

AEFA: Artificial electric field algorithm for global optimization

Anita^a, Anupam Yadav^{b,*}



^a Department of Sciences and Humanities, National Institute of Technology Uttarakhand, Srinagar, 246174, Uttarakhand, India
^b Department of Mathematics, Dr. B.R. Ambedkar, National Institute of Technology Jalandhar, Jalandhar, 144011, Punjab, India

ARTICLE INFO

Keywords:
 Optimization
 Soft computing
 Artificial intelligence
 Electric force

ABSTRACT

Electrostatic Force is one of the fundamental force of physical world. The concept of electric field and charged particles provide us a strong theory for the working force of attraction or repulsion between two charged particles. In the recent years many heuristic optimization algorithms are proposed based on natural phenomenon. The current article proposes a novel artificial electric field algorithm (AEFA) which inspired by the Coulomb's law of electrostatic force. The AEFA has been designed to work as a population based optimization algorithm, the concept of charge is extended to fitness value of the population in an innovative way. The proposed AEFA has been tested over a newly and challenging state-of-the-art optimization problems. The theoretical convergence of the proposed AEFA is also established along with statistical validation and comparison with recent state-of-the-art optimization algorithms. The presented study and findings suggests that the proposed AEFA as an outstanding optimization algorithms for non linear optimization.

1. Introduction

Heuristic and metaheuristics are two classes of optimization algorithms available in the literature. Heuristic optimization algorithms are problem dependent and gradient based. These algorithm stuck in local optima and fail to fetch the global optima or even a near optimal solution for highly non-linear and complex problems. Therefore the second class of optimization algorithms metaheuristic become more popular in recent years. The metaheuristic algorithms are problem independent, easy to implement and have good computation power. Metaheuristic algorithms avoid the stagnation in local solutions and search the entire search space extensively. Generally the proposed metaheuristic algorithms are divided into seven groups: swarm, biology, physics, social, music, chemistry and sport based.

Eberhart and Kennedy presents first swarm based optimization technique PSO [1] inspired from the social behavior of bird flocking or fish schooling. Ant colony optimization ACO proposed by Dorigo is another swarm based technique which mimics the behavior of ants in finding paths from the colony to the food [2]. D. Karaboga presents another swarm based techniques artificial bee colony (ABC). This technique works on the swarming behavior of bees around their hive [3]. Inspired by the migration behavior of monarch butterflies in the northern region an another swarm based technique MBO is proposed by G.G.Wang. In MBO the butterflies of two different land 1 and 2 are

updated by a migration operator and a butterfly adjusting operator [4]. Similarly Yang and Deb presents another swarm based technique Cuckoo Search (CS). CS works on the obligate brood parasitism of some cuckoo species [5]. This algorithm is enhanced by the Levy flight behavior of some birds and fruit flies rather than by simple isotropic random walks.

Genetic algorithm (GA), differential evolution (DE) and biogeography based optimization (BBO) are biology based optimization algorithms. The first biology based algorithm GA is intelligent probabilistic search algorithm proposed by Holland and inspired by the Darwin's theory of evolution according to which the best fittest survive and worst will die [6]. A simple GA is composed of three operations: natural selection, genetic operation, and replacement. The scheme of differential evolution algorithm (DE) proposed by Storn and Price [7] entirely corresponds to a typical GA. The principle difference consists in their mutation operation. In DE Mutation is caused by combinations of individuals, whereas in GA mutation is provided by the small alterations of genes. D Simon [8] presents another biology based algorithm BBO which works on the geographical distribution of organisms. To share the information among solutions BBO mainly uses the biogeography-based migration operator.

Teaching-learning based optimization (TLBO) and imperialist competitive algorithm (ICA) are social based optimization methods. Teaching learning based Optimization (TLBO) [9] works on the effect of the

* Corresponding author.

E-mail addresses: annusajwan93@gmail.com (Anita), anupuam@gmail.com (A. Yadav).

influence of a teacher on the output of learners in a class and ICA based on the human socio-political evolution process [10].

Gravitational search algorithm (GSA), charged system search (CSS), magnetic optimization algorithm (MOA) and intelligent water drops algorithms are physics based optimization algorithms. GSA [11] proposed by Rashedi works on the principal of gravitational force acting between two objects. CSS implements some principles from physics and mechanics [12]. MOA based on the principles of magnetic field theory [13]. Water drops algorithm based on the dynamic of river systems, actions and reactions that happen among the water drops and the water drops with the river beds [14].

Z.W. Geem proposed music based harmony search algorithm (HSA), which mimics the music improvisation in a music player [15]. League championship is a sports based optimization method presented by A H Kashan and based on the competition of sport teams in a sport league [16]. Lam and Li and B Alatas presents the chemical based optimization algorithms chemical reaction optimization (CRO) and artificial chemical reaction optimization algorithm (ACROA) respectively. CRO inspired from the interactions of molecules in a chemical reaction to reach a low energy stable state [17] and ACROA mimics the types and occurring of chemical reactions [18]. Cultural algorithm (CA) [19] and colonial competitive differential evolution [20] are both biology and social based algorithms.

All the above mentioned algorithms have some limitations in one or other aspect and the main reason for this is that different parameters are required for proper working of these algorithms. For the searching of global optimum solution by all these algorithms the proper selection of parameter is essential. A change in the parameter can change the effectiveness of the algorithm. Due to this fact the research is continued to enhance the existing algorithms to suit particular applications. The enhancement is done either by hybridizing the existing algorithms or by modifying the existing algorithms. The enhancement done by modifications is reported in PSO [21–24], ACO [25], ABC [26], DE [27–30], GSA [31,32], HS [33], etc. The enhancements can also be done by hybridization of the existing algorithms. Hybridization combined the best strategies of different algorithms and it is an effective way to make an algorithm efficient. Some hybridized algorithms are [34–37].

These algorithms solve different optimization problems but in general there does not exist any optimization algorithm which can solve all sorts of optimization problems [38]. Therefore the design and analysis of new optimization algorithms always remains an active topic of research [39].

The basic idea of AEFA is inspired by the fundamental electrostatic force. The organization of the article is presented in the following manner: section 2 gives the brief review of the fundamental laws of electrostatic field. In Section 3 the idea of AEFA is proposed along with the details of its inspiration from electrostatic field theory. Section 4 presents the very basic difference in the idea of AEFA and CSS, MOA, PSO and GSA respectively. In section 5, detailed analysis of AEFA is provided along with theoretical convergence. Section 6 demonstrates the experimental and theoretical results of AEFA along with comparison with other state-of-the-algorithms.

2. Physical laws

In physics, the space surrounding by charge objects creates an electric field around it, which exerts a force on other charged objects. The electric field surrounding a charge objects is given by Coulomb's law of electrostatic force. According to Coulomb's law of electrostatic force the electrostatic force between two charged objects is directly proportional to the product of their charges and inversely proportional to the square of the distance between them [40]. The electrostatic forces are two types: attraction and repulsion. The force between unlike charge objects is attraction and between two like charge is repulsion.

By Coulomb's law the magnitude of the electrostatic force between two objects of charges Q_i and Q_j [40] is:

$$F_{ij} = K \frac{Q_i Q_j}{R^2} \quad (1)$$

where F_{ij} is the magnitude of electrostatic force, K is the Coulomb's constant, Q_i and Q_j are charges of i th and j th objects respectively and R is the distance between two charges Q_i and Q_j . Then electric field around charge Q_i is [40]

$$E_i = \frac{F_{ij}}{Q_i} \quad (2)$$

According to Newton's second law of motion, when force F_{ij} is applied to an object of mass M , its acceleration is [40]:

$$a_i = \frac{F_{ij}}{M} \quad (3)$$

or

$$a_i = \frac{E_i Q_i}{M} \quad (4)$$

Eq. (1) shows that the electrostatic force is greater for highly charged and closed objects. Taking idea from the above mentioned physical laws, in the next section the formulation of AEFA algorithm is presented.

3. Artificial electric field algorithm (AEFA)

Artificial electric field algorithm is inspired by the Coulomb's law of electrostatic force which states that an electrostatic (attraction or repulsion) force between two charge particle is directly proportional to the product of their charges and inversely proportional to the square of the distance between their positions. In the proposed algorithm agents are considered as charge particle and their strength is measured by their charges. All these particles can attract or repel each others by an electrostatic force and due to this force objects move in the search space. Hence charges are used as the direct form of communication through electrostatic force and position of the charge correspond to a solution of the problem. The charges are defined as a function of the fitness of the candidate solution and the fitness of the population. In proposed algorithm we only consider the attraction electrostatic force in such a way that a charge particle with greatest charge ("best" individual) attract all other particles of lower charge and move slowly in the search space. So AEFA could consider as an isolated system of charges which obeying Coulomb's law of electrostatic force and the law of motion which are given follows:

Coulomb's First Law of Electrostatic: It states that like charge particles repel each other and unlike charge particles attract each other.

Coulomb's Second Law of Electrostatic: It state that the force of attraction between unlike or the force of repulsion between like charge particle is directly proportional to the product of their charges and inversely proportional to the square of the distance between centers of the charges.

Law of Motion: The current velocity of any charge is equal to the sum of the fractions of its previous velocity and the variations in the velocity. Variation in the velocity or acceleration of any charge is equal to the force exerted upon the system divided by the mass of the particle.

Now based on the above law, we define the Physics of the AEFA algorithm. Let the position of i th particle in the d -dimensional search space be $X_i = (x_i^1, x_i^2, \dots, x_i^d)$ for $i = 1, 2, 3, \dots, N$, where x_i^d is the position of i th particle in the d th dimension. In the proposed AEFA we use the position of the global best fitness obtained by all the charged particles as well as the personal best fitness history of each particle. The position of the best fitness value obtained by any particle i at any time t is given by the following Eq.

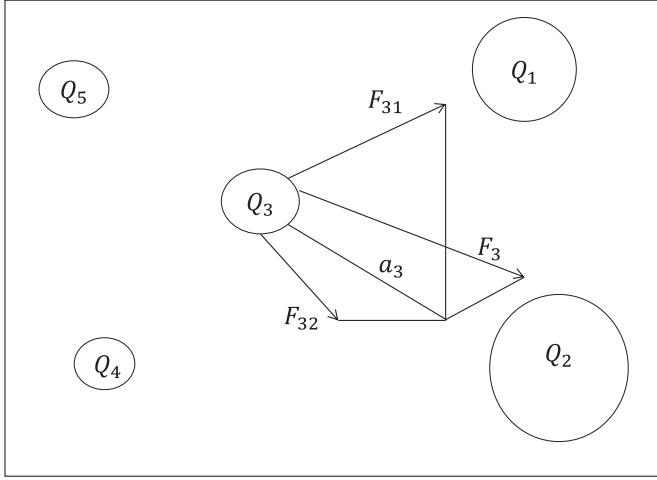


Fig. 1. Resultant Electric force acting on a charged particle.

$$p_i^d(t+1) = \begin{cases} p_i^d(t) & \text{if } f(P_i(t)) < f(X_i(t+1)) \\ x_i^d(t+1) & \text{if } f(X_i(t+1)) \leq f(P_i(t)) \end{cases} \quad (5)$$

The best fitness over all particles is denoted by $P_{best} = X_{best}$.

Now, at any time t the force acting on the charge i from charge j is defined as follows:

$$F_{ij}^d(t) = K(t) \frac{Q_i(t) * Q_j(t)(p_j^d(t) - X_i^d(t))}{R_{ij}(t) + \epsilon} \quad (6)$$

where $Q_i(t)$ and $Q_j(t)$ are the charges of i th and j th particle at any time t , $K(t)$ is the Coulomb's constant at any time t , ϵ is a small positive constant and $R_{ij}(t)$ is the Euclidian distance between two particles i and j given by:

$$R_{ij}(t) = \|X_i(t), X_j(t)\|_2 \quad (7)$$

The Coulomb's constant $K(t)$ is taken as a function of iteration and the max. iteration, it can be calculated by the following equation:

$$K(t) = K_0 * \exp(-\alpha \frac{\text{iter}}{\text{maxiter}}) \quad (8)$$

where α and K_0 are parameter and initial value respectively. iter is the current iteration and maxiter is the predefined maximum number of iterations. The Coulomb's constant is initialized at the beginning of the algorithm to a high value for the exploration of the algorithm and then

iteration by iteration decreasing to control the search accuracy of the algorithm, as illustrated in Fig. 2.

