Swarm and Evolutionary Computation ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Swarm and Evolutionary Computation

journal homepage: www.elsevier.com/locate/swevo

Regular Paper

Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm

Hosein Abedinpourshotorban [a,b], Siti Mariyam Shamsuddin [a], Zahra Beheshti [a], Dayang N.A. Jawawi [b]

[a] UTM Big Data Centre, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
[b] Department of Software Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia

ARTICLE INFO

Article history:
Received 18 December 2014
Received in revised form
18 July 2015
Accepted 21 July 2015

Keywords:
Global optimization
Metaheuristics
Population-based optimization
Golden ratio
Evolutionary algorithms

ABSTRACT

This paper presents a physics-inspired metaheuristic optimization algorithm, known as Electromagnetic Field Optimization (EFO). The proposed algorithm is inspired by the behavior of electromagnets with different polarities and takes advantage of a nature-inspired ratio, known as the golden ratio. In EFO, a possible solution is an electromagnetic particle made of electromagnets, and the number of electromagnets is determined by the number of variables of the optimization problem. EFO is a population-based algorithm in which the population is divided into three fields (positive, negative, and neutral); attraction–repulsion forces among electromagnets of these three fields lead particles toward global minima. The golden ratio determines the ratio between attraction and repulsion forces to help particles converge quickly and effectively. The experimental results on 30 high dimensional CEC 2014 benchmarks reflect the superiority of EFO in terms of accuracy and convergence speed over other state-of-the-art optimization algorithms.

1. Introduction

Metaheuristic algorithms are used to find an approximate optimal solution for difficult optimization problems, for which there is no deterministic method to solve them within a reasonable time. A metaheuristic algorithm is defined as a problem independent algorithm that can find approximate solutions to hard problems. Metaheuristics are inspired by nature and try to solve problems by mimicking ethology, biology, or physics [1].

Evolutionary algorithms (EAs) are stochastic, population-based metaheuristic algorithms. EAs differ from some optimization methods, such as Simulated-Annealing [2] and Tabu search [3], because they evolve a population of solutions to reach an approximate optimal solution instead of one solution [4]. Generally, EAs search the problems domain as follows: a population of random individuals (solutions) is initialized for the first time, and then the fitness of individuals are evaluated by the fitness function. In the next generations, individuals evolve towards the global best solution by means of EAs and the guidance of the fitness function. This process continues until it reaches the maximum number of iterations or finds the expected near-optimum solution.

The ability to balance between exploration (diversification) and exploitation (intensification) plays a significant role in the success of an EA. Exploitation is required to explore the problem surface globally and identify the area of the search space that contains the global best solution (global minima). Exploitation is required to find an accurate solution by intensifying the search in the area that is determined by the exploration stage. Achievement of this balance is the main characteristic of EAs and the way they differ from each other [5]. Generally, EAs are more exploration-oriented rather than exploitation oriented. This characteristic makes them suitable for hard problems with lots of local optimal solutions (local minima) because they keep a population of solutions and investigate a large area to find the global best solution.

Several well-known, nature-inspired EAs are: Genetic Algorithm (GA), which works based on the principle of the Darwinian theory of survival [6,7], Particle Swarm Optimization (PSO) [8–11], which works based on the foraging behavior of a swarm of birds [7], Differential Evolution (DE) [12], which is similar to GA with modified crossover and mutation methods, and Harmony Search (HS) [13,14], which works based on the way that musicians experiment and change their instruments' pitches to improvise better harmonies. These well-known optimization algorithms

E-mail addresses: h_abedinpour@gmail.com (H. Abedinpourshotorban), mariyam@utm.my (S. Mariyam Shamsuddin), bzahra2@live.utm.my (Z. Beheshti), dayang@utm.my (D.N.A. Jawawi).

http://dx.doi.org/10.1016/j.swevo.2015.07.002
2210-6502/

have been applied for the optimization of many optimization problems [15] and have demonstrated acceptable accuracy and speed. However, the challenge in this area is how to be inspired from the existing knowledge about optimization and the way in which nature behaves to improve the existing algorithms or develop a new algorithm, which can quickly find the best solution globally while avoiding optimal solutions locally.

Two recently proposed nature-inspired optimization algorithms are Artificial Bee Colony (ABC) [16,17] and Group Search Optimizer (GSO) [18]. ABC is an optimization algorithm based on the intelligent foraging behavior of honey bee swarms, and the GSO algorithm is inspired by animals' food search behavior. Both algorithms have been successfully applied to various optimization problems and have demonstrated competitive search power.

Physics-inspired heuristics are another type of EAs, that are motivated by physics laws. In 2008, Tayarani [15,19] proposed a magnetic-inspired optimization algorithm called Magnetic Optimization Algorithm (MOA), which is inspired by the attraction force among magnetic particles. In MOA, possible solutions are presented by magnets with different mass, which are scattered in a lattice-like structure all over the search space and apply a force of attraction to their neighbors, according to their fitness. A similar algorithm to MOA is the Gravitational Search Algorithm (GSA) [20], which considers the search agents (particles) as masses that attract each other based on the gravitational forces between them. Some other recent algorithms have used a similar idea for optimization based on the physics-inspired laws of forces. Magnetic Charged System Search (MCSS) [21] utilizes the governing laws for magnetic forces in addition to electrical forces for optimization. Ions Motion Optimization (IMO) [22] is proposed based on the attraction and repulsion of anions and cations to perform optimization.

Motivated by the previously proposed force-based algorithms and the fact that in most of them, particles are attracted to the fittest particle, which increases the chances of finding local minima, this paper proposes a physics-inspired optimization algorithm called Electromagnetic Field Optimization (EFO), which is inspired by two phenomena. The first phenomenon is the attraction–repulsion forces among electromagnets with different polarities, and the second phenomenon is a nature-inspired ratio called the golden ratio [23], which is also known as god's fingerprint. In our algorithm, particles move a distance away from particles with low fitness (bad solutions) and get closer to the fittest particles (good solutions) based on the attraction–repulsion forces among three electromagnetic fields. The repulsion force helps particle to avoid local minima and the attraction force leads particles toward global minima. Experimental results on the CEC 2014 benchmarks show that our proposed algorithm outperforms existing algorithms.

2. Electromagnetic field optimization

An electromagnet is a type of magnet in which electrical current produces a magnetic field. In contrast to the permanent magnet, an electromagnet has single polarity (positive or negative), which is determined by the direction of the electrical current and can be changed by changing the direction of the electrical current. Moreover, there are two different forces among electromagnets: attraction and repulsion. Electromagnets with the same polarity repel each other, and those with opposite polarity attract each other. The attraction force among electromagnets is (5–10%) stronger than the repulsion force. Our algorithm uses these concepts and replaces the ratio between attraction and repulsion forces with the golden ratio. This helps particles to adequately investigate the problem search space and find a near-optimal solution.

EFO is a population-based algorithm and each solution vector is represented by one group of electromagnets (electromagnetic particle). The number of electromagnets of an electromagnetic particle is determined by the number of variables of the optimization problem. Therefore, each electromagnet of the electromagnetic particle corresponds to one variable of the optimization problem. Moreover, all electromagnets of the same electromagnetic particle have the same polarity. However, each electromagnet can apply a force of attraction or repulsion on the peer-electromagnets that correspond to the same variable of the optimization problem.

EFO searches the problems domain as follows: first, a population of electromagnetic particles is generated randomly, and the fitness of each particle is evaluated by a fitness function; then, particles are sorted according to their fitness. Second, sorted particles are divided into three groups, and a portion of the electromagnetic population is allocated to each group; the first group is called the positive field and consists of the fittest electromagnetic particles with positive polarity, the second group is called the the negative field and consists of the electromagnetic particles with the lowest fitness and negative polarity, and the remaining electromagnetic particles form a group called the neutral field, which has a small negative polarity almost near zero. Finally, in each iteration of the algorithm, a new electromagnetic particle is shaped and evaluated by a fitness function. If the generated electromagnetic particle is fitter than the worst electromagnetic particle in the population, then the generated particle will be inserted into the sorted population according to its fitness and obtain a polarity based on its position in the population; moreover, the worst particle will be eliminated. This process continues until it reaches the maximum number of iterations or finds the expected near-optimal solution.

