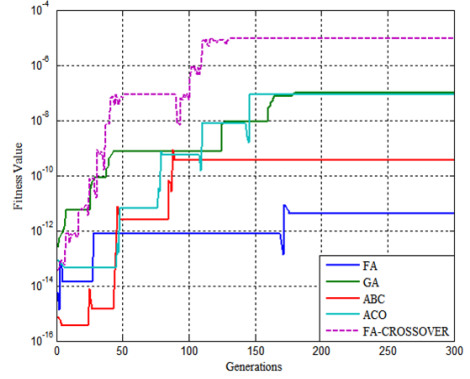


Fig. 11. F10 maximization function convergence rate

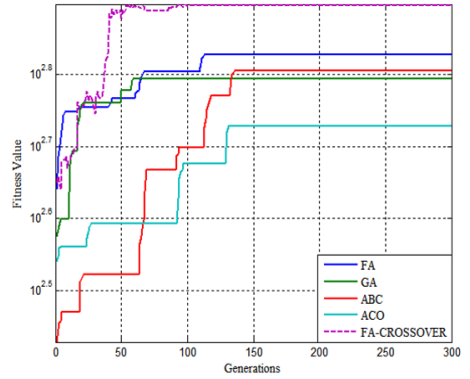


Fig. 12. F11 maximization function convergence rate

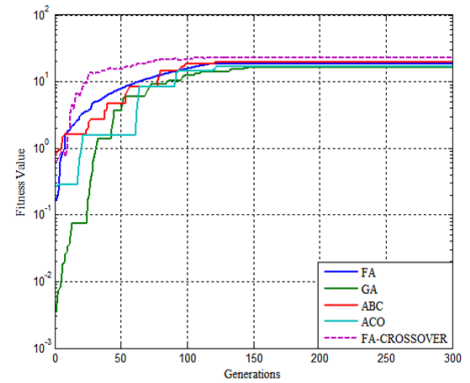


Fig. 13. F12 maximization function convergence rate

Concluding the overall development, representation, operation and the detailed implementation of the hybrid model of FA and GA, the whole research work carried out in this study is summarized in this section. The data flow diagram of the approach is shown in figure 1 and its pseudo code is shown pseudo code. The FA algorithm is shown in algorithm 1 whereas the al-

Table 3  
Performance Evaluation of Algorithms for Maximization Functions

Function	Technique	Worst Case Solution	Average Case Solution	Best Case Solution	Standard Deviation
F9	FA	-1.45E-06	-3.34E-05	-5.62E-04	1.24E-04
	GA	-3.28E-05	-1.46E-04	-4.68E-04	1.29E-04
	ABC	-1.56E-03	-1.10E-02	-4.55E-03	9.65E-03
	ACO	-1.55E-04	-1.30E-03	-4.55E-03	1.36E-03
	FA-CROSS	-3.57E-08	-1.42E-07	-6.57E-07	1.81E-07
F10	FA	6.35E-12	1.58E-12	3.57E-13	1.77E-12
	GA	9.84E-08	2.88E-08	4.55E-09	2.89E-08
	ABC	9.97E-10	5.46E-10	3.57E-09	3.45E-10
	ACO	9.67E-08	3.33E-08	3.57E-09	3.64E-08
	FA-CROSS	9.57E-06	3.73E-06	3.57E-07	3.66E-06
F11	FA	6.01E+02	6.44E+02	6.72E+02	2.08E+01
	GA	5.71E+02	5.94E+02	6.23E+02	1.48E+01
	ABC	5.76E+02	6.06E+02	5.34E+02	1.85E+01
	ACO	4.75E+02	5.05E+02	5.34E+02	2.04E+01
	FA-CROSS	7.65E+02	7.77E+02	7.87E+02	5.87E+00
F12	FA	1.37E+01	1.63E+01	1.84E+01	1.38E+00
	GA	1.07E+01	1.40E+01	1.64E+01	1.77E+00
	ABC	1.26E+01	1.61E+01	1.69E+01	1.90E+00
	ACO	1.07E+01	1.39E+01	1.69E+01	1.84E+00
	FA-CROSS	1.84E+01	2.00E+01	2.27E+01	1.33E+00
F13	FA	8.67E+01	9.94E+01	1.10E+02	7.67E+00
	GA	1.74E+01	1.08E+02	1.21E+02	2.24E+01
	ABC	9.57E+01	1.14E+02	1.22E+02	7.40E+00
	ACO	1.04E+02	1.11E+02	1.22E+02	6.15E+00
	FA-CROSS	1.24E+02	1.28E+02	1.31E+02	2.19E+00
F14	FA	8.14E+02	8.19E+02	8.25E+02	3.05E+00
	GA	8.18E+02	8.22E+02	8.27E+02	2.78E+00
	ABC	8.14E+02	8.18E+02	8.24E+02	2.23E+00
	ACO	8.15E+02	8.19E+02	8.24E+02	2.28E+00
	FA-CROSS	8.27E+02	8.30E+02	8.33E+02	1.90E+00