The total electric force acts on the i th particle by all the other particles at any time t in a d -dimensional search space is given by the following equation:

$$F_i^d(t) = \sum_{j=1, j \neq i}^N \text{rand}() F_{ij}^d(t) \quad (9)$$

where $\text{rand}()$ is a uniform random number in the interval $[0, 1]$, which is used to provide the stochastic nature to the algorithm. N is the number of particles in the search space and F_i is the resultant force acting on the i th charge particle, as illustrated in Fig. 1.

The electric field of the i th particle at any time t and in d th dimension is given by the following equation:

$$E_i^d(t) = \frac{F_i^d(t)}{Q_i(t)} \quad (10)$$

Hence by the Newton's second law of motion i.e Eq. (3) and by Eq. (10) the acceleration of the i th particle at any time t and d th dimension is given by the equation:

$$a_i^d(t) = \frac{Q_i(t) E_i^d(t)}{M_i(t)} \quad (11)$$

where $M_i(t)$ is the unit mass of i th particle at any time t . The velocity and position of the particle are updated as follows:

$$V_i^d(t+1) = \text{rand}() * V_i^d(t) + a_i^d(t) \quad (12)$$

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1) \quad (13)$$

where $\text{rand}()$ is a uniform random number in the interval $[0, 1]$.

The charge of the particles are calculated by the fitness functions and assuming that charge of each particle is equal. Eq. (15) is one of the example of a suitable charge function. The charge function may be of different forms meeting the requirements and some will be better than others for specific classes of optimization problems. The basic requirement to choose a charge function is that, the charge of best particle should be largest value (normalized) i.e. $Q_{best} = 1$ while all other particle of worse fitness have a smaller value lying in $[0, 1]$. The choice of charge does depends on the goal maximization or minimization. For maximization or minimization problem it should be monotonically increasing or monotonically decreasing respectively, which guaranteeing the largest charge and hence greater force for the best fitness value.

$$Q_i(t) = Q_j(t) \quad i, j = 1, 2, \dots, N \quad (14)$$

$$q_i(t) = \exp\left(\frac{\text{fit}_{p_i}(t) - \text{worst}(t)}{\text{best}(t) - \text{worst}(t)}\right) \quad (15)$$

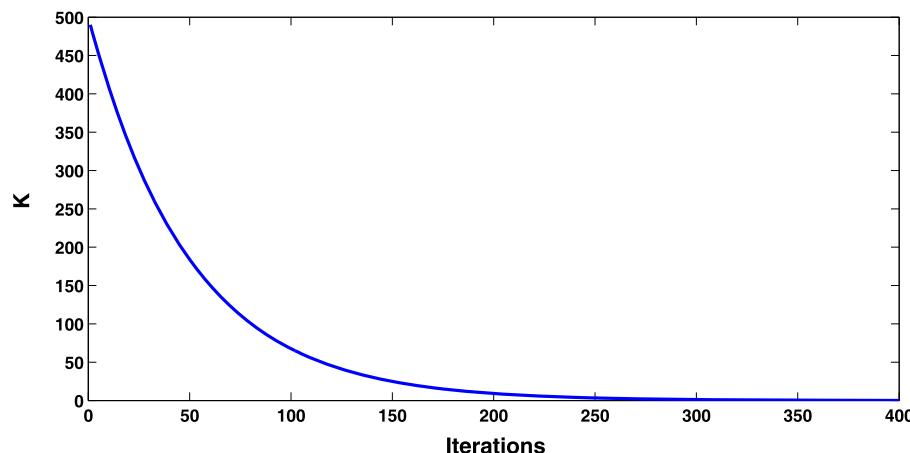


Fig. 2. Decreasing Coulomb's constant.

$$Q_i(t) = \frac{q_i(t)}{\sum_{i=1}^N q_i(t)} \quad (16)$$

where fit_i is the fitness value of i th particle at any time t . $best(t)$ and $worst(t)$ are defined as follows (for maximization problem):

$$best(t) = \max(fit_j(t), j \in (1, 2, \dots, N)) \quad (17)$$

$$worst(t) = \min(fit_j(t), j \in (1, 2, \dots, N)) \quad (18)$$

For minimization problem the above equations changed to the following equations respectively:

$$best(t) = \min(fit_j(t), j \in (1, 2, \dots, N)) \quad (19)$$

$$worst(t) = \max(fit_j(t), j \in (1, 2, \dots, N)) \quad (20)$$

The procedure of AEFA is explained in [Algorithm 1](#) and the flow chart of AEFA is depicted in [Fig. 3](#).

Algorithm 1 Pseudo code of Artificial Electric Field Algorithm (AEFA).

Initialization

Randomly initialize $(X_1(t), X_2(t), \dots, X_N(t))$ of population size N in the search range $[X_{\min}, X_{\max}]$

Initialize the velocity to a random value

Evaluate the fitness values $(fit_1(t), fit_2(t), \dots, fit_N(t))$ of agent X

Set iteration $t = 0$

Reproduction and Updating

while Stopping Criterion is not satisfied do

Calculate $K(t)$, $best(t)$ and $worst(t)$

for $i=1: N$ do

 Evaluate the fitness values $fit_i(t)$

 Calculate the total force in each direction $F_i(t)$

 Calculate the acceleration $a_i(t)$

$V_i(t+1) = rand() \times V_i(t) + a_i(t)$

$X_i(t+1) = X_i(t) + V_i(t+1)$

end for

end while

3.1. Learning strategy in AEFA

For success of any metaheuristic optimization technique the learning factor is a very important component. This learning component improved and accelerate the performance of that technique. In our algorithm the force given by Eq. (6) play the role of learning component. In Eq. (6) the term $p_j^d(t)$ used the history of personal best fitness of any particle j , which is given by Eq. (5) and the terms Q_i and Q_j given by Eq. (14) and Eq. (15) used the global best fitness history. Due to these two terms, each particle in the search space is attracted toward the best particle and the term force is responsible for the movement of each particle toward the best particle and improved the performance of AEFA.

4. Comparative study of AEFA with CSS, MOA, PSO and GSA

The nature of the proposed AEFA seems similar to CSS [12], MOA [13], PSO [21] and GSA [11]. But in reality AEFA is highly different in nature from CSS, MOA, PSO and GSA. In order to establish the novelty of the AEFA a detailed comparison of the AEFA with CSS, MOA, PSO and GSA is demonstrated in the following section along with highlights of their fundamental differences.

4.1. CSS algorithm

A novel heuristic optimization method: charged system search (CSS) [12] is a multi-dimensional search algorithm inspired by the Coulomb's law of electrostatic force and Newtonian law of motion. The agents are

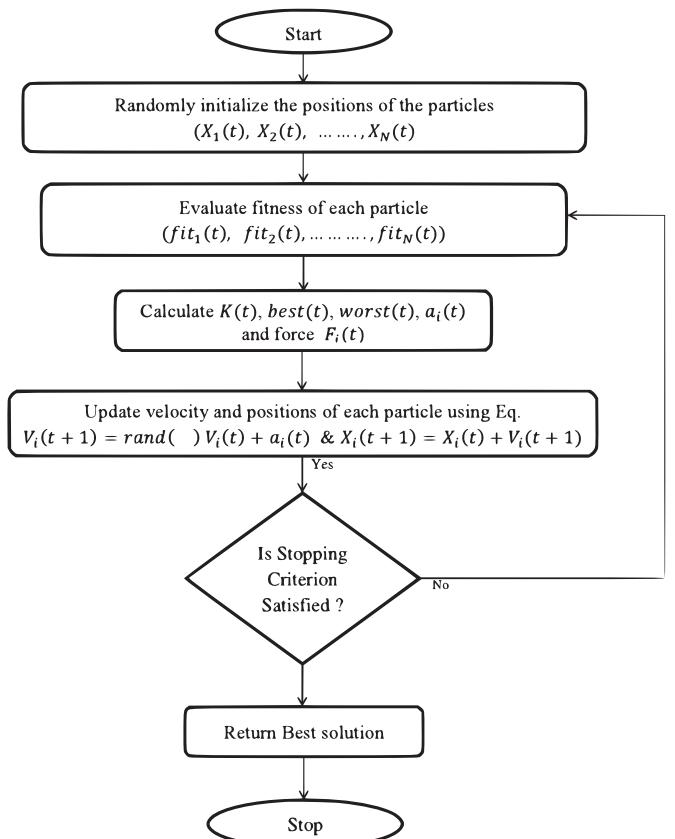


Fig. 3. Flow chart of AEFA.

considered as a charged sphere and moves in the search space by an attraction force. Charge of the agents are given by the following fitness function:

$$q_i = \frac{fit(i) - worst}{best - worst}, i = 1, 2, \dots, N \quad (21)$$

Where fit_i is the fitness value of the i th particle, $best$ and $worst$ are the so far best and worst fitness values of all particles. The resultant force acting on a charged particle is given by the following Eq.

$$F_i = q_i \sum_{i,j} \left(\frac{q_i r_{ij} i_j}{a^3} + \frac{q_i l_2}{r_{ij}^2} \right) p_{ij} (X_i - X_j) \quad (22)$$

$$\{j = 1, 2, \dots, N, i_1 = 1, i_2 = 0 \Leftrightarrow r_{ij} < a \text{ and } i_1 = 0, i_2 = 1 \Leftrightarrow r_{ij} \geq a\}$$

where a is the radius of the sphere, N is the number of the particles, q_i and q_j are charges of i th and j th agents respectively and r_{ij} is the separation distance between two charged particles. The separation distance is given by the following Eq.

$$r_{ij} = \frac{\|X_i - X_j\|}{\left\| \frac{(X_i + X_j)}{2} - X_{best} \right\| + \epsilon} \quad (23)$$

The velocity and position update equations of each agents are given as follows:

$$X_{j,new} = rand() \frac{F_j}{m_j} \Delta t^2 + rand() k_a V_{j,old} \Delta t + X_{j,old} \quad (24)$$

$$V_{j,new} = \frac{X_{j,new} - X_{j,old}}{\Delta t} \quad (25)$$

where $rand()$ is a random number uniformly distributed in $(0, 1)$, m_j is mass of the j th agent which is equal to q_j , k_a is an acceleration coefficient, k_v is the velocity coefficient and Δt is the time step. The accel-

Table 1
Difference between AEFA and CSS.

Sr. no.	AEFA	CSS [12]
1	Particles are considered as point charges	Particles are considered as charged spheres.
2	Includes Coulomb's constant $K(t)$ for determination of force and acceleration	Does not include the Coulomb's constant $K(t)$ for determination of force.
3	The Velocity update Eq. does not include any new parameter	The velocity update Eq. includes two new parameters acceleration and velocity coefficient k_a and k_v respectively.
4	Less number of parameter	Parameters are more than AEFA.
5	Formula for charge is $q_i = \exp\left(\frac{fit_{p_i} - worst(t)}{best(t) - worst(t)}\right)$	Formula for charge is: $q_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)}$
6	It provide a detailed theoretical analysis for the convergence of particle trajectory and stability of AEFA	No theoretical work had been done for the convergence.

eration and velocity coefficients are determined by the following equations:

$$k_a = \frac{(1 + \frac{itr}{maxitr})}{2}, k_v = \frac{(1 - \frac{itr}{maxitr})}{2} \quad (26)$$

4.1.1. Differences in AEFA and CSS

Even though both the algorithm AEFA and CSS are inspired from the Coulomb's law of electrostatic force and Newtonian law of motion but both have different formulations. The main differences are presented in Table 1.