EFO determines the position of each electromagnet of a generated electromagnetic particle as follows: from the electromagnetic particles of each electromagnetic field (positive, negative, and neutral), three peer electromagnets are randomly selected (one electromagnet from each field). Afterwards, the generated electromagnet gets the position and polarity (small negative polarity) of the selected electromagnet from the neutral

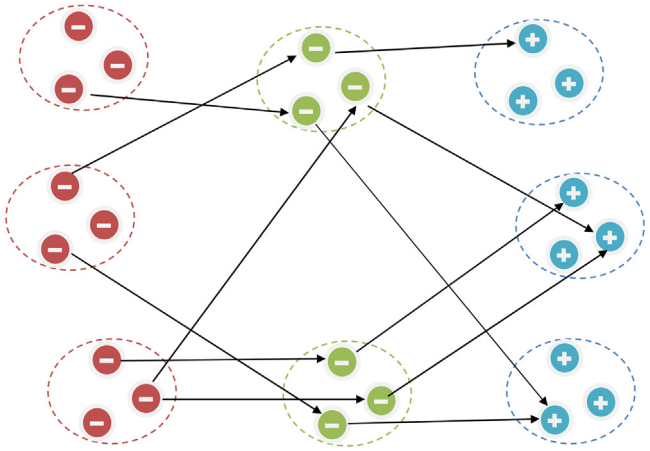

Fig. 1. Direction of forces among electromagnets. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

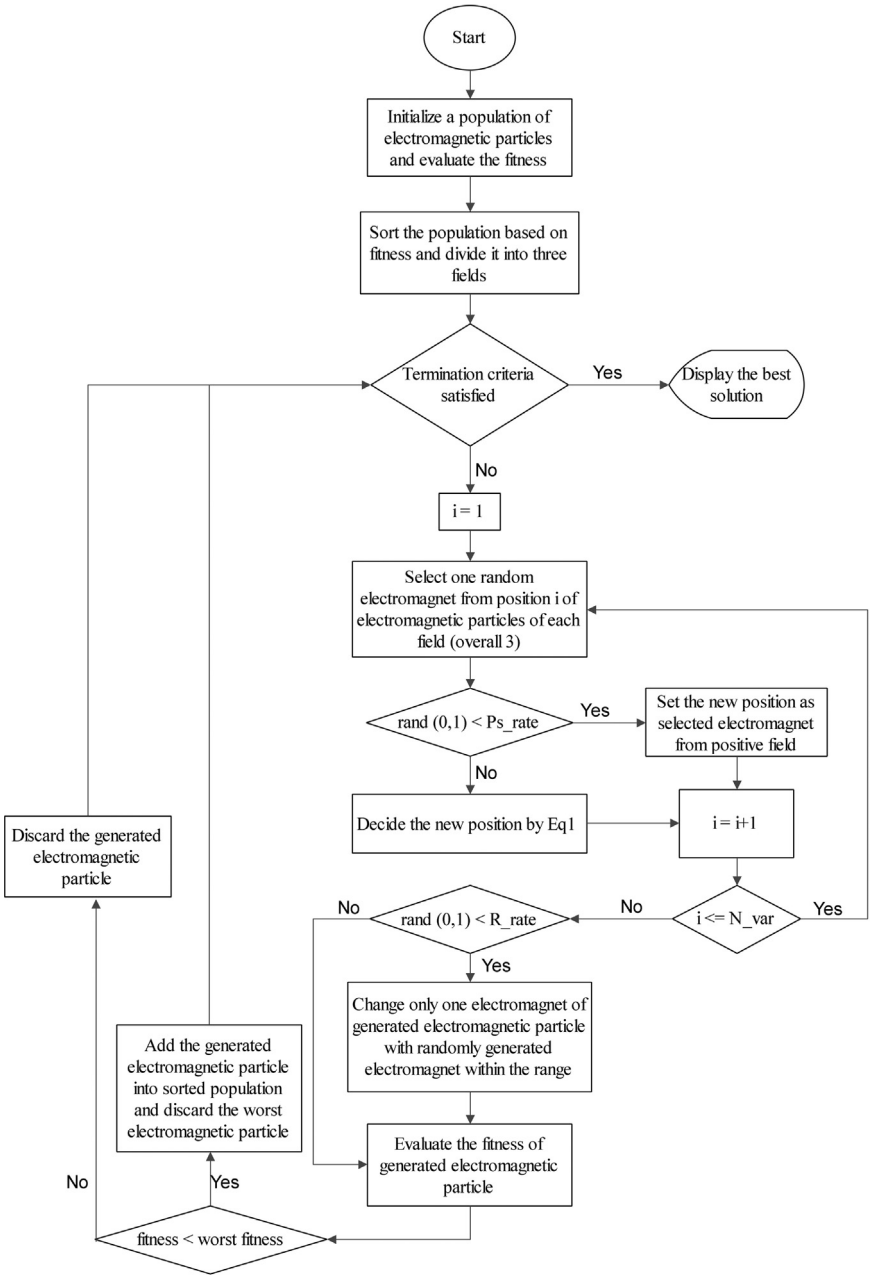

Fig. 2. General flow of EFO.

field and gets affected by the selected electromagnets from the positive field (attraction) and negative field (repulsion) by random force intensity. In other words, the generated electromagnet moves a distance away from the bad solutions and approaches good solutions. Fig. 1 shows the direction of forces among electromagnets of different fields; in this figure, positive, neutral, and negative fields are colored, respectively, as blue, green, and red. As illustrated in Fig. 1, each electromagnet of the electromagnetic particles of the neutral field is affected by two random peer electromagnets from the negative field (repulsion) and positive field (attraction).

The coexistence of two opposite forces among electromagnets and the fact that the new solution is generated by moving a distance away from bad solutions and moving closer to the good solutions cause effective search and fast convergence. However, to keep diversity and avoid local minima, randomness is an indispensable part of EFO. Therefore, for some of the generated electromagnetic particles (not all), only one of the electromagnets is changed with a randomly generated electromagnet. The reason for applying randomness to some electromagnetic particles is that the existence of the random variables in all solutions will prevent finding a good solution. However, applying randomness to some solutions brings diversity into the population and prevents local minima. The general flow of EFO is presented in Fig. 2, and the pseudo-code of the proposed algorithm is presented in Algorithm 1.

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

Algorithm 1.

```
1  /*Part 1: algorithm parameter initialization*/
2  Input: f(x)/*objective function*/
3  Input: N_var=number of electromagnets of electromagnetic particle/* number of problem variables */
4  Input: N_emp=number of electromagnetic particles in population
5  Input: R_rate=probability of changing one electromagnet of generated particle with a random electromagnet
6  Input: Ps_rate=probability of selecting electromagnets of generated particle from the positive field */without change*/
7  Input: P_field=portion of population, which belongs to positive field */good solutions */
8  Input: N_field=portion of population, which belongs to negative field */bad solutions */
9  Input: min= lower boundary; max=upper boundary
10 Constant: phi=1.6180339887498948/* golden ratio */
11 /*Part 2: initialization of a random electromagnetic field*/
12 for i=1 to N_emp do
13   for j=1 to N_var do
14       em_pop[i, j]=min +rand(0,1)* (max–min)
15       new_emp[j]=em_pop [i, j]
16   end for
17       fit[i]=f(new_emp)
18 end for
19 sortPopulation(em_pop,fit)/* sort population from best to worst based on fitness*/
20 /*part 3: main loop (optimization)*/
21 RI=1/*index of one electromagnet of generated particle, which might be initialized by random number*/
22 while (stop criterion is not met) do
23     force= rand(0,1)/* should be outside of the loop, to be the same for all electromagnets of generated particle */
24     for i=1 to N_var do
25     I_pos= rand(1, floor(N_emp* P_field) )/* Index of a random particle from positive field*/
26       I_neg=rand( floor((1 - N_field)* N_emp), N_emp )/* Index of a random particle from negative field*/
27       I_neu=rand( ceil(N_emp* P_field), ceil((1 - N_field)* N_emp) )/* Index of a random particle from neutral field*/
28       if (rand(0,1) < Ps_rate) then
29         new_emp[i]=em_pop [I_pos, i]
30       Else
31         new_emp[i]=em_pop[I_neu, i]+phi * force * (em_pop[I_pos, i] - em_pop[I_neu, i]) -  force * (em_pop[I_neg, i] -
   em_pop[I_neu, i])/*search strategy*/
32     end if
33     If (new_emp[i] > max or new_emp[i] < min ) then
34       new_emp[i]=min +rand(0,1)* (max–min)/* replace with random value if outside boundary*/
35     end if
36   end for
37   if ( rand(0,1) < R_rate) then*/randomization of one electromagnet of some generated electromagnetic particles*/
38     new_emp[RI]=min +rand(0,1)* (max–min)
39     RI++
40     if (RI > N_var) then
41       RI=1
42   end if
43   end if
44   New_fit=f(new_emp)/*fitness of generated particle*/
45   If (New_fit < worst(fit) ) then
46     insertInSortedPopulation(new_emp)/* insert in the sorted population based on fitness*/
47   end if
48 end while
49 return em_pop[1,1: N_var]
```