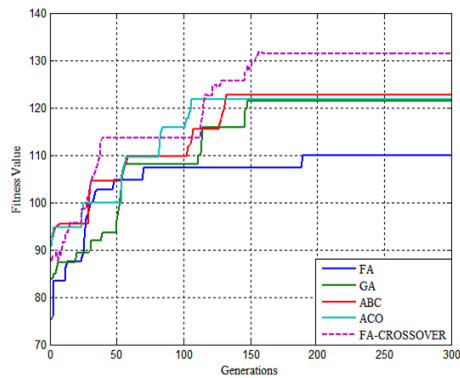


Fig. 14. F13 maximization function convergence rate

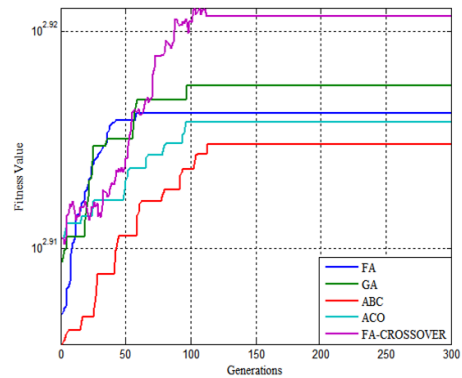


Fig. 15. F14 maximization function convergence rate

gorithm of GA is presented in algorithm 2. The proposed model is given in algorithm 3. The proposed

model was tested on eight minimization and six maximization functions outlined in Eq. 6 to 19.

## 7. Conclusion

In this paper, a major drawback associated with standard firefly algorithm (FA) that degrades its solution quality has been resolved by introducing a rich operator from genetic algorithm (GA). The standard firefly algorithm is suffered from poor exploitation capability if the randomization factor during firefly position changing stage is taken large. By taking this factor large, the solution search space is highly explorative resulting in larger solution search space with each solution having high differences in the optimal values with other solutions in the solution search space. During each iteration, the difference between the current solution and the previous solution may be large causing the solution search space to cover wider range of solutions with each solution apparently different from other solutions. This issue leads to skipping the most optimal solution even present in the vicinities of so far found solutions. This is resolved by introducing crossover operator of genetic algorithm which is used as exploitative operator to exchange the solutions found to cover more solutions in the current solution search space with little differences. This leads to avoid the skipping of solutions present in the vicinities of the current solution due to exchanges of many values with each other. The implementation of GA inside the FA involves few technical steps. The standard FA consists of fireflies whereas the standard GA has the concept of genes and chromosomes. To embed GA operators in FA, the first step is the representation of FA's fireflies by GA's chromosomes. Each firefly of FA becomes chromosome in GA and all the fireflies of FA represent the chromosomes of the population. To implement the crossover operator, the genes in the newly created chromosomes are interchanged according to ratio specified in the experimental setup. After these operations are performed on chromosomes, the fitness values of the optimization functions are evaluated. If the fitness value of the optimization functions is the required value, then the algorithm terminates otherwise the chromosomes are replaced by the fireflies in the next iteration and the process repeats until the maximum number of iteration reaches the termination criteria is met.