4.2. MOA algorithm

The magnetic field optimization algorithm works on the principles of magnetic field theory. In MOA the agents are the magnetic particles which operate in a lattice like interaction environment. The position of the particle are given by $X_{ij}^k(t) = R(lb^k, ub^k)$ for $i, j = 1, 2, \dots, S$, $k = 1, 2, \dots, d$ where i, j is the location of the particle in the lattice, t is the number of iteration, S is the size of lattice, d is the dimension of the search space, R is a uniform random number, and lb and ub are lower and upper bound of the k th dimension of search space. The objective function value of each particle is stored in the magnetic field and the magnetic field of each particle is given by:

$$B_{ij} = \frac{B_{ij} - best}{worst - best} \quad (27)$$

where $best = \min_{i,j=1}^S \{B_{ij}(t)\}$ and $worst = \max_{i,j=1}^S \{B_{ij}(t)\}$. Now the force applied by a particle to its nearest neighbor's is calculated as follows:

$$F_{ij}^d = \frac{(x_{uv}^k(t) - x_{ij}^k(t)) \times B_{uv}(t)}{D(x_{ij}^k(t), x_{uv}^k(t))} \quad (28)$$

where D is the distance between two particles given by:

$$D_1(x_{ij}(t), x_{uv}(t)) = \frac{1}{d} \sum_{k=1}^d \left| \frac{x_{ij}^k(t) - x_{uv}^k(t)}{ub^k - lb^k} \right| \quad (29)$$

Each particle update its velocity and position by the following equations:

$$V_{ij}^k(t+1) = \frac{F_{ij}^k}{M_{ij}^k} \times R(lb^k, ub^k) \quad (30)$$

$$X_{ij}^k(t+1) = X_{ij}^k(t) + V_{ij}^k(t+1) \quad (31)$$

where M_{ij} is the mass of the particle and given by:

$$M_{ij}(t) = \alpha + \rho \times B_{ij}(t) \quad (32)$$

where α and ρ are parameters.

4.2.1. Difference between AEFA and MOA

Even though both the algorithm AEFA and MOA used the concept of magnetic field theory but they have large difference in the search strategy, working principles and formulations. The main difference are depicted in the Table 2.

4.3. Comparative study of AEFA and PSO

Even though AEFA and PSO are both population based optimization algorithm and the optimization obtained by the particles movement in the search space, however both have different movement strategy. The main differences are listed as follows:

1. The searching idea of AEFA and PSO are different. AEFA works on Coulomb's law of electrostatic force and PSO mimics the social behavior of bird flocking.
2. In AEFA the direction of the particle is calculated based on the overall force exerted by all the other particles but in PSO the particle direction is calculated only by its personal and the global best positions.
3. In AEFA the updating is performed by the force and the force is directly proportional to the objective function value so the particle

Table 2
Difference between AEFA and MOA.

Sr. no.	AEFA	MOA
1	Agents are considered as point charge particles	Agents are considered as magnetic particles.
2	Directly inspired from Coulomb's law of electrostatic force and include the Coulomb's constant which control the accuracy of the search space	It use only the concept of attraction of magnetic field theory
3	AEFA works on attraction-repulsion principle and only the particle of higher charge attract the particle of lower charge and create an electric field around the particle	It works only attraction principle and the particle attract its neighbor's particles
4	AEFA consider unit mass for each particle	MOA calculate the mass of each particle by $M_{ij}(t) = \alpha + \rho \times B_{ij}(t)$
5	Formula for charge is $q_i = \exp\left(\frac{fit_{p_i} - worst(t)}{best(t) - worst(t)}\right)$	Formula for magnetic field is: $B_{ij} = \frac{B_{ij} - best}{worst - best}$
6	It is a memory based algorithm and used the particles personal best and global best fitness history to update the positions and velocity	It is memory less and only used the global best of the algorithm.
7	It provide a detailed analysis and theoretical convergence of particle trajectory and stability of AEFA	No theoretical work had been done for the convergence of the algorithm.

Table 3
Difference between AEFA and GSA.

Sr. no.	AEFA	GSA [11]
1	Inspired by Coulomb's law of electrostatic force	Inspired by Newton's law of gravity.
2	Charges are considered as particles	Masses are considered as agents
3	Particles move in the search space by electrostatic force	Agents move in the search space by gravitational force
4	The force and acceleration expressions includes the Coulomb's constant and charges	The force and acceleration expressions include the gravitational constant, active and passive masses
5	Formula for charge is $q_i = \exp\left(\frac{fit_p_i - worst(t)}{best(t) - worst(t)}\right)$ and $Q_i(t) = Q_j(t) = \frac{q_i(t)}{\sum_{i=1}^N q_i(t)}$	Formula for active, passive and inertial masses is: $m_i(t) = \frac{m_i(t) - worst(t)}{best(t) - worst(t)}$ and $M_{ai} = M_{pi} = M_{ii} = M_i(t) = \frac{m_i(t)}{\sum_{i=1}^N m_i(t)}$
6	It is a memory based algorithm and used the particles personal best and global best fitness history to update the positions and velocity	It is memory less and only used the global best of the algorithm.
7	Provides theoretical analysis for the convergence of the algorithm	No theoretical work had been done for the convergence of the algorithm.

can see the search space around themselves in the influence of electrostatic force while in PSO the particles are updated without considering the objective function value (i.e. the quality of solutions).

4. In AEFA, updating is performed by considering the distances between the solution while PSO updating without considering the distances between the solutions.
5. In AEFA the parameters are less than PSO.

4.4. Comparative study of AEFA and GSA

In this section, we compare the proposed algorithm AEFA with GSA [11] by theoretically. The theoretical comparison between AEFA and GSA is given in Table 3.

5. A detailed analysis of AEFA

This sections presents the rigorous analysis of the theoretical convergence, behavior of acceleration update equation and electric field on exploration and exploitation ability over some Learning-based Real-Parameter Single Objective Optimization problems proposed in IEEE CEC 2015 [41].

5.1. Convergence of AEFA

In this section we presents a theoretical analysis for ensuring the convergence of proposed algorithm AEFA. First we prove the conditions of stability for the convergence of charged particles in AEFA. Let $X_i(t)$ be the position of any particle i in the search space at any time t . The velocity and position update Eq. of AEFA are as follows:

$$V_i^d(t+1) = cV_i^d(t) + \frac{\sum_{j=1, j \neq i}^N rand_j F_{ij}^d(t)}{Q_i(t)} \quad (33)$$

$$V_i^d(t+1) = cV_i^d(t) + \frac{\sum_{j=1, j \neq i}^N K(t)rand_j Q_j(t)(p_j^d(t) - X_i^d(t))}{R_{ij} + \epsilon} \quad (34)$$

$$X_i^d(t+1) = X_i^d(t) + V_i^d(t+1) \quad (35)$$

To make the analysis easy we reduced the d-dimensional and N-particles system to a simplified one-dimensional and one-particle system, which can be easily extended to d-dimensional and N-particle system. Now the velocity and position update Eq. becomes:

$$V_i(t+1) = cV_i(t) + \frac{\sum_{j=1, j \neq i}^N rand_j F_{ij}(t)}{Q_i(t)} \quad (36)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (37)$$

Considering a set A of N particles. Dividing A into two subsets G_i and H_i which are defined as follows:

Definition 1. $G_i = \{j \mid fitP_j < fitX_i \forall i \in A\}$ is a set of all particles whose fitness is better than particles i .

Definition 2. $H_i = \{j \mid fitP_j > fitX_i \forall i \in A\}$ is a set of all particles whose fitness is worse than particles i .

Using the above two definitions the velocity update Eq. becomes:

$$V_i(t+1) = cV_i(t) - \frac{\sum_{j \in G_i} rand_j K(t)Q_j(t)(X_i(t) - P_j)}{R_{ij} + \epsilon} - \frac{\sum_{j \in H_i} rand_j K(t)Q_j(t)(X_i(t) - P_j)}{R_{ij} + \epsilon} \quad (38)$$

where P_j is a constant.

$$V_i(t+1) = cV_i(t) - \left(\frac{\sum_{j \in G_i} rand_j K(t)Q_j}{R_{ij} + \epsilon} + \frac{\sum_{j \in H_i} rand_j K(t)Q_j}{R_{ij} + \epsilon} \right) X_i(t) + \frac{\sum_{j \in G_i} rand_j K(t)Q_j P_j}{R_{ij} + \epsilon} + \frac{\sum_{j \in H_i} rand_j K(t)Q_j P_j}{R_{ij} + \epsilon} \quad (39)$$

For simplification we use following definitions:

$$F_{G_i} = \frac{\sum_{j \in G_i} rand_j K(t)Q_j}{R_{ij} + \epsilon} \quad (40)$$

$$F_{H_i} = \frac{\sum_{j \in H_i} rand_j K(t)Q_j}{R_{ij} + \epsilon} \quad (41)$$

$$F_{GH_i} = F_{G_i} + F_{H_i} \quad (42)$$

$$F_i = \frac{\sum_{j \in G_i} rand_j K(t)Q_j P_j}{R_{ij} + \epsilon} + \frac{\sum_{j \in H_i} rand_j K(t)Q_j P_j}{R_{ij} + \epsilon} \quad (43)$$

With these simplifications, the velocity update Eq. (39) can be written as

$$V_i(t+1) = cV_i(t) - F_{GH_i} + F_i \quad (44)$$

Substituting Eq. (44) into position update Eq. (37) we get the following non-homogeneous recurrence relation

$$X_i(t+1) - (1 + c - F_{GH_i})X_i(t) + cX_i(t-1) = F_i \quad (45)$$

The above Eq. (45) is a second order non-homogeneous difference equation, where F_i can be taken as an external input, which does not affect the stability of the particles.

In order to test the convergence of AEFA first we discuss the methodology of convergence of stochastic sequences. Since AEFA has stochastic nature and each particle's position has random component, so we consider a stochastic sequence $X_i(t)$ of particle's position. There are many ways to define the convergence of a stochastic sequence,

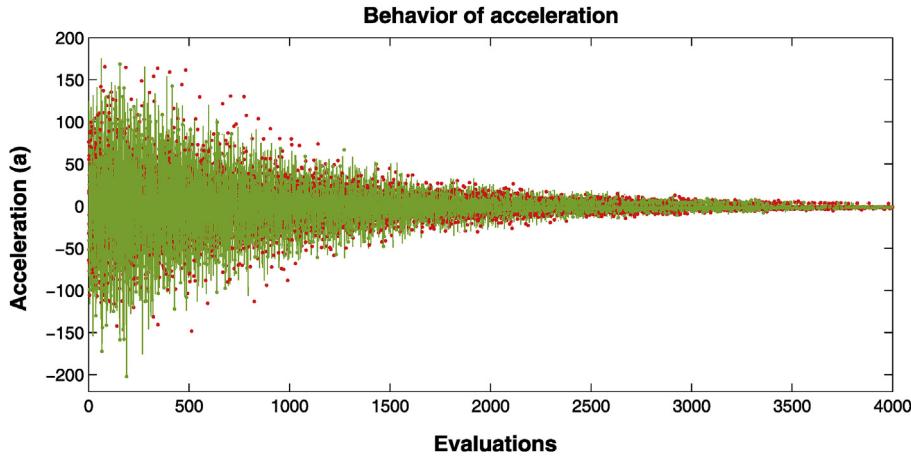


Fig. 4. Behavior of acceleration update Eq. of AEFA.

here we use the mean convergence notion according to which a sequence $X_i(t)$ converges to some position vector X^* if and only if $\lim_{t \rightarrow \infty} E[X_i(t)] = X^*, \forall i = 1, 2, 3, \dots, N$ exists, where $E[X_i(t)]$ is the expected value of position $X_i(t)$. So in terms of this definition AEFA is converges to a solution vector X^* in the search space.