Line 31 of the EFO pseudo-code, which is labeled as *search strategy*, is the most important part of the algorithm and the only mathematical equation in the EFO algorithm. In this part, a randomly selected electromagnet from the neutral field is affected (virtually) by selected electromagnets from the positive field (attraction) and negative field (repulsion) to determine the position of the generated electromagnet. The new position is calculated as follows:

$$\mathrm{EMP}_j^{\mathrm{New}} = \mathrm{EMP}_j^{K_j} + \left((\varphi * r) * \left(\mathrm{EMP}_j^{P_j} - \mathrm{EMP}_j^{K_j}\right)\right) - \left(r * \left(\mathrm{EMP}_j^{N_j} - \mathrm{EMP}_j^{K_j}\right)\right) \quad (1)$$

where EMP is the electromagnetic particle; r is the random value between 0 and 1 (generated once for each new electromagnetic

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

particle); j is the variable index (index of generated electromagnet); K is the random index from neutral field (generated for each electromagnet of generated particle); P is the random index from positive field (generated for each electromagnet of generated particle); N is the random index from negative field (generated for each electromagnet of generated particle).

For generation of a new electromagnetic particle, Eq. (1) is calculated for all of the electromagnets (variables) of the generated electromagnetic particle (solution). For instance, when the optimization problem has five variables, Eq. (1) should be calculated for (j; 1:5) to generate a new solution. This means that several particles cooperate to generate a new electromagnetic particle. The minimum number of cooperative particles to shape a new particle is 3, when for all electromagnets of the generated particle, the random function selects electromagnets from the same particles, and the maximum number of cooperative particles is calculated by multiplying the number of problem variables by three. For example, when the number of problem variables is 5, the minimum number of cooperative particles to shape a new particle is 3 and the maximum is 15. Meanwhile, in other optimization algorithms, this number is considerably lower than that of EFO. For example, in GA, only 2 solutions (parents) shape a new solution [24], and in PSO, three different positions (current position, personal best, and global best) determine the new position of the particles [25].

The new electromagnetic particle generation process for the optimization of function A is demonstrated in Fig. 3; in this figure, positive, neutral, and negative fields are colored, respectively, as blue, green, and pink.

A. The optimization function, defined as

$$f(x) = \left(\frac{1}{4}x_1\right)^2 + \left(\frac{1}{2}x_2\right)^2$$

where global optimum $x^* = 0$ and $f(x^*) = 0$.

The high degree of cooperation among particles is the most important characteristic of EFO, which causes fast convergence towards global minima. Another important feature of EFO is randomization, which brings diversity into the population and avoids finding local minima. Moreover, EFO takes advantage of the golden ratio to search the optimization problem search space effectively. These characteristics make EFO a robust optimization algorithm.

2.1. Derivation of Eq. (1)

Eq. (1) is composed of two parts: the first part is calculating the distance between the randomly selected electromagnets from the positive field and neutral field by using Eq. (2), and the second part is calculating the distance between the randomly selected electromagnets from the negative field and neutral field by using Eq. (3). To make the neutral electromagnets move toward the positive electromagnets and move a distance away from the negative electromagnets, we add the position of the neutral electromagnets to a random portion of the calculated distance by using Eq. (2) and negative of the calculated distance by using Eq. (3). As we mentioned earlier, the force power of the positive field is greater than the force power of the negative field. Therefore, in Eq. (4), we multiplied the calculated distance between the positive and neutral electromagnets by Phi (approximately 1.61), which is proven to have the best performance based on our experiments. By combining Eqs. (2)–(4), we will obtain Eq. (1).

$$D_j^{P_jK_j} = \mathrm{EMP}_j^{P_j} - \mathrm{EMP}_j^{K_j} \tag{2}$$

$$D_j^{N_jK_j} = \mathrm{EMP}_j^{N_j} - \mathrm{EMP}_j^{K_j} \tag{3}$$

$$\mathrm{EMP}_j^{\mathrm{New}} = \mathrm{EMP}_j^{K_j} + \left((\varphi * r) * D_j^{P_jK_j}\right) - \left(r * D_j^{N_jK_j}\right) \tag{4}$$

2.2. EFO parameters setting

Like all optimization algorithms, parameters setting plays a significant role in the performance of EFO. The most important parameter of EFO is N_emp, which determines the number of electromagnetic particles of the population. A small number of particles inside the population will cause finding local minima instead of global minima due to the lack of knowledge about the search space. Additionally, a large population will lead to slow convergence. We experimentally found out that a population smaller than 50 tends to find local minima even for low-dimensional problems, and a population greater than problem dimension (D) increases the computational time.

In EFO, the population is divided into three groups with different polarities, and the P_field and N_field parameters determine the percentage of the allocated population to each group. P_field is a portion of the population assigned to the positive field. Based on our experiments, the best value for P_field is between 0.05 and 0.1. N_field determines a portion of the population allocated to the negative field, and the remaining is allocated to the neutral field. Experimental results show that the best value for N_field is between 0.4 and 0.5.

Other important parameters of EFO are Ps_rate and R_rate. Ps_rate is the probability of selecting electromagnets of the generated electromagnetic particle from electromagnets of the positive field without changing them, and R_rate is the possibility of changing one electromagnet of the generated electromagnetic particle with a randomly generated electromagnet. Based on our experiments, the best values for Ps_rate and R_rate are between 0.1 and 0.4.

A comprehensive study on EFO parameters is presented in Section 4. Table 1 shows the summary of the EFO parameters' ranges.

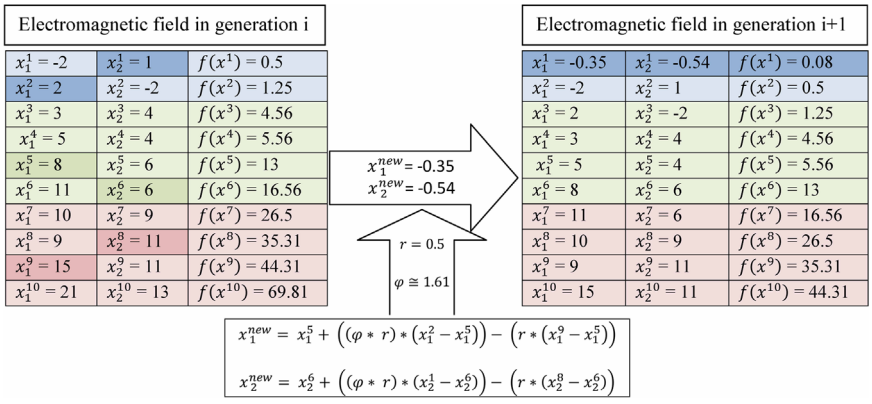

Fig. 3. New electromagnetic particle generation.

2.3. An example of optimization by EFO

Optimization of one highly multimodal function called the Rastrigin function is provided in this section to demonstrate the movement of electromagnetic particles towards global minima. In this experiment, we set the number of electromagnetic particles (N_emp) to 500. We use a large population in this experiment to illustrate the particles' movement clearly, while for optimization of this function, a population consisting of 50 particles is sufficient and a larger population only increases the computational time.

Table 1
EFO parameters setting.