## References

- [1] Christian Blum and Xiaodong Li. Swarm intelligence in optimization. In *Swarm Intelligence*, pages 43–85. Springer, 2008.
- [2] Madeleine Beekman, Gregory A Sword, and Stephen J Simpson. Biological foundations of swarm intelligence. In *Swarm intelligence*, pages 3–41. Springer, 2008.
- [3] Gerardo Beni and Jing Wang. Swarm intelligence in cellular robotic systems. In *Robots and Biological Systems: Towards a New Bionics?*, pages 703–712. Springer, 1993.
- [4] Marco Dorigo. Ant colony optimization. *Scholarpedia*, 2(3):1461, 2007.
- [5] James Kennedy and Russell C Eberhart. The particle swarm: social adaptation in information-processing systems. In *New ideas in optimization*, pages 379–388. McGraw-Hill Ltd., UK, 1999.
- [6] Peter Korošec, Jurij Šilc, and Bogdan Filipič. The differential ant-stigmergy algorithm. *Information Sciences*, 192:82–97, 2012.
- [7] Abdolreza Hatamlou, Salwani Abdullah, and Hossein Nezamabadi-Pour. A combined approach for clustering based on k-means and gravitational search algorithms. *Swarm and Evolutionary Computation*, 6:47–52, 2012.
- [8] Amir Hossein Gandomi and Amir Hossein Alavi. Krill herd: a new bio-inspired optimization algorithm. *Communications in Nonlinear Science and Numerical Simulation*, 17(12):4831–4845, 2012.
- [9] Xin-She Yang and Suash Deb. Eagle strategy using lévy walk and firefly algorithms for stochastic optimization. In *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, pages 101–111. Springer, 2010.
- [10] Iztok Fister, Xin-She Yang, and Dušan Fister. Firefly algorithm: a brief review of the expanding literature. In *Cuckoo Search and Firefly Algorithm*, pages 347–360. Springer, 2014.
- [11] Shen Yin, Yuchen Jiang, Yang Tian, and Okayay Kaynak. A data-driven fuzzy information granulation approach for freight volume forecasting. *IEEE Transactions on Industrial Electronics*, 64(2):1447–1456, 2017.
- [12] Ravinesh C Deo, Mohammad Ali Ghorbani, Saeed Samadianfar, Tek Maraseni, Mehmet Bilgili, and Mustafa Biazar. Multi-layer perceptron hybrid model integrated with the firefly optimizer algorithm for windspeed prediction of target site using a limited set of neighboring reference station data. *Renewable energy*, 116:309–323, 2018.
- [13] Mohammad Ali Ghorbani, Shahaboddin Shamsirband, Davoud Zare Haghi, Atefe Azani, Hossein Bonakdari, and Isa Ebtehaj. Application of firefly algorithm-based support vector machines for prediction of field capacity and permanent wilting point. *Soil and Tillage Research*, 172:32–38, 2017.
- [14] Surafel Lulseged Tilahun and Hong Choon Ong. Modified firefly algorithm. *Journal of Applied Mathematics*, page 12, 2012.
- [15] Aref Yelghi and Cemal Köse. A modified firefly algorithm for global minimum optimization. *Applied Soft Computing*, 62:29–44, 2018.
- [16] Dac-Khuong Bui, Tuan Nguyen, Jui-Sheng Chou, H Nguyen-Xuan, and Tuan Duc Ngo. A modified firefly algorithm-artificial neural network expert system for predicting compressive and tensile strength of high-performance concrete. *Construction and Building Materials*, 180:320–333, 2018.
- [17] Lifang He and Songwei Huang. Modified firefly algorithm based multilevel thresholding for color image segmentation. *Neurocomputing*, 240:152–174, 2017.
- [18] Hui Wang, Wenjun Wang, Laizhong Cui, Hui Sun, Jia Zhao, Yun Wang, and Yu Xue. A hybrid multi-objective firefly algo-