Now to use the above mentioned definition of convergence to AEFA, applying the mathematical expectation operator to Eq. (45) it becomes

$$E[X_i(t+1)] - (1 + c - E(F_{GH_i}))E[X_i(t)] + cE[X_i(t-1)] = E(F_i) \quad (46)$$

Let,

$$\phi_i = E(F_{GH_i}) = \frac{1}{2}K\left(\frac{\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j}{R_{ij} + \epsilon}\right) \quad (47)$$

$$\psi_i = E(F_i) = \frac{1}{2}K\left(\frac{\sum_{j \in G_i} Q_j P_j + \sum_{j \in H_i} Q_j P_j}{R_{ij} + \epsilon}\right) \quad (48)$$

$$K = E[K(t)] = E[K_0 \exp(-\frac{\alpha * t}{t_{max}})] = \frac{K_0 t_{max}^2}{\alpha^2} \quad (49)$$

Making these substitutions in Eq. (46) we get,

$$E[X_i(t+1)] - (1 + c - \phi_i)E[X_i(t)] + cE[X_i(t-1)] = \psi_i \quad (50)$$

The characteristic Eq. of the above second order linear homogeneous system is

$$\lambda^2 - (1 + c - \phi_i)\lambda + c = 0 \quad (51)$$

Now based on the theory of stability for discrete dynamics system [42], the convergence of AEFA is obtained from the stability of second order liner non-homogeneous system Eq. (45). The second order system is stable if and only if its eigenvalues lies inside the unit circle in the complex plane i.e. the absolute value of the eigenvalues are less than 1. Thus we obtained the following result:

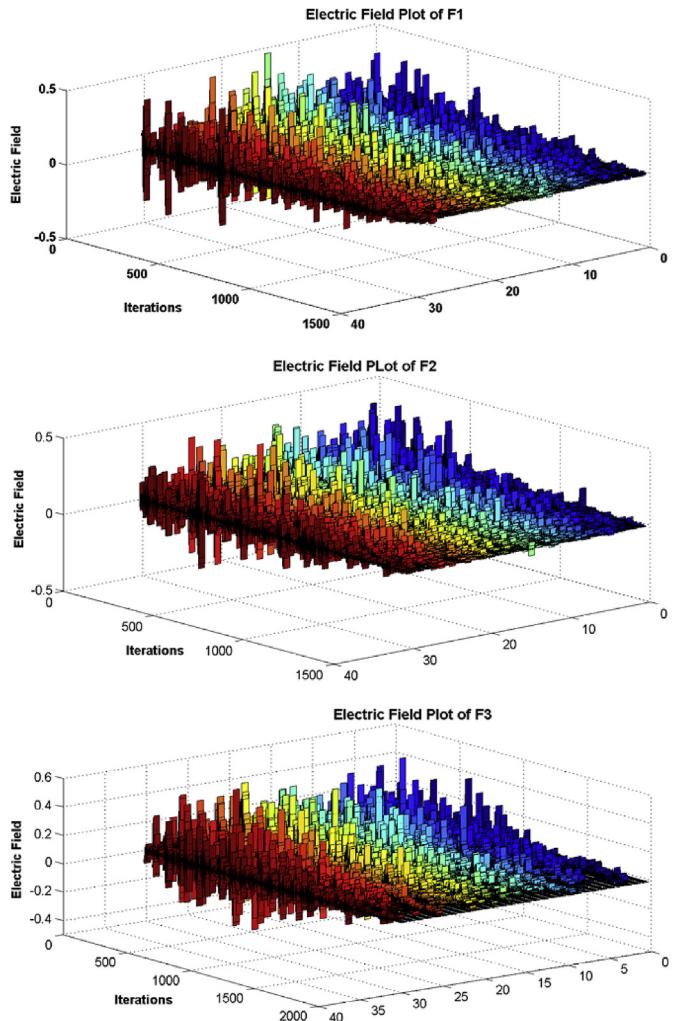
Theorem 1. The sequence of particles $E[X_i(t)]$ converges to X_{best} (position vector of the particle with best fitness value) if and only if $0 \leq c < 1$ and $0 < \phi_i < 2(1 + c)$ for a given $c \geq 0$ and $K > 0$.

Proof. The eigenvalues of $\lambda^2 - (1 + c - \phi_i)\lambda + c = 0$ is given by

$$\lambda_i = \frac{1 + c - \phi_i \pm \sqrt{(1 + c - \phi_i)^2 - 4c}}{2} \quad (52)$$

The condition of convergence for the sequence $E[X_i(t)]$ is that the absolute values of both the eigenvalues λ_1 and λ_2 are less than 1 i.e.

$$\lambda_i = \left| \frac{1 + c - \phi_i \pm \sqrt{(1 + c - \phi_i)^2 - 4c}}{2} \right| < 1 \quad (53)$$

Fig. 5. Electric Field Plots of problems f_1 to f_3 .

Since the discriminant $(1 + c - \phi_i)^2 - 4c$ has three different values so, there arises three cases which are given follows:

Case (1). $(1 + c - \phi_i)^2 - 4c = 0$ or $(1 + c - \phi_i)^2 = 4c$ i.e. the discriminant is zero.

In this case, the eigenvalues λ_1, λ_2 are real and equal to each other and given by $\lambda_1 = \lambda_2 = \frac{1+c-\phi_i}{2}$. This Condition $(1 + c - \phi_i)^2 = 4c$

Table 4

IEEE CEC15 Learning Based Benchmark Test Suite [41], with Search Range = $[-100, 100]^D$ and f_{\min} is minimum fitness value.

Nature of Functions	Function No.	Function Name	f_{\min}
Unimodal Functions	f_1	Rotated High Conditioned Elliptic Function	100
	f_2	Rotated Cigar Function	200
Simple Multimodal Functions	f_3	Shifted and Rotated Ackley's Function	300
	f_4	Shifted and Rotated Rastrigin's Function	400
	f_5	Shifted and Rotated Schwefel's Function	500
Hybrid Functions	f_6	Hybrid Function 1 (N = 3)	600
	f_7	Hybrid Function 2 (N = 4)	700
	f_8	Hybrid Function 3 (N = 5)	800
Composition Functions	f_9	Composition Function 1 (N = 3)	900
	f_{10}	Composition Function 2 (N = 3)	1000
	f_{11}	Composition Function 3 (N = 5)	1100
	f_{12}	Composition Function 4 (N = 5)	1200
	f_{13}	Composition Function 5 (N = 5)	1300
	f_{14}	Composition Function 6 (N = 7)	1400
	f_{15}	Composition Function 7 (N = 10)	1500

Table 5

Fine tuned parameter setting for AEFA.

K_0	α	pop size (N)	Dimension (D)	MaxFES
500	30	20	30	10000*D.

is equivalent to $\phi_i = 1 + c \pm 2\sqrt{c}$ i.e. $\lambda_1 = \lambda_2 = \pm\sqrt{c}$ and $|\lambda_1| = |\lambda_2| = \sqrt{c}$. Hence $|\lambda_1| = |\lambda_2| < 1$ for $0 \leq c < 1$.

Case (2). $(1 + c - \phi_i)^2 - 4c > 0$ or $(1 + c - \phi_i)^2 > 4c$ i.e. the discriminant is positive.

In this case, the eigenvalues λ_1, λ_2 are real and distinct which are given by

$$\lambda_1 = \frac{1 + c - \phi_i + \sqrt{(1 + c - \phi_i)^2 - 4c}}{2} \quad (54)$$

and

$$\lambda_2 = \frac{1 + c - \phi_i - \sqrt{(1 + c - \phi_i)^2 - 4c}}{2} \quad (55)$$

This condition is equivalent to $\phi_i > 1 + c + 2\sqrt{c}$ or $\phi_i < 1 + c - 2\sqrt{c}$.

Case (a). If $\phi_i > 1 + c + 2\sqrt{c}$, then $\max\{|\lambda_1, \lambda_2|\} < 1$ is assured when $\lambda_2 > -1$, thus resulting in $c < 1$ and $1 + c - 2\sqrt{c} < \phi_i < 2 + 2c$.

Case (b). If $\phi_i < 1 + c - 2\sqrt{c}$, then $\max\{|\lambda_1, \lambda_2|\} < 1$ is assured when $\lambda_1 < 1$, thus resulting in $c < 1$ and $0 < \phi_i < 1 + c - 2\sqrt{c}$.

Combining case (a) and (b) we get, for stability $c < 1$ and $1 + c - 2\sqrt{c} < \phi_i < 2 + 2c$.

Case 3. $(1 + c - \phi_i)^2 - 4c < 0$ or $(1 + c - \phi_i)^2 < 4c$ i.e. the discriminant is negative.

In this case both the eigenvalues are complex conjugate of each other and having equal modulus and given follows:

$$\lambda_1 = \frac{1 + c - \phi_i + i\sqrt{(1 + c - \phi_i)^2 - 4c}}{2} \quad (56)$$

and

$$\lambda_1 = \frac{1 + c - \phi_i - i\sqrt{(1 + c - \phi_i)^2 - 4c}}{2} \quad (57)$$

$$|\lambda_1| = |\lambda_2| = \sqrt{\frac{(1+c-\phi_i)^2+(1+c-\phi_i)^2-4c}{4}} = \sqrt{\frac{2(1+c-\phi_i)^2-4c}{2}}$$

The stability condition for convergence $|\lambda_1| < 1$ and $|\lambda_2| < 1$ is satisfied when $\sqrt{\frac{2(1+c-\phi_i)^2-4c}{2}} < 1$ which results in $\phi_i > 1 + c - \sqrt{2 + 2c}$. The stability condition of convergence for this case is equivalent to $1 + c - 2\sqrt{c} < \phi_i < 1 + c + 2\sqrt{c}$.

Case (1)–(3) conclude that the stability of particles achieved only when stability conditions imposed by cases (1)–(3) are met simultaneously. Hence the sequence of particles $E[X_i(t)]$ is convergent if and only if the following conditions are satisfied:

$$0 \leq c < 1 \text{ and } 0 < \phi_i < 2 + 2c \quad (58)$$

When the sequence $E[X_i(t)]$ is convergent to some position vector X^* , which can be computed from the Eq. $X^* - (1 + c - \phi_i)X^* + cX^* = \psi_i$, which implies

$$\phi_i X^* = \psi_i \quad (59)$$

Now, substituting the values of ϕ_i and ψ_i in Eq. (59) we get

$$\frac{1}{2}K\left(\frac{\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j}{R_{ij} + \epsilon}\right)X^* = \frac{1}{2}K\left(\frac{\sum_{j \in G_i} Q_j P_j + \sum_{j \in H_i} Q_j P_j}{R_{ij} + \epsilon}\right) \quad (60)$$

It can be written as: $(\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j)X^* = (\sum_{j \in G_i} Q_j P_j + \sum_{j \in H_i} Q_j P_j)$

Further it can be written as: $(\sum_{j \in G_i, j \neq \text{best}} Q_j + \sum_{j \in H_i} Q_j)X^* + Q_{\text{best}}X^* = (\sum_{j \in G_i, j \neq \text{best}} Q_j P_j + \sum_{j \in H_i} Q_j P_j) + Q_{\text{best}}P_{\text{best}}$

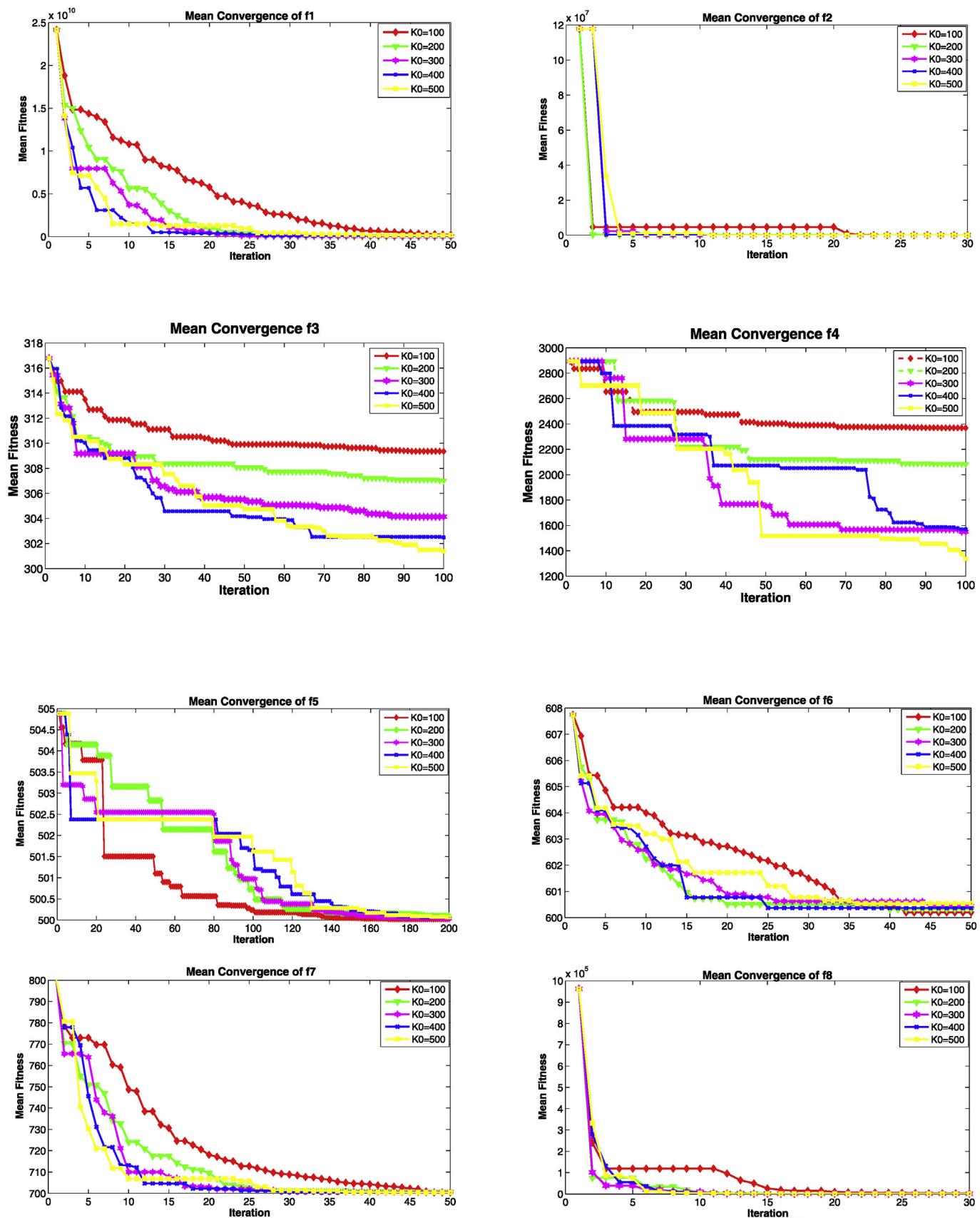
$$(\sum_{j \in G_i, j \neq \text{best}} Q_j X^* - \sum_{j \in G_i, j \neq \text{best}} Q_j P_j) + (\sum_{j \in H_i} Q_j X^* - \sum_{j \in H_i} Q_j P_j) + Q_{\text{best}}(X^* - P_{\text{best}}) = 0 \quad (61)$$