Parameters	Lowest value	Highest value
N_emp	50	*D*
P_field	0.05	0.1
N_field	0.4	0.5
Ps_rate	0.1	0.4
R_rate	0.1	0.4

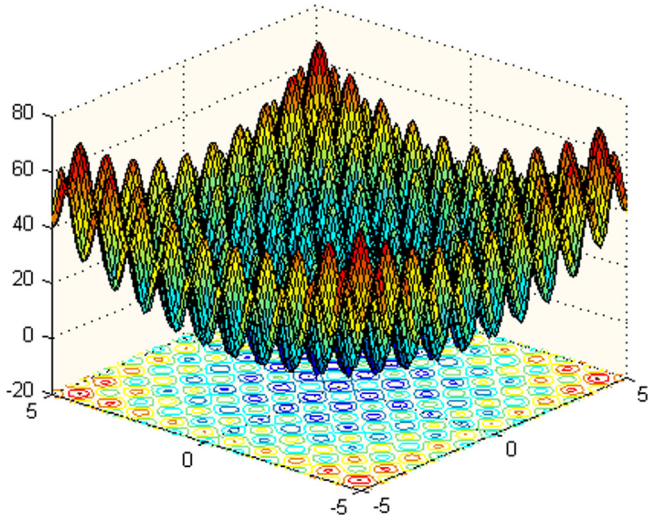

Fig. 4. 2D Rastrigin function.

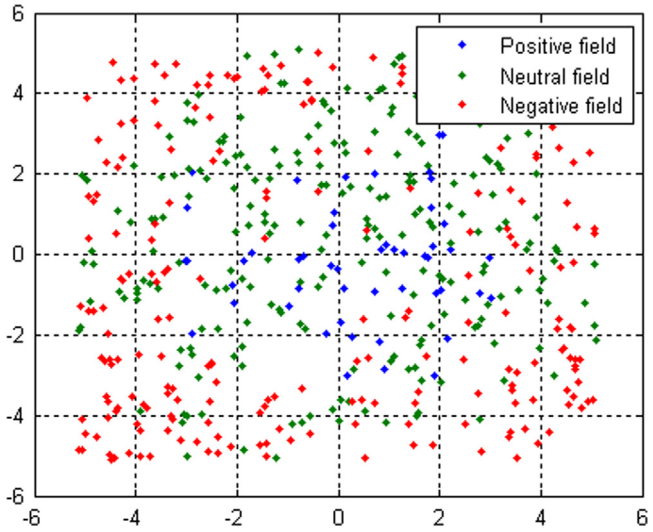

Fig. 5. Position of particles in iteration no. 1.

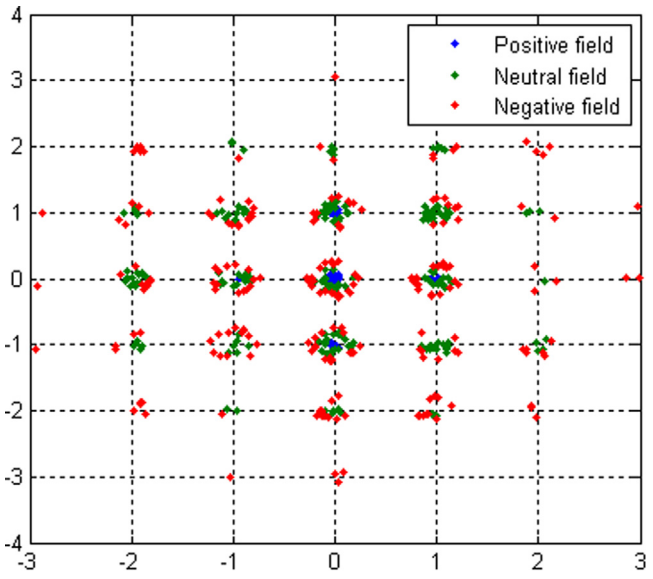

Fig. 6. Position of particles in iteration no. 2,000.

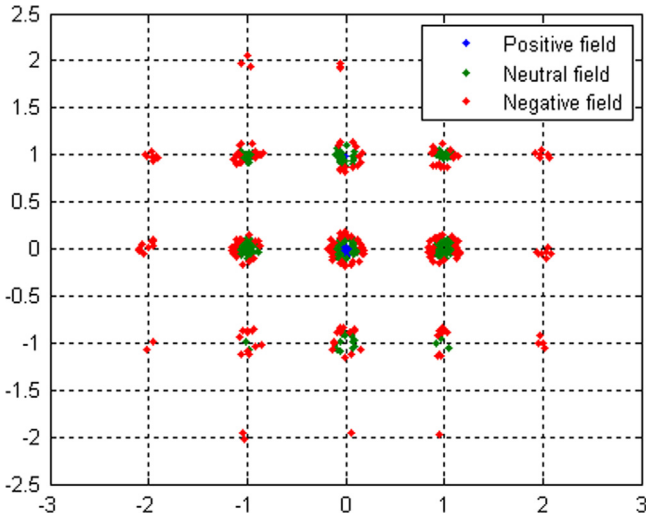

Fig. 7. Position of particles in iteration no. 3500.

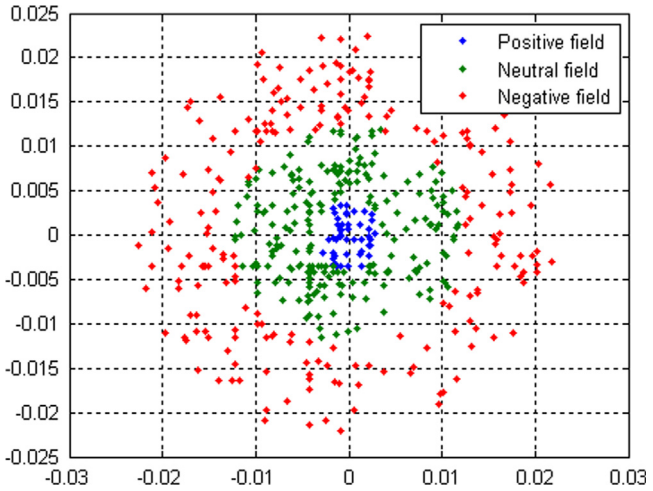

Fig. 8. Position of particles in iteration no. 5000.

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

Table 2
Parameter settings used in experiments.

Algorithm	P_field	N_field	Ps_rate	R_rate	C	W_0	W_1	N_o	N_e	N_s	Limit
EFO	0.1	0.45	0.2	0.3	–	–	–	–	–	–	–
ABC	–	–	–	–	–	–	–	50% of colony size	50% of colony size	1	N_e*D
CLPSO	–	–	–	–	1.49445	0.9	0.4	–	–	–	–

Table 3
Mean and standard deviation (± SD) of error for the CEC 2014 benchmark functions (dimension = 30).