- rithm for big data optimization. *Applied Soft Computing*, 69: 806–815, 2018.
- [19] Urvinder Singh and Rohit Salgotra. Synthesis of linear antenna arrays using enhanced firefly algorithm. *Arabian Journal for Science and Engineering*, pages 1–16, 2018.
- [20] Moath Sababha, Mohamed Zohdy, and Maged Kafafy. The enhanced firefly algorithm based on modified exploitation and exploration mechanism. *Electronics*, 7(8):132, 2018.
- [21] Ronald Bartz, Sierk Fiebig, Thilo Franke, Paul Falkenberg, and Joachim Axmann. Enhanced firefly algorithm with implicit movement. In *World Congress of Structural and Multidisciplinary Optimisation*, pages 700–709. Springer, 2017.
- [22] Jui-Sheng Chou and Ngoc-Tri Ngo. Modified firefly algorithm for multidimensional optimization in structural design problems. *Structural and Multidisciplinary Optimization*, 55(6): 2013–2028, 2017.
- [23] Hui Wang, Xinyu Zhou, Hui Sun, Xiang Yu, Jia Zhao, Hai Zhang, and Laizhong Cui. Firefly algorithm with adaptive control parameters. *Soft computing*, 21(17):5091–5102, 2017.
- [24] Afnizanfaizal Abdullah, Safaai Deris, Mohd Saberi Mohamad, and Siti Zaiton Mohd Hashim. A new hybrid firefly algorithm for complex and nonlinear problem. In *Distributed Computing and Artificial Intelligence*, pages 673–680. Springer, 2012.
- [25] Amir Mohammad Beigi and Ali Maroosi. Parameter identification for solar cells and module using a hybrid firefly and pattern search algorithms. *Solar Energy*, 171:435–446, 2018.
- [26] Mohammed Mahseur, Yassine Meraihi, Abdelmadjid Boukra, and Amar Ramdane-Cherif. Qos multicast routing based on a hybrid quantum evolutionary algorithm with firefly algorithm. In *Electrical Engineering-Boumerdes (ICEE-B), 2017 5th International Conference on*, pages 1–6. IEEE, 2017.
- [27] Kanika Sood, Anurag Jain, and Amit Verma. A hybrid task scheduling approach using firefly algorithm and gravitational search algorithm. In *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, pages 2997–3002. IEEE, 2017.
- [28] Prachitara Satapathy, Snehamoy Dhar, and Pradipta Kishore Dash. Stability improvement of pv-bess diesel generator-based microgrid with a new modified harmony search-based hybrid firefly algorithm. *IET Renewable Power Generation*, 11(5): 566–577, 2017.
- [29] Nur Farahlina Johari, Azlan Mohd Zain, Noorfa Haszlinna Mustaffa, and Amirmudin Udin. Machining parameters optimization using hybrid firefly algorithm and particle swarm optimization. In *Journal of Physics: Conference Series*, volume 892, page 012005. IOP Publishing, 2017.
- [30] Xuewen Xia, Ling Gui, Guoliang He, Chengwang Xie, Bo Wei, Ying Xing, Rui Feng Wu, and Yichao Tang. A hybrid optimizer based on firefly algorithm and particle swarm optimization algorithm. *Journal of computational science*, 26:488–500, 2018.
- [31] Sanjivani Kale and Yelisetty SSR Murthy. Hybrid firefly algorithm based regression test case prioritisation. *International Journal of Business Intelligence and Data Mining*, 12(4):340–357, 2017.
- [32] İbrahim Berkan Aydılek. A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems. *Applied Soft Computing*, 66:232–249, 2018.
- [33] Marina Bošković, Radovan R Bulatović, Slaviša Šalinić, Goran R Miodragović, and Gordana M Bogdanović. Optimization of dynamic quantities of a four-bar mechanism using the hybrid cuckoo search and firefly algorithm (h-cs-fa). *Archive of Applied Mechanics*, pages 1–22, 2018.
- [34] Katerina Tashkova, Jurij Šilc, Nataša Atanasova, and Sašo Džeroski. Parameter estimation in a nonlinear dynamic model of an aquatic ecosystem with meta-heuristic optimization. *Ecological Modelling*, 226:36–61, 2012.
- [35] Matej Črepinšek, Marjan Mernik, and Shih-Hsi Liu Liu. Analysis of exploration and exploitation in evolutionary algorithms by ancestry trees. *International Journal of Innovative Computing and Applications*, 3(1):11–19, 2011.