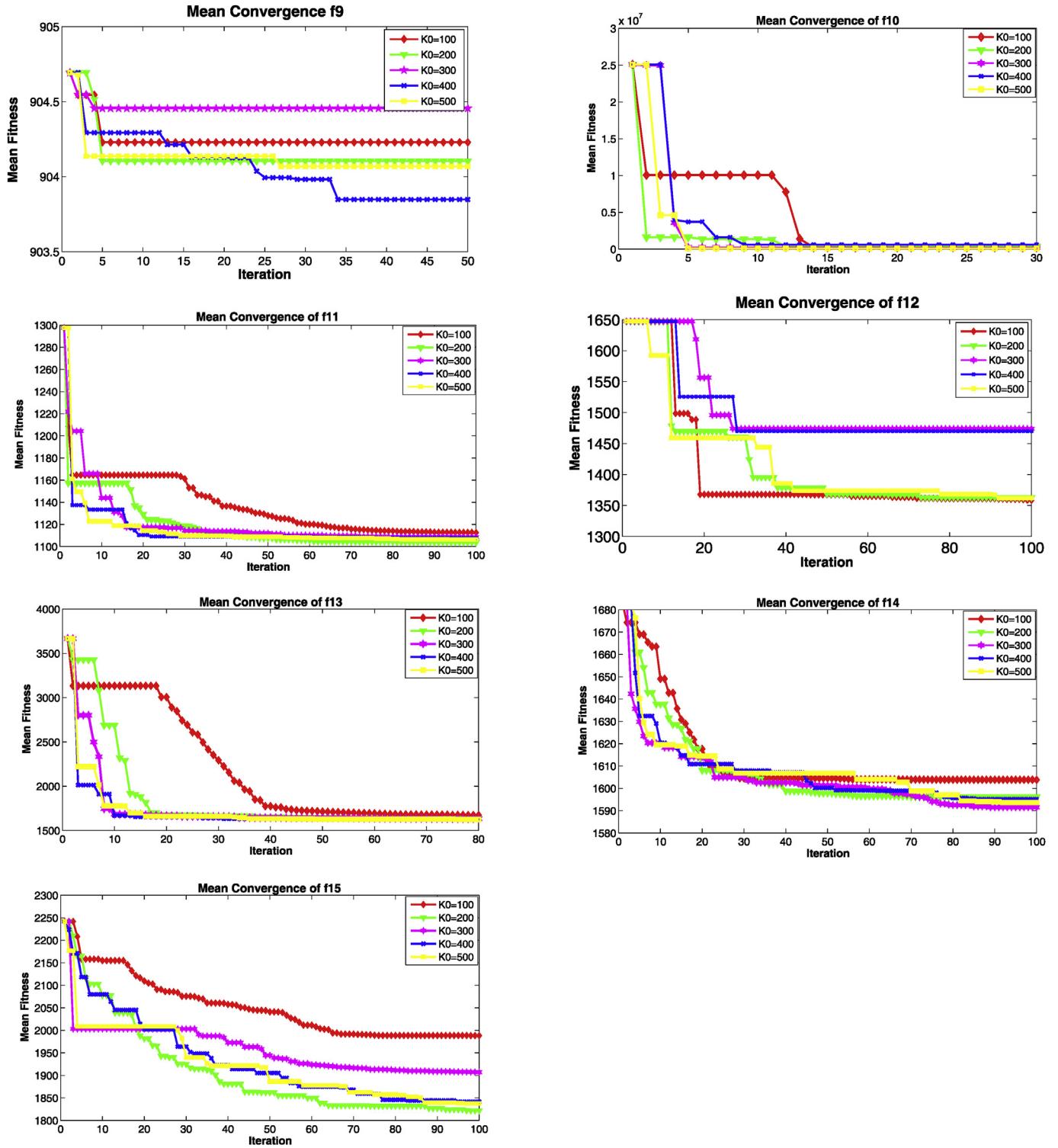
On observation we get above Eq. (61) ensured only if $X^* = P_j = P_{\text{best}}$. This completes the proof of **Theorem 1**.

By analysis of convergence for particles trajectory we find that $0 \leq c < 1$ and $0 < \phi_i < 2 + 2c$. Substituting the value of ϕ_i in Eq. (58), we get $0 < \frac{1}{2}K\left(\frac{\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j}{R_{ij} + \epsilon}\right) < 2 + 2c$. Thus for convergence of particles $0 < K < \frac{4(1+c)(R_{ij}+\epsilon)}{(\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j)}$ i.e. a small value of the Coulomb's constant $K(t)$ will guarantee to converge and the larger value of K i.e. $K > \frac{4(1+c)(R_{ij}+\epsilon)}{(\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j)}$ will result to the unstable state. At the initial stage of the algorithm the distance R_{ij} is large and the charge $(\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j)$ is small so $K(t)$ is large and may be not meet the convergence condition but it promotes the ability to enhance the search space to avoid the trapping in the local optima. On increase in iterations, the separation distance R_{ij} tends to reduce while $(\sum_{j \in G_i} Q_j + \sum_{j \in H_i} Q_j)$ tends to increase, since the charge particles move toward the best particle in the search space of AEFA due to the memory strategy of personal and global best fitness history of the particles. The Coulomb's constant $K(t)$ gradually decreases to a small value nearly zero and satisfy the convergent condition which is depicted in Fig. 2. Hence AEFA presents the capability of exploitation to obtain the optimal solutions and approach to a stable stage.

5.2. Behavior of acceleration update equation

In order to observe the effect of acceleration on exploration and exploitation ability of the proposed AEFA. The acceleration of the population is plotted against the function evaluations. From Fig. 4 it can

Fig. 6. Convergence behavior of AEFA for different values of K_0 .

Fig. 7. Convergence behavior of AEFA for different values of K_0 .

be observed that the intensity of the acceleration in the search region is high in the initial stage of the AEFA which promotes the enhanced exploration of the search region by the particles. On increase in the function evaluations the intensity of the acceleration decreases gradually and it promotes the better exploitation ability of AEFA.

5.3. Behavior of electric field

To check the role of electric field on exploration and exploitation ability of the proposed AEFA. The ribbon plot of impact of electric field values of the population is plotted against the iterations for three (f_1 to f_3) problems. From these figures it can be easily observed that the intensity of the electric field in the search region is high in the initial stage of the AEFA which promotes the enhanced exploration of the

Table 6

Comparative results of objective function values for CEC15 30D.