Function	GSA	CLPSO	ABC	GSO	CoDE	EFO
1.	3.14E+08	1.20E+08	3.08E+07	2.20E+08	7.81E+06	**5.75E+05**
	(6.12E+07)	(3.16E+07)	(1.90E+07)	(5.38E+07)	(5.32E+06)	**(3.37E+05)**
2.	1.55E+10	3.15E+09	1.82E+04	1.44E+10	3.28E+07	**2.18E+02**
	(3.15E+09)	(7.96E+08)	(1.24E+04)	(2.10E+09)	(1.32E+07)	**(5.60E+02)**
3.	8.64E+04	4.04E+04	2.13E+03	9.96E+04	**2.20E+02**	2.06E+03
	(7.90E+03)	(1.16E+04)	(1.51E+03)	(1.76E+04)	**(7.34E+01)**	(2.11E+03)
4.	1.39E+03	4.96E+02	1.05E+02	1.84E+03	1.71E+02	**5.08E+01**
	(2.38E+02)	(7.33E+01)	(2.63E+01)	(2.51E+02)	(1.34E+01)	**(4.18E+01)**
5.	**2.00E+01**	2.10E+01	2.04E+01	2.04E+01	2.08E+01	**2.00E+01**
	(4.56E−04)	(7.07E−02)	(4.71E−02)	(8.21E−02)	(4.55E−02)	**(2.99E−02)**
6.	3.04E+01	2.85E+01	1.92E+01	1.26E+02	3.01E+01	**5.09E+00**
	(2.47E+00)	(1.47E+00)	(1.61E+00)	(6.15E+00)	(1.40E+00)	**(1.59E+00)**
7.	1.63E+02	2.92E+01	3.12E−01	9.62E+01	1.30E+00	**1.02E−02**
	(2.53E+01)	(5.31E+00)	(1.30E−01)	(1.92E+01)	(9.26E−02)	**(1.50E−02)**
8.	1.46E+02	1.62E+02	6.45E+00	4.17E+02	1.05E+02	**9.29E−01**
	(9.73E+00)	(1.27E+01)	1.59E+00	(3.38E+01)	(7.50E+00)	**(9.03E−01)**
9.	1.64E+02	2.51E+02	**1.25E+02**	8.09E+02	2.00E+02	1.55E+02
	(1.82E+01)	(1.22E+01)	**(2.12E+01)**	(6.48E+01)	(1.22E+01)	(4.71E+01)
10.	3.91E+03	4.26E+03	1.01E+02	7.66E+03	3.34E+03	**3.08E+00**
	(6.06E+02)	(2.91E+02)	(7.57E+01)	(1.01E+03)	(2.25E+02)	**(1.53E+00)**
11.	4.38E+03	6.30E+03	**2.90E+03**	1.76E+04	6.17E+03	7.25E+03
	(5.09E+02)	(4.23E+02)	**(2.51E+02)**	(1.39E+03)	(2.72E+02)	(2.53E+02)
12.	**2.03E−02**	2.47E+00	4.66E−01	1.14E+00	1.72E+00	2.99E+00
	(1.14E−02)	(3.52E−01)	(8.11E−02)	(2.12E−01)	(2.22E−01)	(3.85E−01)
13.	3.73E+00	6.79E−01	3.82E−01	5.54E−01	5.86E−01	**3.01E−01**
	(2.88E−01)	(8.40E−02)	(4.32E−02)	(4.85E−02)	(8.43E−02)	**(6.25E−02)**
14.	7.26E+01	5.14E+00	**2.39E−01**	5.59E+00	4.00E−01	3.75E−01
	(1.44E+01)	(2.71E+00)	**(2.98E−02)**	(1.01E+01)	(5.41E−02)	(1.33E−01)
15.	1.37E+03	8.27E+02	1.38E+01	2.98E+03	2.07E+01	**1.33E+01**
	(6.14E+02)	(4.27E+02)	(3.13E+00)	(1.75E+03)	(1.87E+00)	**(2.13E+00)**
16.	1.37E+01	1.29E+01	1.13E+01	4.48E+01	1.26E+01	**1.04E+01**
	(2.70E−01)	(2.28E−01)	(3.30E−01)	(8.60E−01)	(2.75E−01)	**(3.92E−01)**
17.	2.30E+07	4.31E+06	5.98E+06	4.50E+07	**1.64E+04**	2.31E+05
	(6.61E+06)	(1.72E+06)	(3.90E+06)	(1.46E+07)	**(9.04E+03)**	(1.72E+05)
18.	**5.36E+02**	1.12E+07	7.51E+03	8.16E+06	5.43E+03	3.07E+03
	(3.37E+02)	(6.16E+06)	(7.16E+03)	(1.06E+07)	(2.73E+03)	(2.80E+03)
19.	1.73E+02	4.87E+01	1.81E+01	2.32E+02	1.30E+01	**1.02E+01**
	(3.09E+01)	(1.61E+01)	(7.10E+00)	(4.93E+01)	(8.47E−01)	**(1.57E+01)**
20.	2.54E+05	2.06E+04	1.49E+04	1.26E+05	**1.13E+02**	8.31E+03
	(1.26E+05)	(9.05E+03)	(9.14E+03)	(3.31E+04)	**(1.98E+01)**	(4.62E+03)
21.	1.01E+07	7.15E+05	8.89E+05	1.88E+07	**2.67E+03**	1.41E+05
	(4.52E+06)	(3.96E+05)	(6.78E+05)	(6.27E+06)	**(4.83E+02)**	(9.40E+04)
22.	1.27E+03	4.98E+02	4.55E+02	3.05E+03	4.89E+02	**3.11E+02**
	(3.24E+02)	(1.27E+02)	(1.21E+02)	(4.81E+02)	(9.56E+01)	**(2.20E+02)**
23.	**2.66E+02**	3.37E+02	3.19E+02	4.26E+02	3.16E+02	3.15E+02
	(9.55E+01)	(4.88E+00)	(2.36E+00)	(1.53E+01)	(2.68E−01)	(9.01E−12)
24.	**2.15E+02**	2.70E+02	2.30E+02	4.03E+02	2.41E+02	2.30E+02
	(8.30E+00)	(3.52E+00)	(1.72E+00)	(8.92E+00)	(3.53E+00)	(5.26E+00)
25.	**2.05E+02**	2.22E+02	2.12E+02	3.35E+02	2.08E+02	**2.05E+02**
	(2.49E+00)	(2.80E+00)	(1.59E+00)	(2.00E+01)	(1.24E+00)	**(2.43E+00)**
26.	1.90E+02	1.01E+02	**1.00E+02**	2.05E+02	1.01E+02	1.17E+02
	(2.67E+01)	(5.94E−01)	**(5.90E−02)**	(1.55E+00)	(8.04E−02)	(4.88E+01)
27.	1.74E+03	5.00E+02	4.84E+02	3.55E+03	5.07E+02	**4.42E+02**
	(3.65E+02)	(3.04E+01)	(1.51E+01)	(1.85E+02)	(1.28E+02)	**(6.42E+01)**
28.	2.31E+03	1.61E+03	1.18E+03	1.99E+04	1.14E+03	**9.05E+02**
	(9.41E+02)	(1.80E+02)	(1.46E+02)	(3.25E+03)	(2.19E+01)	**(5.31E+01)**
29.	3.30E+07	9.25E+04	1.71E+03	8.13E+05	7.30E+03	**1.30E+03**
	(1.12E+08)	(3.55E+04)	(4.92E+02)	(6.96E+05)	(3.08E+03)	**(4.04E+02)**
30.	1.79E+06	3.02E+04	8.24E+03	7.73E+05	5.57E+03	**2.73E+03**
	(6.33E+05)	(8.06E+03)	(3.90E+03)	(2.26E+05)	(1.19E+03)	**(9.21E+02)**

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

Table 4
Mean and standard deviation (± SD) of error for the CEC 2014 benchmark functions (dimension = 50).

Function	GSA	CLPSO	ABC	GSO	CoDE	EFO
1.	4.37E+08	1.48E+08	2.81E+07	2.39E+07	1.21E+07	**1.52E+06**
	(2.51E+08)	(2.57E+07)	(1.01E+07)	(6.94E+06)	(4.48E+06)	**(5.46E+05)**
2.	3.04E+10	6.81E+09	2.88E+04	4.88E+07	1.89E+07	**6.83E+03**
	(4.95E+09)	(1.12E+09)	(4.11E+04)	(2.58E+07)	(9.45E+06)	**(7.04E+03)**
3.	1.52E+05	9.86E+04	1.06E+04	2.00E+04	**4.16E+03**	3.18E+04
	(7.53E+03)	(1.68E+04)	(3.66E+03)	(6.02E+03)	**(1.89E+03)**	(1.39E+04)
4.	4.36E+03	9.77E+02	1.59E+02	2.92E+02	1.44E+02	**9.44E+01**
	(8.38E+02)	(1.39E+02)	(2.76E+01)	(5.23E+01)	(1.55E+01)	**(2.91E+01)**
5.	**2.00E+01**	2.11E+01	2.04E+01	**2.00E+01**	2.10E+01	**2.00E+01**
	(8.83E−05)	(4.87E−02)	(3.53E−02)	**(2.90E−02)**	(6.56E−02)	**(2.67E−02)**
6.	5.57E+01	5.08E+01	3.78E+01	4.72E+01	5.57E+01	**1.60E+01**
	(3.12E+00)	(2.46E+00)	(2.65E+00)	(4.07E+00)	(2.67E+00)	**(2.31E+00)**
7.	2.85E+02	6.32E+01	5.72E−01	1.81E+00	1.20E+00	**2.66E−02**
	(4.55E+01)	(8.63E+00)	(1.36E−01)	(5.33E−01)	(7.20E−02)	**(5.17E−02)**
8.	2.81E+02	2.92E+02	1.26E+01	7.70E+01	2.30E+02	**1.37E+00**
	(1.96E+01)	(1.87E+01)	(1.74E+00)	(1.60E+01)	(1.45E+01)	**(1.42E+00)**
9.	3.64E+02	4.73E+02	2.58E+02	3.09E+02	3.80E+02	**1.91E+02**
	(3.12E+01)	(2.13E+01)	(2.83E+01)	(4.95E+01)	(1.89E+01)	**(1.41E+02)**
10.	7.68E+03	7.62E+03	2.29E+02	1.01E+03	7.26E+03	**4.70E+00**
	(7.20E+02)	(5.19E+02)	(1.07E+02)	(3.73E+02)	(3.84E+02)	**(2.04E+00)**
11.	8.54E+03	1.14E+04	**5.74E+03**	6.79E+03	1.21E+04	1.36E+04
	(7.53E+02)	(5.09E+02)	**(3.27E+02)**	(9.11E+02)	(4.27E+02)	(6.24E+02)
12.	**1.32E−02**	2.67E+00	4.71E−01	6.03E−01	2.47E+00	3.94E+00
	(1.06E−02)	(3.28E−01)	(5.73E−02)	(1.91E−01)	(2.74E−01)	(3.38E−01)
13.	3.79E+00	7.61E−01	4.51E−01	5.59E−01	6.53E−01	**3.51E−01**
	(2.45E−01)	(8.47E−02)	(4.11E−02)	(6.64E−02)	(6.56E−02)	**(6.35E−02)**
14.	6.16E+01	1.60E+01	**2.98E−01**	3.09E−01	4.31E−01	4.20E−01
	(1.46E+01)	(3.71E+00)	**(2.50E−02)**	(2.80E−02)	(8.50E−02)	(2.15E−01)
15.	2.07E+04	3.31E+03	3.14E+01	1.47E+02	3.78E+01	**2.47E+01**
	(1.16E+04)	(1.93E+03)	(6.02E+00)	(3.90E+01)	(2.26E+00)	**(7.71E+00)**
16.	2.26E+01	2.24E+01	**1.97E+01**	2.28E+01	2.28E+01	2.22E+01
	(4.78E−01)	(2.33E−01)	**(4.02E−01)**	(3.05E−01)	(3.26E−01)	(3.40E−01)
17.	3.49E+07	1.77E+07	1.01E+07	4.00E+06	**1.81E+05**	2.84E+05
	(1.48E+07)	(4.74E+06)	(4.96E+06)	(1.42E+06)	**(1.24E+05)**	(1.05E+05)
18.	6.93E+08	2.51E+07	9.92E+03	1.28E+03	3.62E+03	**8.97E+02**
	(8.46E+08)	(8.22E+06)	(9.94E+03)	(9.26E+02)	(2.31E+03)	**(9.37E+02)**
19.	2.02E+02	9.23E+01	**3.33E+01**	5.37E+01	3.62E+01	5.84E+01
	(3.38E+01)	(1.19E+01)	**(1.06E+01)**	(3.36E+01)	(1.08E+01)	(2.17E+01)
20.	1.07E+05	5.17E+04	3.96E+04	1.94E+04	**5.04E+02**	1.10E+04
	(4.22E+04)	(1.06E+04)	(1.29E+04)	(1.11E+04)	**(3.17E+02)**	(6.66E+03)
21.	4.19E+06	5.71E+06	7.30E+06	3.36E+06	**2.12E+04**	2.72E+05
	(1.42E+06)	(2.15E+06)	(4.36E+06)	(1.65E+06)	**(1.61E+04)**	(1.51E+05)
22.	2.45E+03	1.36E+03	1.14E+03	1.48E+03	1.44E+03	**1.12E+03**
	(7.30E+02)	(1.71E+02)	(1.89E+02)	(3.27E+02)	(1.59E+02)	**(2.86E+02)**
23.	**2.14E+02**	3.90E+02	3.57E+02	3.54E+02	3.55E+02	3.44E+02
	(7.68E+01)	(8.19E+00)	(7.30E+00)	(1.97E+00)	(1.77E−01)	(3.25E−10)
24.	**2.47E+02**	3.39E+02	2.71E+02	2.70E+02	2.83E+02	2.69E+02
	(1.12E+01)	(6.72E+00)	(1.78E+00)	(7.39E+00)	(1.80E+00)	(6.14E+00)
25.	**2.03E+02**	2.46E+02	2.22E+02	2.48E+02	2.18E+02	2.18E+02
	(4.02E+00)	(5.30E+00)	(2.80E+00)	(8.18E+00)	(1.94E+00)	(3.58E+00)
26.	2.00E+02	1.10E+02	**1.01E+02**	1.94E+02	1.04E+02	1.62E+02
	(8.47E−02)	(2.83E+01)	**(6.75E−02)**	(2.55E+01)	(1.82E+01)	(5.97E+01)
27.	3.29E+03	1.33E+03	1.08E+03	1.63E+03	1.28E+03	**7.98E+02**
	(7.49E+02)	(3.66E+02)	(3.78E+02)	(9.43E+01)	(1.47E+02)	**(6.71E+01)**
28.	5.71E+03	3.02E+03	2.15E+03	7.04E+03	1.92E+03	**1.55E+03**
	(1.11E+03)	(4.20E+02)	(3.42E+02)	(1.17E+03)	(1.26E+02)	**(2.22E+02)**
29.	**2.00E+02**	4.10E+05	3.32E+03	5.43E+03	2.00E+04	1.91E+03
	(7.50E−02)	(1.64E+05)	(1.46E+03)	(2.32E+03)	(7.15E+03)	(5.37E+02)
30.	4.97E+06	6.00E+04	1.61E+04	5.17E+04	1.97E+04	**1.24E+04**
	(5.07E+06)	(1.52E+04)	(4.10E+03)	(1.36E+04)	(2.00E+03)	**(2.45E+03)**

B. Rastrigin function, defined as

$$minf(x) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10)$$

where global optimum $x^* = 0$ and $f(x^*) = 0$ for $-5.12 \le x_i \le 5.12$ (Fig. 2).

As illustrated in Fig. 4, the Rastrigin function has several local minima. We ran EFO on this function for 5000 iterations, captured the position of the electromagnetic particles after iteration no. 1, 2000, 3500, and 5000, and illustrated the particles' movement towards global minima (Figs. 5–8). In this experiment, the EFO parameters are set as follows:

- N_emp500
- N_var2
- P_field0.1
- N_field0.45
- Ps_rate0.3
- R_rate0.2

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

The movements of the electromagnetic particles from shallow local minima towards deep local minima and then the global minimum are illustrated in Figs. 5–8. As shown in Fig. 8, all particles migrated to the global minimum region, to intensify the search for a near-optimal solution.

2.4. Computational time complexity of EFO

To show the complexity of a given algorithm, the running time is stated as a function relating the input length to the number of steps. The Big-O notation is mostly used to serve this purpose.

To show the complexity of EFO, we assume that (n) is the number of electromagnetic particles and (d) is the dimension of the optimization problem. If (m) is the maximum number of fitness function evacuation, the complexity of each part of Algorithm 1 is calculated as in Eqs. (5)–(7) and the overall complexity is calculated as in Eq. (8). From Eq. (8) we can conclude that EFO algorithm has a polynomial time complexity.

$$\text{part 1} : O(1) \tag{5}$$

$$\text{part 2} : O(n*d)+O(n* \log(n)) \tag{6}$$

$$\text{part 3} : O(m*n*d)+O(m*n) \tag{7}$$

Table 5
Summary of Wilcoxon's rank sum at 5% significant level.

Vs. EFO		30 Dimension	50 Dimension
GSA	+(better)	5	6
	−(worse)	23	23
	≅(no sig.)	2	1
CLPSO	+(better)	2	2
	−(worse)	26	27
	≅(no sig.)	2	1
ABC	+(better)	4	7
	−(worse)	23	21
	≅(no sig.)	3	2
GSO	+(better)	1	5
	−(worse)	28	21
	≅(no sig.)	1	4
CoDE	+(better)	6	8
	−(worse)	22	17
	≅(no sig.)	2	5

Table 6
Mean Friedman ranks of error for the CEC 2014 benchmark functions (dimension=30).