F	Metric	PSO	ABC	BBO	WWO	FWA_EI	ACO	SaDE	ICMLSP	dynFWA	SPS-L-SHADE-EIG	AEFA
f1	best	5.45E+10	1.21E+02	1.98E+06	2.57E+05	7.10E+05	9.90E+03	2.43E+03	1.00E+02	1.15E+04	1.00E+02	2.02E+02
	worst	7.26E+10	5.16E+10	1.03 + E07	2.94E+06	1.75E+06	6.98E+10	6.73E+04	1.00E+02	6.96E+04	1.00E+02	1.46E+03
	median	5.45E+10	3.58E+03	7.58E+06	1.28E+06	1.03E+06	3.88E+03	2.55E+04	1.00E+02	1.90E+04	1.00E+02	5.57E+02
	std	6.22E-04	1.71E+09	3.69E+06	7.14E+05	3.97E+05	7.69E+09	2.21E+04	0	1.06E+04	4.01E-13	3.13E+01
f2	best	4.16E+04	3.38E+04	2.77E+05	2.00E+02	2.00E+02	3.79E+04	2.00E+02	2.00E+02	2.00E+02	2.17E+02	2.05E+02
	worst	1.07E+07	6.82E+04	3.29E+06	2.89E+02	2.00E+02	1.44E+07	1.36E+05	2.00E+02	2.33E+02	2.00E+02	1.25E+04
	median	4.17E+04	3.38E+04	7.45E+05	2.08E+02	2.00E+02	4.20E+04	2.00E+02	2.00E+02	2.17E+02	2.00E+02	5.89E+03
	std	8.95E-08	7.28E-11	7.40E+05	2.40E+01	2.67E-03	1.05E+03	3.75E+04	0	9.11E-09	3.09E-11	2.70E+03
f3	best	3.42E+02	3.25E+02	3.20E+02	3.20E+02	3.27E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02
	worst	3.52E+02	3.46E+02	3.20E+02	3.20E+02	3.51E+02	3.21E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02
	median	3.42E+02	3.25E+02	3.20E+02	3.20E+02	3.27E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02	3.20E+02
	std	6.45E-12	1.12E+00	3.08E-02	6.46E-06	3.65E-05	1.66E+02	5.43E-02	0	3.06E-07	2.44E-01	1.44E-02
f4	best	4.38E+03	4.41E+03	4.31E+02	4.39E+02	5.64E+02	5.80E+03	4.15E+02	4.11E+02	4.09E+02	4.08E+02	4.07E+02
	worst	9.24E+02	8.26E+03	4.64E+02	5.30E+02	7.66E+02	9.79E+03	5.64E+02	4.11E+02	4.09E+02	4.13E+02	4.07E+02
	median	4.38E+03	4.63E+03	4.42E+02	4.93E+02	6.28E+02	6.22E+03	4.26E+02	4.11E+02	4.09E+02	4.08E+02	4.07E+02
	std	0	1.55E+02	1.07E+01	1.93E+01	4.55E+01	1.73E+02	2.91E+01	0	0	1.43E-01	1.37E-03
f5	best	5.00E+02	5.02E+02	1.78E+03	3.89E+03	2.63E+03	5.02E+02	4.01E+03	7.99E+02	6.27E+02	6.26E+02	6.25E+02
	worst	5.08E+02	5.06E+02	3.51E+03	5.63E+03	4.83E+03	5.08E+02	7.69E+03	7.99E+02	4.42E+02	6.39E+02	6.26E+02
	median	5.00E+02	5.02E+02	2.92E+03	3.08E+03	3.86E+03	5.02E+02	5.12E+03	7.99E+02	6.29E+02	6.26E+02	6.25E+02
	std	3.81E-12	1.06E-01	5.34E+02	3.97E+02	4.94E+02	1.81E-01	3.18E+02	0	5.08E-13	1.55E-01	3.68E-02
f6	best	6.05E+02	6.01E+02	5.36E+05	4.26E+03	1.83E+04	6.01E+02	8.96E+02	7.80E+02	8.08E+02	6.14E+02	8.49E+02
	worst	6.06E+02	6.05E+02	6.27E+06	1.84E+05	1.22E+05	6.06E+02	1.83E+04	7.80E+02	8.94E+02	6.65E+02	8.49E+02
	median	6.05E+02	6.01E+02	1.87E+06	6.74E+04	5.10E+04	6.01E+02	6.34E+03	7.80E+02	8.94E+02	6.47E+02	8.49E+02
	std	9.09E-13	2.26E-01	1.55E+06	4.41E+04	3.50E+04	5.38E-01	5.64E+03	0	6.70E-01	1.10E-01	3.43E-02
f7	best	7.96E+02	7.02E+02	7.11E+02	7.08E+02	7.10E+02	7.01E+02	7.01E+02	7.01E+02	7.01E+02	7.01E+02	7.01E+02
	worst	8.37E+02	7.98E+02	7.16E+02	7.15E+02	7.81E+02	8.36E+02	7.06E+02	7.01E+02	7.01E+02	7.01E+02	7.01E+02
	median	7.96E+02	7.02E+02	7.13E+02	7.13E+02	7.13E+02	7.01E+02	7.02E+02	7.01E+02	7.01E+02	7.01E+02	7.01E+02
	std	1.33E-11	3.20E+00	1.35E+00	7.83E+00	1.77E+01	1.03E+01	1.29E+00	0	5.15E-02	2.36E-02	1.32E-04
f8	best	7.75E+06	8.18E+02	1.82E+05	1.27E+03	5.18E+03	8.17E+02	8.11E+02	8.05E+02	1.00E+03	8.02E+02	8.63E+02
	worst	3.86E+07	6.67E+06	1.93E+06	8.74E+04	8.01E+04	3.47E+07	7.00E+03	8.05E+02	8.14E+03	8.08E+02	8.63E+02
	median	7.75E+06	8.19E+02	7.28E+05	4.11E+04	3.11E+04	8.23E+02	1.12E+03	8.05E+02	1.00E+02	8.02E+02	8.63E+02
	std	4.57E-08	2.45E+04	5.88E+05	2.59E+04	1.00E+03	7.55E+05	1.47E+03	0	2.07E-01	9.27E-02	9.69E-04
f9	best	9.13E+02	9.12E+02	1.00E+03	1.00E+03	1.00E+03	9.13E+02	1.00E+03	1.00E+03	1.17E+03	1.00E+03	1.00E+03
	worst	9.14E+02	9.14E+02	1.01E+03	1.00E+03	1.00E+03	9.15E+02	1.00E+03	1.00E+03	1.34E+03	1.00E+03	1.00E+03
	median	9.13E+02	9.12E+02	1.00E+03	1.00E+03	1.00E+03	9.13E+02	1.00E+03	1.00E+03	1.17E+03	1.00E+03	1.00E+03
	std	6.04E-02	8.98E-02	6.13E-01	3.05E-01	2.76E-01	4.85E-02	1.30E+00	0	3.54E-01	1.38E-01	3.26E-02
f10	best	1.21E+08	2.66E+06	1.88E+03	1.21E+03	1.24E+03	7.15E+06	1.22E+03	1.15E+03	1.10E+03	1.10E+03	1.27E+03
	worst	8.71E+08	1.15E+08	3.54E+03	1.98E+03	2.35E+03	6.50E+08	2.39E+03	1.15E+03	1.27E+03	1.24E+03	1.27E+03
	median	3.89E+07	2.66E+06	2.73E+03	1.44E+03	1.41E+03	7.15E+06	1.48E+03	1.15E+03	1.27E+03	1.14E+03	1.27E+03
	std	3.89E+07	2.20E+06	4.15E+02	2.29E+02	3.23E+02	1.18E+07	4.27E+02	0	5.78E-02	1.96E-01	1.13E-01
f11	best	1.38E+03	1.14E+03	1.71E+03	1.40E+03	1.13E+03	1.40E+03	1.10E+03	1.10E+03	1.31E+03	1.40E+03	1.40E+03
	worst	2.15E+03	1.46E+03	1.98E+03	1.41E+03	2.40E+03	1.85E+03	1.58E+03	1.10E+03	1.10E+03	1.40E+03	1.40E+03
	median	1.38E+03	1.14E+03	1.86E+03	1.41E+03	1.41E+03	1.14E+03	1.50E+03	1.10E+03	1.10E+03	1.40E+03	1.40E+03
	std	3.45E+01	7.50E+00	6.79E+01	2.27E+00	3.23E+02	3.06E+01	5.68E+01	0	7.45E-13	4.66E-02	5.78E-04
f12	best	2.11E+04	1.73E+03	1.31E+03	1.31E+03	2.16E+03	1.30E+03	1.31E+03	1.31E+03	1.30E+03	1.30E+03	1.30E+03
	worst	1.29E+06	4.27E+03	1.31E+03	1.31E+03	3.06E+03	3.13E+03	1.31E+03	1.31E+03	1.31E+03	1.30E+03	1.30E+03
	median	2.11E+04	1.75E+03	1.31E+03	1.30E+03	2.16E+03	1.30E+03	1.31E+03	1.31E+03	1.31E+03	1.30E+03	1.30E+03
	std	3.10E+04	6.81E+01	1.54E+00	5.47E-01	1.35E+00	3.77E+03	7.35E-01	0	3.62E-03	1.07E-01	4.61E-01
f13	best	3.33E+03	1.65E+03	1.30E+03	1.30E+03	1.62E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03
	worst	4.45E+03	2.59E+03	1.30E+03	1.30E+03	1.30E+03	4.15E+03	1.30E+03	1.30E+03	1.50E+03	1.30E+03	1.33E+03
	median	3.33E+03	1.66E+03	1.30E+03	1.30E+03	1.65E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03	1.30E+03
	std	7.45E+01	1.93E+01	4.03E-03	7.86E-03	5.79E-01	1.07E+02	2.42E-03	0	1.07E-08	1.80E-04	1.32E-03
f14	best	1.86E+03	1.65E+03	3.47E+04	3.27E+04	3.26E+04	1.65E+03	3.26E+04	8.07E+03	1.50E+03	3.25E+04	1.50E+03
	worst	2.28E+03	1.89E+03	3.65E+04	3.60E+04	3.67E+04	2.18E+03	3.50E+04	8.07E+03	3.39E+03	3.25E+04	1.51E+03
	median	1.86E+03	1.65E+03	3.52E+04	3.38E+04	3.36E+04	1.66E+03	3.37E+04	8.07E+03	3.39E+03	3.25E+04	1.51E+03
	std	2.43E+01	6.20E+00	5.43E+02	1.02E+03	1.37E+03	1.91E+01	8.19E+02	0	3.12E-08	2.03E-01	2.47E+00
f15	best	4.23E+03	2.63E+03	1.60E+03	1.60E+03	2.80E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03
	worst	6.38E+03	3.18E+03	1.60E+03	1.60E+03	5.93E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03
	median	4.23E+03	2.66E+03	1.60E+03	1.60E+03	2.80E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03	1.60E+03
	std	1.41E-10	7.18E+00	1.67E-01	2.23E-13	2.12E-04	6.97E+01	6.44E-13	0	0	2.54E-01	2.10E-02

search region by the charged particles. On increase in the iterations the intensity of the electric field decreases gradually and it promotes the better exploitation ability of AEFA. This pattern is clearly visible for the ribbon plots of these three functions depicted in Fig. 5.

6. Experimental analysis and results

6.1. Experimental settings

To evaluate the performance of proposed algorithm AEFA, we tested it on 15 benchmark problems on Learning-based Real-Parameter Single

Objective Optimization problems proposed in IEEE CEC 2015 [41]. The test functions are denoted as $f_1 - f_{15}$ and the summary of the test functions is given in Table 4. This test suit consist 15 benchmark functions of different types of optimization problems including unimodal ($f_1 - f_2$), simple multimodal ($f_3 - f_5$), hybrid ($f_6 - f_8$) and composite functions ($f_9 - f_{15}$).

The experiments are running on MATLAB-2014a, Window 10, CPU; intel Xeon E5-2630V3/2.4 GHz, 8-core, Intel C612, 8 GB RAM. In this paper we use 30-D functions and the results were obtained after 25 runs. The results were evaluated for a $10000 * D$ maximum number of functions evaluations (NFE) on each functions and to justify the com-

Table 7
Wilcoxon signed rank test results for CEC15 30D AEFA VS.

Function	BBO	ICMLSP	dynFWA	ABC	ACO	PSO
f_1	+	–	+	–	+	+
f_2	+	–	+	+	+	+
f_3	+	+	+	+	+	+
f_4	=	+	+	+	+	+
f_5	+	+	+	–	–	–
f_6	+	–	–	–	–	–
f_7	=	+	+	+	=	+
f_8	+	–	+	+	+	+
f_9	+	+	+	–	–	–
f_{10}	+	–	+	+	+	+
f_{11}	+	–	–	–	–	+
f_{12}	+	+	+	+	+	+
f_{13}	+	+	+	+	+	+
f_{14}	+	+	+	+	+	+
f_{15}	+	+	+	+	+	+

Table 8
Friedman test results for CEC15 30D.

Algorithm	Friedman rank	General rank
ABC	4.3013	3
ACO	5.0187	7
BBO	4.9013	6
PSO	6.552	8
ICMLSP	4.6253	5
dynFWA	4.608	4
SPS-L-SHADE-EIG	3.384	2
AEFA	2.6093	1

parison for each algorithm the same initial population was generated using MATLAB based seed function. The fine tuned parameter setting for proposed algorithm is presented in Table 5.

6.2. Evaluation on different values of Coulomb's constant K_0

We first test the performance of AEFA on each test problem with different values of K_0 ranging from $K_0 = 100$ to 500 with a step of 100 and then select the best value for K_0 . For each specified value of K_0 , we run AEFA over 1000 maximum number of iterations and record the mean fitness values. Figs. 6 and 7 demonstrate the effect of different K_0 values for problem $f_1 - f_{15}$. It can be observed from the figures that AEFA have fast convergence for most of the problems namely f_2 , f_3 , f_4 , f_8 , f_{10} , f_{12} and f_{13} for $K_0 = 500$, while for problems f_1 , f_6 , f_7 and f_9 , AEFA converges rapidly for $K_0 = 400$. For problems f_{11} and f_{15} AEFA exhibits best performance for $K_0 = 200$ and for problems f_5 and f_{14} AEFA converges rapidly for $K_0 = 100$ and $K_0 = 300$ respectively. From this experiment we found that the parameter K_0 plays important role in the convergence of AEFA and since for seven out of 15 problems AEFA have fast convergence for $K_0 = 500$, we adopt $K_0 = 500$ for all the following comparative experiments. Based on the above tuning results over the selected benchmark test suit, we suggest $K_0 = 500$ for most of the global optimization problems. The study demonstrate that the Coulomb's constant enhance the interactions between the charge particles to a better extent. It can effectively improve the diversity of the solution space and suppress the premature convergence during the search.

6.3. Comparative experiments

We compare AEFA with following state-of-art algorithms on CEC 2015 benchmark set:

1. The self-adaptive DE (SaDE) [43]
2. Particle swarm optimization (PSO) [21]
3. Artificial bee colony (ABC) [3]

4. The Ant colony optimization (ACO) [2]
5. The Water Wave Optimization (WWO) [44]
6. The FWA-EI [45]
7. The basic BBO [8]
8. ICMLSP [46]
9. dynFWA [47]
10. SPS-L-SHADE-EIG [48]

6.3.1. Results and discussion

The experiment was performed 25 times on each algorithm and record the best, worst, median and standard deviation values. The maximum number of function evaluations was fixed at 10000×30 . We do not fine tune the parameters for each problem, instead we fixed the parameters for each algorithm and the best values of parameters of each algorithms are selected on the basis of their literature. The results of 15 benchmark problems of IEEE CEC 2015 Learning-based Real-Parameter Single Objective Optimization problems are presented in Table 6. On each row of the table the best fitness and median value among all the algorithms for the corresponding problem are presented by the boldface (the bold values are better than the non-bold values).

Out of 15 for seven problems namely f_3 , f_4 , f_7 , f_{12} , f_{13} , f_{14} and f_{15} , AEFA outperformed or equally performed in comparison with all the other existing algorithms. AEFA outperformed than BBO on 12 problems and for 3 problems there is no significant difference between them. AEFA is significantly better than FWA-EI on 9 problems and worse than FWA-EI on 2 problems, better than WWO on 8 problems and worse than WWO on 2 problems and for remaining problems there is no significant difference between AEFA and FWA-EI and WWO. This shows that the search ability of AEFA is better than the migration operators of BBO and propagation operators of WWO.