Function	GSA	CLPSO	ABC	GSO	CoDE	EFO
1.	5.867	4.033	2.967	5.100	2.033	1.000
2.	5.633	4.000	2.000	5.367	3.000	1.000
3.	5.200	4.000	2.500	5.800	1.233	2.267
4.	5.133	4.000	1.933	5.867	3.000	1.067
5.	1.000	5.300	5.667	2.567	4.033	2.433
6.	4.333	3.500	2.00	6.000	4.167	1.000
7.	6.000	4.000	1.967	5.000	3.000	1.033
8.	4.267	4.733	2.000	6.000	3.000	1.000
9.	2.367	5.000	1.800	6.000	3.833	2.000
10.	4.133	4.700	2.500	5.500	3.167	1.000
11.	2.000	3.883	1.500	6.000	3.617	4.000
12.	1.000	5.833	2.000	3.067	4.000	5.100
13.	6.000	4.700	2.067	3.300	3.933	1.000
14.	6.000	4.733	1.301	2.733	3.367	2.866
15.	5.000	4.200	1.267	5.800	2.967	1.766
16.	4.967	3.733	2.467	6.000	2.800	1.033
17.	5.067	3.400	4.400	5.643	1.200	1.290
18.	1.233	5.700	3.544	5.010	3.613	1.900
19.	5.100	3.933	2.667	5.900	2.200	1.200
20.	5.800	3.600	3.067	5.200	1.000	2.333
21.	5.033	3.400	3.500	5.967	1.000	2.100
22.	5.000	3.000	2.500	6.000	2.667	1.833
23.	2.400	4.667	3.600	5.933	2.733	1.667
24.	1.000	5.000	2.600	6.000	3.833	2.567
25.	1.933	5.000	3.867	6.000	2.600	1.600
26.	4.667	3.633	1.700	6.000	2.400	2.600
27.	5.000	3.233	2.833	6.000	2.367	1.567
28.	4.500	4.133	2.800	6.000	2.500	1.067
29.	2.133	4.933	2.668	5.866	3.900	1.500
30.	5.900	4.000	2.734	5.100	2.233	1.033
Sum	123.666	127.98	78.416	160.72	85.396	53.822

Table 7
Mean Friedman ranks of error for the CEC 2014 benchmark functions (dimension=50).

Function	GSA	CLPSO	ABC	GSO	CoDE	EFO
1.	6.000	5.000	3.633	3.167	2.200	1.000
2.	6.000	5.000	1.800	3.833	3.167	1.200
3.	6.000	5.000	2.100	3.133	1.100	3.667
4.	6.000	5.000	2.700	4.000	2.133	1.167
5.	2.000	5.167	5.767	3.000	4.066	1.000
6.	5.233	4.033	2.034	3.300	5.400	1.000
7.	6.000	5.000	2.000	3.967	3.033	1.000
8.	5.333	5.634	2.000	3.000	4.033	1.000
9.	3.867	6.000	2.000	2.867	4.533	1.733
10.	5.133	5.334	2.000	3.000	4.533	1.000
11.	3.233	4.200	1.067	2.500	4.867	5.133
12.	1.000	4.733	2.667	2.733	4.667	5.200
13.	6.000	4.633	2.767	2.667	3.933	1.000
14.	6.000	5.001	1.533	1.833	3.433	3.200
15.	5.967	5.032	1.967	4.000	2.467	1.567
16.	5.267	4.767	1.000	4.000	3.933	2.033
17.	5.800	5.000	4.133	3.067	1.333	1.667
18.	5.133	5.333	3.901	1.931	3.400	1.302
19.	6.000	4.767	2.000	2.967	2.234	3.032
20.	6.000	4.700	4.066	2.867	1.000	2.367
21.	4.167	5.133	5.067	3.633	1.000	2.000
22.	5.967	3.033	3.400	3.467	3.533	1.600
23.	1.167	5.967	4.567	4.367	2.967	1.965
24.	1.033	6.000	3.000	3.100	4.933	2.934
25.	1.000	5.400	3.733	5.600	2.667	2.600
26.	4.700	3.633	1.300	5.567	2.367	3.433
27.	5.900	3.400	2.850	4.800	3.000	1.050
28.	5.133	4.000	2.700	5.867	2.200	1.100
29.	1.000	6.000	3.000	3.833	4.967	2.200
30.	4.333	5.033	2.501	4.633	2.967	1.533
Sum	136.366	146.933	83.253	106.699	96.066	60.683

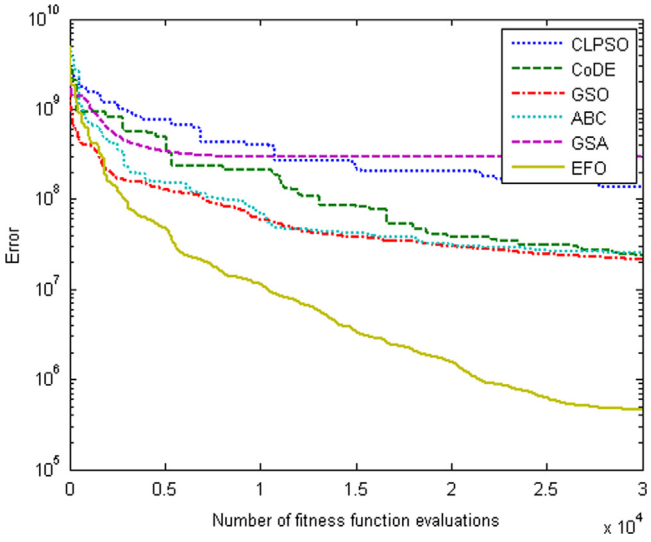

Fig. 9. Convergence curves of shifted and rotated high conditioned elliptic function.

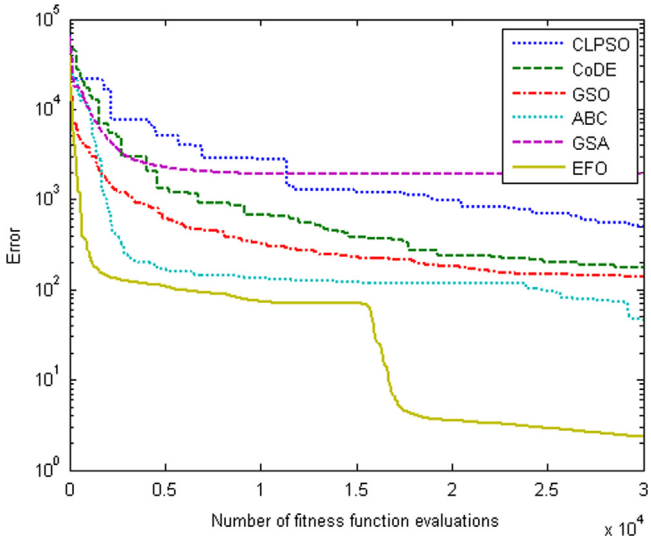

Fig. 10. Convergence curves of shifted and rotated Rosenbrock's function.

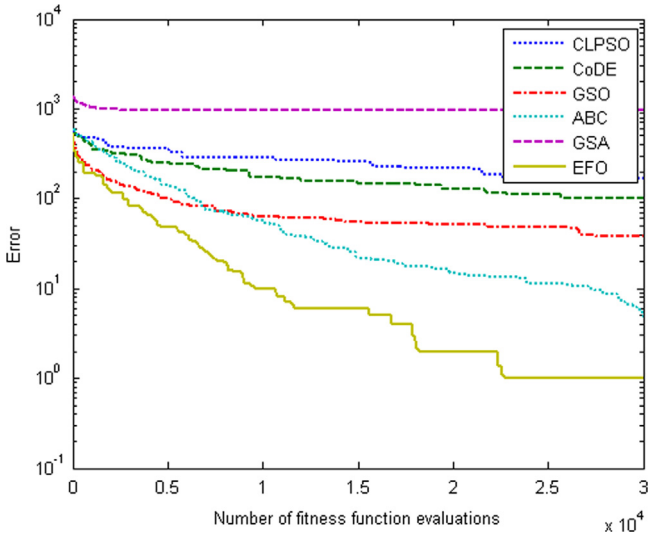

Fig. 11. Convergence curves of shifted Rastrigin's function.

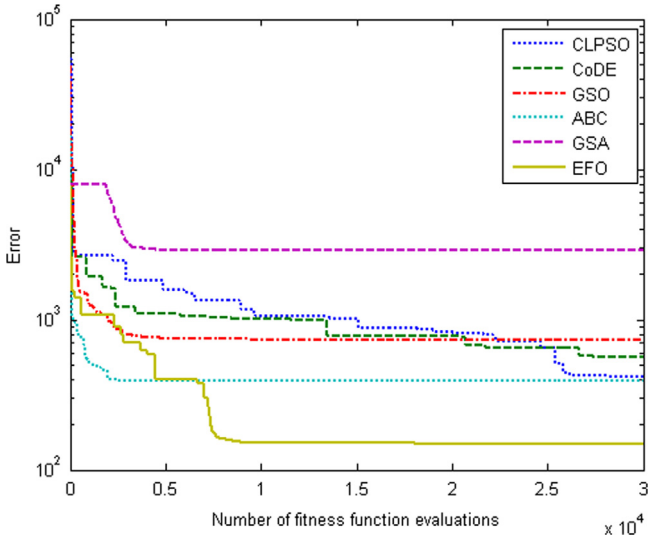

Fig. 12. Convergence curves of hybrid function 6 (CEC 2014).