In comparison with four state-of-art algorithms, AEFA has significantly better performance than PSO on 11 problems and worse than PSO on 4 problems, better than ABC on 10 problems and worse than ABC on 5 problems, better than ACO on 9 problems and worse than 5 problems and there is no significant difference between AEFA and ACO on problem f_7 . AEFA determine significantly better results than SaDE on 6 problems and worse than SaDE on 3 problems and for remaining 6 problems there is no significant difference between them.

In comparison with three CEC 2015 competition algorithms, AEFA performs significantly better than ICMLSP on 4 problems and worse than ICMLSP on 6 problems, better than dynFWA on 6 problems and worse than dynFWA on 3 problems, better than SPS-L-SHADE-EIG on 3 problems and worse than SPS-L-SHADE-EIG on 5 problems and for the remaining problems there is no significant difference between AEFA and ICMLSP, dynFWA and SPS-L-SHADE-EIG.

The statistical results shows that AEFA has significant improvement over all the other existing algorithms for most of the problems, which

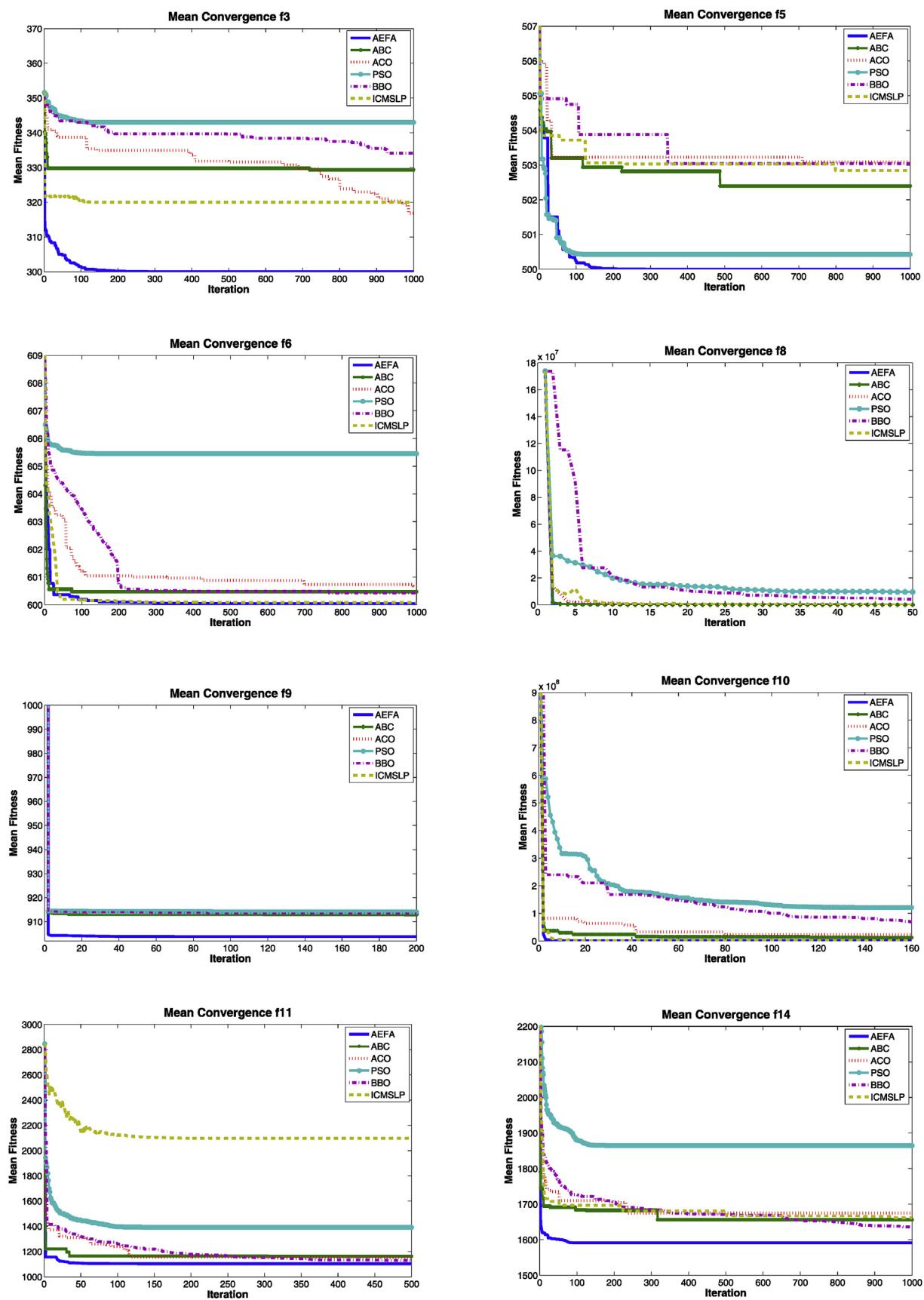


Fig. 8. Convergence Curves of Comparative Algorithms on.

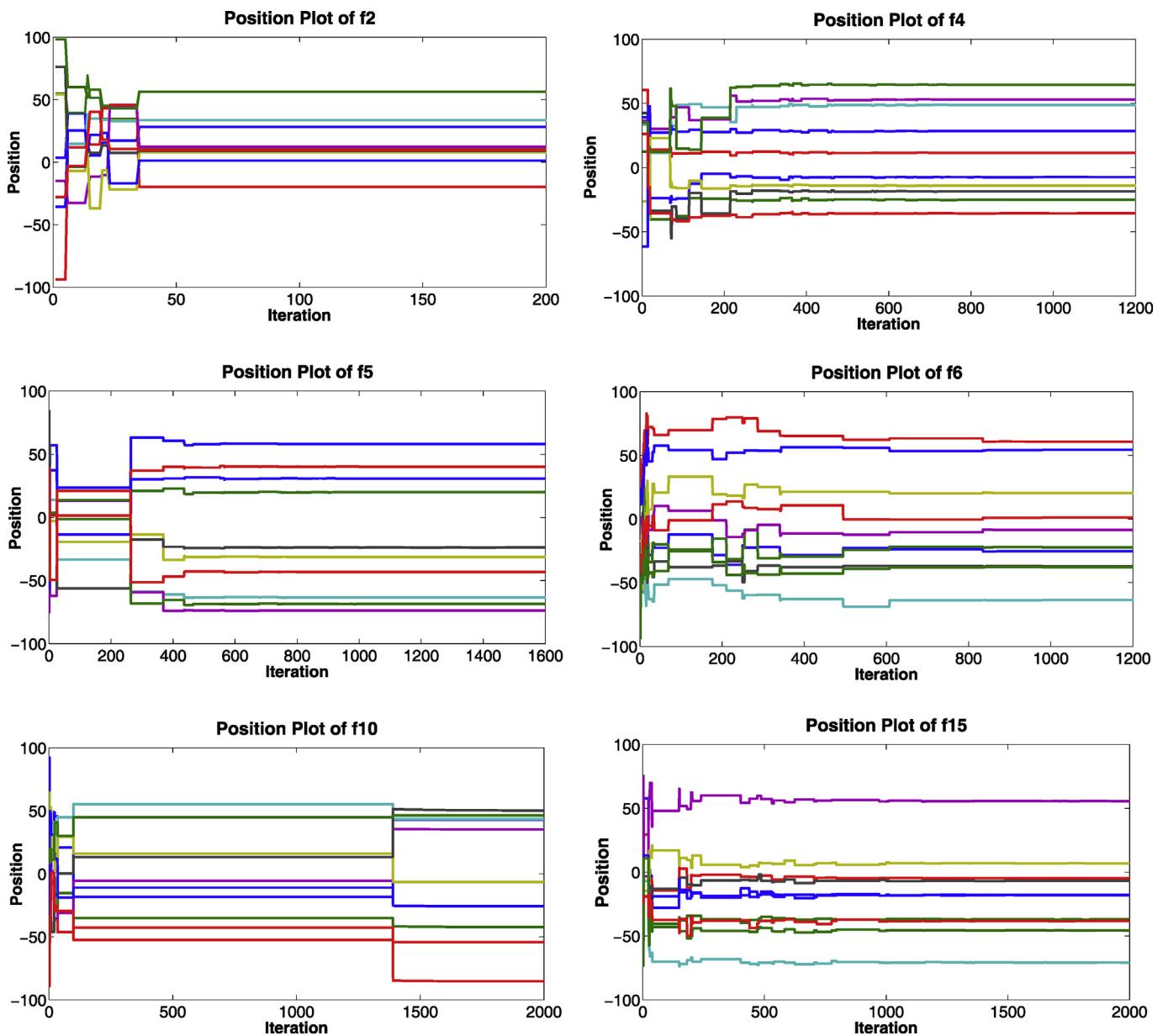


Fig. 9. Position plots for 10D of CEC15 problems.

demonstrates the effectiveness of the attraction-repulsion search techniques, formation of charge function and Coulomb's constant which adjusts the accuracy of the search space. Hence based on the selected benchmark problems we were able to conclude that AEFA is superior or comparable to the other algorithms.

6.4. Statistical test

In order to analyze the experimental results we performed two non-parametric test Wilcoxon signed rank test and Friedman test [49]. Wilcoxon signed rank test is used for pairwise comparison of algorithm performance. The test is performed for best fitness values of 25 runs over 1000 iterations of each algorithms. In this study the test is performed for AEFA versus BBO, ICMLSP, dynFWA, ABC, ACO and PSO for 30D with $\alpha = 0.05$ significance level. Table 7 represents results of Wilcoxon signed rank test. In Table 7 +, -, and = shows that the AEFA outperformed, poorly and equally performed in comparison to other algorithms respectively. The majority of + sign in the table justify that AEFA is significantly better than other algorithms.

The Friedman test is used to find the differences in treatments or algorithms across multiple test attempts. This test ranks the data within each row (or block) and then tests for a difference across columns. We adopt Friedman test to compare the comprehensive performance of each algorithm on a set of 15 problems of CEC 2015 benchmark. Hence, results of Friedman test provides an overview of the algorithm performance. The results of Friedman test is presented in Table 8. The general rank in table indicate the general performance of the algorithms over the selected benchmark set. From the results we can see that AEFA obtains best Friedman rank. The order of performance of the algorithms over this test suit is AEFA, SPS-L-SHADE-EIG, ABC, dynFWA, ICMLSP, BBO, ACO and PSO. The test results indicate the effectiveness of proposed attraction-repulsion based search strategy of AEFA and its out-performance over all the other existing algorithms.

6.5. Convergence plots

The convergence curves of the comparative algorithms are presented in Fig. 8 on some selected test problems over maximum 1000 iterations.

Table 9
Computational complexity for 30D.

Function	ABC	ACO	PSO	BBO	AEFA
f_1	5.01E+01	1.86E+01	5.21E+01	5.57E+01	7.94E+01
f_2	1.03E+02	4.08E+01	4.34E+01	1.14E+02	8.49E+01
f_3	2.29E+02	1.89E+01	1.07E+02	1.35E+02	8.58E+01
f_4	3.38E+02	9.58E+00	1.27E+02	2.46E+02	8.12E+01
f_5	4.25E+02	9.57E+01	1.79E+02	2.84E+02	7.86E+01
f_6	5.07E+02	1.47E+02	1.28E+02	3.72E+02	8.03E+01
f_7	5.60E+02	1.69E+02	1.11E+02	4.30E+02	7.87E+01
f_8	6.13E+02	1.91E+02	1.45E+02	4.89E+02	8.05E+01
f_9	6.67E+02	2.13E+02	1.40E+02	5.48E+02	8.42E+01
f_{10}	7.22E+02	2.34E+02	1.59E+02	6.06E+02	8.41E+01
f_{11}	7.93E+02	2.49E+02	1.62E+02	6.58E+02	8.22E+01
f_{12}	8.63E+02	2.79E+02	1.74E+02	7.26E+02	8.29E+01
f_{13}	9.31E+02	3.00E+02	1.80E+02	7.84E+02	8.01E+01
f_{14}	1.00E+03	3.26E+02	1.38E+02	8.48E+02	8.12E+01
f_{15}	1.15E+03	3.07E+02	2.63E+02	8.70E+02	8.24E+01

From the figures we observed that for all the eight problems namely f_3 , f_5 , f_6 , f_8 , f_9 , f_{10} , f_{11} and f_{14} both the convergence speeds and the final results of our approach AEFA are better than all the other existing algorithms. These convergence curves shows that AEFA has a better balance between exploration and exploitation than the other existing algorithms and the convergence of AEFA is more rapid and more accurate in comparison with all the other algorithms.

6.6. Position plots

In this section we plots the position graphs of some selected benchmarks problems for 10D. Fig. 9 show the approaches of different dimensions of functions f_2 , f_4 , f_5 , f_6 , f_{10} and f_{15} towards the best fitness value over the iterations. For most of the problems we found that the different dimension of the problem reaches to its near global minimum value within 1000 iterations. Only for problem f_{10} the pattern is interestingly different than the others. For f_{10} the diversity is initially high for 100 iterations then constant for large period 100 to 1390 iterations again diverse and then finally reach to near optima after 1400 iterations.

6.7. Algorithm complexity

The time complexity of the AEFA and other algorithms are studied based on the strategy defined in CEC15 Learning-based Real-Parameter Single Objective Optimization problems [41] for 30D. The strategy for the measurement of algorithm complexity is presented in [Algorithm 2](#).

Algorithm 2 Strategy for the measurement of algorithm complexity.

Run the test program below:

for $i = 1:1000000$ **do**

```
x = 0.55 + (double)i;
x = x + x; x = x/2; x = x * x; x = sqrt(x); x = log(x);
x = exp(x); x = x/(x + 2);
```

end for

Computing time for the above = T_0 ;

Computing time for each problem with 200000 function evaluations = T_1

The complete computing time for the algorithm with 200000 evaluations for each problem = T_2

Execute above step five times and get five T_2 values.

$T_2 = \text{Mean}(T_2)$

The complexity of the algorithm is reflected by: T_0 , T_1 , T_2 and $(T_2 - T_1)/T_0$

The comparative analysis of the algorithm complexity of AEFA with ABC, PSO, ACO and BBO is presented in [Table 9](#) for 30D problems. It has been observed from the table that the time complexity of our

approach AEFA is significantly less than ABC and BBO on 14 problems and ACO on 10 problems and for the remaining problems slightly more than the existing algorithms. In comparison with PSO the time complexity of AEFA is significantly less than for 13 problems and more than for 2 problems. Therefore in terms of time complexity AEFA is outperformed than the other existing algorithms.

7. Conclusion

In this article a novel optimization algorithm artificial electric field algorithm (AEFA) is introduced. The AEFA is based on the Coulomb's law of electrostatic force and Newton's law of motion. In AEFA the particles are considered as charged particles and move in the search space due to the Coulomb's electrostatic force. This force is attractive and a way of transferring the information between charges. The validity of designed algorithm AEFA was tested numerically as well as theoretically. To check the validity of designed algorithm CEC 2015 Learning-based Real-Parameter Single Objective Optimization problems are solved by AEFA. The presented results are explained and discussed in a variety of ways, which shows the validity of purposed algorithm. Then we investigate the stability conditions for the convergence of AEFA and provide the theoretical proof for that, which confirm that the algorithm AEFA tends to a stable stage.

Acknowledgement

The authors are thankful to the National Institute of Technology Uttarakhand and Dr. B.R. Ambedkar National Institute of Technology Jalandhar for providing necessary support to this research.

References

- [1] R. Eberhart, J. Kennedy, A new optimizer using particle swarm theory, in: Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium on, IEEE, 1995, pp. 39–43.
- [2] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, *IEEE Trans. Evol. Comput.* 1 (1) (1997) 53–66.
- [3] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm, *J. Glob. Optim.* 39 (3) (2007) 459–471.
- [4] G.-G. Wang, X. Zhao, S. Deb, A novel monarch butterfly optimization with greedy strategy and self-adaptive, in: Soft Computing and Machine Intelligence (ISCM), 2015 Second International Conference on, IEEE, 2015, pp. 45–50.
- [5] X.-S. Yang, S. Deb, Cuckoo search via lévy flights, in: Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on, IEEE, 2009, pp. 210–214.
- [6] K.-S. Tang, K.-F. Man, S. Kwong, Q. He, Genetic algorithms and their applications, *IEEE Signal Process. Mag.* 13 (6) (1996) 22–37.
- [7] R. Storn, K. Price, Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces, *J. Glob. Optim.* 11 (4) (1997) 341–359.

- [8] D. Simon, Biogeography-based optimization, *IEEE Trans. Evol. Comput.* 12 (6) (2008) 702–713.
- [9] R.V. Rao, Teaching-learning-based optimization algorithm, in: *Teaching Learning Based Optimization Algorithm*, Springer, 2016, pp. 9–39.
- [10] B. Xing, W.-J. Gao, Imperialist competitive algorithm, in: *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*, Springer, 2014, pp. 203–209.
- [11] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, Gsa: a gravitational search algorithm, *Inf. Sci.* 179 (13) (2009) 2232–2248.
- [12] A. Kaveh, S. Talatahari, A novel heuristic optimization method: charged system search, *Acta Mech.* 213 (3) (2010) 267–289.
- [13] M.-H. Tayarani-N, M. Akbarzadeh-T, Magnetic optimization algorithms a new synthesis, in: *Evolutionary Computation, 2008. CEC 2008. (IEEE World Congress on Computational Intelligence)*. IEEE Congress on, IEEE, 2008, pp. 2659–2664.
- [14] H. Shah-Hosseini, The intelligent water drops algorithm: a nature-inspired swarm-based optimization algorithm, *Int. J. Bio-Inspired Comput.* 1 (1–2) (2009) 71–79.
- [15] Z.W. Geem, J.H. Kim, G.V. Loganathan, A new heuristic optimization algorithm: harmony search, *Simulation* 76 (2) (2001) 60–68.
- [16] A.H. Kashan, An efficient algorithm for constrained global optimization and application to mechanical engineering design: league championship algorithm (lca), *Comput. Aided Des.* 43 (12) (2011) 1769–1792.
- [17] A.Y. Lam, V.O. Li, Chemical-reaction-inspired metaheuristic for optimization, *IEEE Trans. Evol. Comput.* 14 (3) (2010) 381–399.
- [18] B. Alatas, Acroa: artificial chemical reaction optimization algorithm for global optimization, *Expert Syst. Appl.* 38 (10) (2011) 13170–13180.
- [19] X. Jin, R.G. Reynolds, Using knowledge-based evolutionary computation to solve nonlinear constraint optimization problems: a cultural algorithm approach, in: *Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on*, vol. 3, IEEE, 1999, pp. 1672–1678.
- [20] M. Ghasemi, M. Taghizadeh, S. Ghavidel, A. Abbasian, Colonial competitive differential evolution: an experimental study for optimal economic load dispatch, *Appl. Soft Comput.* 40 (2016) 342–363.
- [21] R. Poli, J. Kennedy, T. Blackwell, Particle swarm optimization, *Swarm Intell.* 1 (1) (2007) 33–57.
- [22] J.J. Liang, A.K. Qin, P.N. Suganthan, S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, *IEEE Trans. Evol. Comput.* 10 (3) (2006) 281–295.
- [23] S. Alam, G. Dobbie, S.U. Rehman, Analysis of particle swarm optimization based hierarchical data clustering approaches, *Swarm Evolut. Comput.* 25 (2015) 36–51.
- [24] M.S. Nobile, P. Cazzaniga, D. Besozzi, R. Colombo, G. Mauri, G. Pasi, Fuzzy self-tuning pso: a settings-free algorithm for global optimization, *Swarm Evolut. Comput.* (2018).
- [25] M. Dorigo, M. Birattari, Ant colony optimization, in: *Encyclopedia of Machine Learning*, Springer, 2011, pp. 36–39.
- [26] M. Metlicka, D. Davendra, Chaos driven discrete artificial bee algorithm for location and assignment optimisation problems, *Swarm Evolut. Comput.* 25 (2015) 15–28.
- [27] Y. Wang, Z.-Z. Liu, J. Li, H.-X. Li, G.G. Yen, Utilizing cumulative population distribution information in differential evolution, *Appl. Soft Comput.* 48 (2016) 329–346.
- [28] R. Mallipeddi, P.N. Suganthan, Q.-K. Pan, M.F. Tasgetiren, Differential evolution algorithm with ensemble of parameters and mutation strategies, *Appl. Soft Comput.* 11 (2) (2011) 1679–1696.
- [29] Y. Wang, H.-X. Li, T. Huang, L. Li, Differential evolution based on covariance matrix learning and bimodal distribution parameter setting, *Appl. Soft Comput.* 18 (2014) 232–247.
- [30] K. Opara, J. Arabas, Comparison of mutation strategies in differential evolution—a probabilistic perspective, *Swarm Evolut. Comput.* (2018).
- [31] S. Jiang, Y. Wang, Z. Ji, Convergence analysis and performance of an improved gravitational search algorithm, *Appl. Soft Comput.* 24 (2014) 363–384.
- [32] A. Yadav, K. Deep, J.H. Kim, A.K. Nagar, Gravitational swarm optimizer for global optimization, *Swarm Evolut. Comput.* 31 (2016) 64–89.
- [33] X. Yuan, J. Zhao, Y. Yang, Y. Wang, Hybrid parallel chaos optimization algorithm with harmony search algorithm, *Appl. Soft Comput.* 17 (2014) 12–22.
- [34] H. Liu, Z. Cai, Y. Wang, Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization, *Appl. Soft Comput.* 10 (2) (2010) 629–640.
- [35] I. Boussaïd, A. Chatterjee, P. Siarry, M. Ahmed-Nacer, Hybridizing biogeography-based optimization with differential evolution for optimal power allocation in wireless sensor networks, *IEEE Trans. Veh. Technol.* 60 (5) (2011) 2347–2353.
- [36] İ. Karen, A.R. Yıldız, N. Kaya, N. Öztürk, F. Öztürk, Hybrid approach for genetic algorithm and taguchi's method based design optimization in the automotive industry, *Int. J. Prod. Res.* 44 (22) (2006) 4897–4914.
- [37] J. Robinson, S. Sinton, Y. Rahmat-Samii, Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna, in: *Antennas and Propagation Society International Symposium, 2002. IEEE*, vol. 1, IEEE, 2002, pp. 314–317.
- [38] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, *IEEE Trans. Evol. Comput.* 1 (1) (1997) 67–82.
- [39] E. Taillard, Few guidelines for analyzing methods, in: *Tutorial, 6th Metaheuristics Int. Conf.*, 2005.
- [40] D. Holliday, R. Resnick, J. Walker, *Fundamentals of Physics*, John Wiley and Sons, 1993.
- [41] J. Liang, B. Qu, P. Suganthan, Q. Chen, Problem Definitions and Evaluation Criteria for the Cec 2015 Competition on Learning-Based Real-Parameter Single Objective Optimization, Technical Report 201411A, Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2014.
- [42] K. Åstrom, B. Wittenmark, *Computer Controlled System*, 1997.
- [43] A.K. Qin, V.L. Huang, P.N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, *IEEE Trans. Evol. Comput.* 13 (2) (2009) 398–417.
- [44] Y.-J. Zheng, Water wave optimization: a new nature-inspired metaheuristic, *Comput. Oper. Res.* 55 (2015) 1–11.
- [45] B. Zhang, Y.-J. Zheng, M.-X. Zhang, S.-Y. Chen, Fireworks algorithm with enhanced fireworks interaction, *IEEE ACM Trans. Comput. Biol. Bioinform.* 14 (1) (2017) 42–55.
- [46] L. Chen, C. Peng, H.-L. Liu, S. Xie, An improved covariance matrix leaning and searching preference algorithm for solving cec 2015 benchmark problems, in: *Evolutionary Computation (CEC), 2015 IEEE Congress on*, IEEE, 2015, pp. 1041–1045.
- [47] C. Yu, L.C. Kelley, Y. Tan, Dynamic search fireworks algorithm with covariance mutation for solving the cec 2015 learning based competition problems, in: *Evolutionary Computation (CEC), 2015 IEEE Congress on*, IEEE, 2015, pp. 1106–1112.
- [48] S.-M. Guo, J.S.-H. Tsai, C.-C. Yang, P.-H. Hsu, A self-optimization approach for l-shade incorporated with eigenvector-based crossover and successful-parent-selecting framework on cec 2015 benchmark set, in: *Evolutionary Computation (CEC), 2015 IEEE Congress on*, IEEE, 2015, pp. 1003–1010.
- [49] J. Derrac, S. García, D. Molina, F. Herrera, A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms, *Swarm Evolut. Comput.* 1 (1) (2011) 3–18.