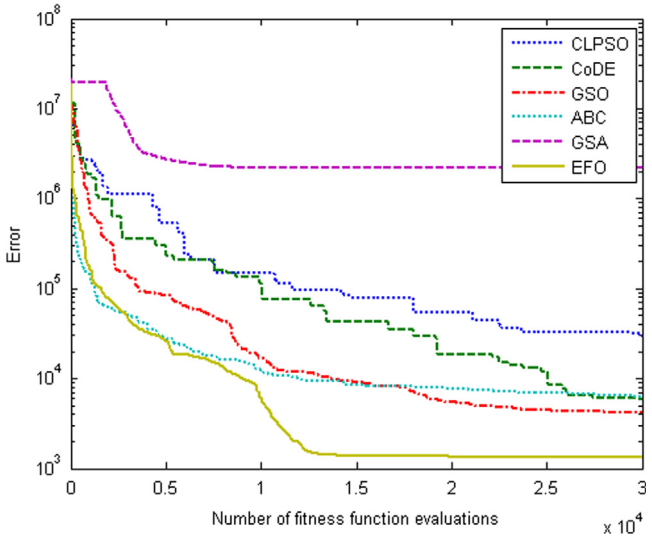

Fig. 13. Convergence curves of composition function 8 (CEC 2014).

overall complexity : $O(1)+O((m+1)*n*d)+O(n*\log(n))+O(m*n)$ (8)

3. Computational evaluation

In this section, a comprehensive evaluation of EFO is presented to illustrate the computational power of our algorithm.

3.1. Experimental setup and results

For the evaluation of EFO, a comprehensive experimental evaluation and comparison with five well-known optimization algorithms (ABC, GSO, GSA, CLPSO and CoDE) is provided based on 30 benchmarks of the CEC 2014 competition. Among the compared algorithms, ABC [16,26], GSO [18], and GSA[20,27] are recently published algorithms while CLPSO [9] and CoDE [28] are competitive variants of PSO and DE.

During our experiments, we kept the population size of all algorithms equal to 50, and other parameters of the compared algorithms have been set as suggested by authors. GSO, GSA and CoDE are parameter-less algorithms, and the parameter settings of the other algorithms are provided in Table 2.

Comparison between EFO and the other optimization algorithms is performed based on 30 and 50 dimensional versions of the CEC 2014 benchmark functions. The maximum number of fitness evaluations is set to $D*10^3$. The mean and standard deviation of the optimization error (best-optimum) of 30 independent runs of each algorithm are presented in Tables 3 and 4.

To analyze the obtained results statistically, two non-parametric statistical tests called Wilcoxon's rank sum test [29–33] and Friedman test [29] are conducted at the 5% significant level. The statistical testing of the algorithms based on Wilcoxon's rank sum test is reported in Table 5. The symbols $(+, -, \cong)$ indicate that a given algorithm performed significantly better $(+)$, significantly worse $(-)$, or not significantly different $(\cong)$ compared to EFO.

As illustrated in Table 5, EFO outperforms all the compared algorithms and finds better solutions for most of the CEC 2014 problems over both dimensions. For instance, EFO outperforms the most competitive algorithm among the compared algorithms (CoDE) in optimization of 73% of the 30D problems, while for the remaining problems, EFO performs either the same or slightly worse than CoDE.

As there are more than two algorithms for comparison, it is necessary to compare them based on Friedman test to avoid

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

Table 8
CEC 2014 subset used for convergence comparison. D: dimension, C: characteristic, U: unimodal, M: multimodal, S: separable, N: non-separable.

Function NO.in CEC 2014	Function name	D	C	Range
1.	Shifted and rotated high conditioned elliptic function	30	UN	$-100 \leq x_i \leq 100$
4.	Shifted and rotated Rosenbrock's function	30	MN	$-100 \leq x_i \leq 100$
8.	Shifted Rastrigin's function	30	MS	$-100 \leq x_i \leq 100$
22.	Hybrid function 6	30	MN	$-100 \leq x_i \leq 100$
30.	Composition function 8	30	MN	$-100 \leq x_i \leq 100$

Table 9
Variation of mean and standard deviation (± SD) of error with variation of N_emp (dimension = 30).

Function NO. in CEC 2014/N_emp	30	50	100	150
1.	5.63E+05 (2.82E+05)	**5.61E+05 (3.43E+05)**	3.75E+06 (1.27E+06)	1.15E+07 (5.64E+06)
4.	6.83E+01 (4.20E+01)	**4.63E+01 (3.24E+01)**	5.67E+01 (4.32E+01)	8.93E+01 (3.77E+01)
8.	6.57E−01 (7.47E−01)	**4.97E−01 (5.24E−01)**	6.52E+00 (2.23E+00)	2.55E+01 (2.94E+00)
22.	**3.12E+02 (1.99E+02)**	4.42E+02 (2.24E+02)	4.34E+02 (2.18E+02)	6.69E+02 (1.49E+02)
30.	3.06E+03 (7.66E+02)	2.86E+03 (1.08E+03)	**2.44E+03 (7.99E+02)**	2.69E+03 (6.01E+02)

Table 10
Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for shifted and rotated high conditioned elliptic function (dimension = 30).

P_field/N_field	0.3	0.4	0.5	0.6
0.05	6.81E+05 (4.19E+05)	4.67E+05 (2.48E+05)	**3.17E+05 (1.86E+05)**	5.69E+05 (4.88E+05)
0.1	6.27E+05 (3.74E+05)	6.98E+05 (3.78E+05)	4.33E+05 (2.43E+05)	5.79E+05 (3.91E+05)
0.15	1.38E+06 (6.88E+05)	1.16E+06 (5.50E+05)	9.66E+05 (6.49E+05)	1.20E+06 (8.33E+05)
0.2	1.66E+06 (7.65E+05)	1.95E+06(8.38E+05)	1.27E+06(6.77E+05)	1.67E+06(1.11E+06)

Table 11
Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for shifted and rotated Rosenbrock's function (dimension = 30).

P_field/N_field	0.3	0.4	0.5	0.6
0.05	4.18E+01 (4.35E+01)	4.13E+01 (3.46E+01)	4.83E+01 (3.26E+01)	4.92E+01 (3.23E+01)
0.1	**3.13E+01 (4.74E+01)**	6.74E+01 (3.32E+01)	5.36E+01 (4.12E+01)	3.55E+01 (4.77E+01)
0.15	6.09E+01 (3.06E+01)	3.73E+01 (3.71E+01)	6.08E+01 (3.42E+01)	7.48E+01 (5.01E+01)
0.2	6.71E+01 (3.55E+01)	8.18E+01 (3.88E+01)	5.74E+01 (4.32E+01)	9.07E+01 (4.51E+01)

Table 12
Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for shifted Rastrigin's function (dimension = 30).

P_field/N_field	0.3	0.4	0.5	0.6
0.05	5.97E−01 (8.39E−01)	4.97E−01 (7.04E−01)	**3.98E−01 (5.14E−01)**	6.97E−01 (1.15E+00)
0.1	6.98E−01 (4.78E−01)	1.09E+00 (1.19E+00)	8.95E−01 (1.09E+00)	6.35E−01 (6.69E−01)
0.15	1.06E+00 (9.98E−01)	5.97E−01 (8.39E−01)	1.29E+00 (9.44E−01)	6.96E−01 (1.15E+00)
0.2	1.59E+00 (9.61E−01)	1.19E+00 (7.85E−01)	6.96E−01 (8.19E−01)	9.95E−01 (1.05E+00)

transitivity between the results. Tables 6 and 7 show the results of Friedman test ranks on both dimensions. As can be seen from Tables 6 and 7, EFO algorithm has lower summary rank compared to other algorithm. Therefore, we can conclude that EFO outperforms all of the compared algorithms in both dimensions.

3.2. Convergence comparison

As shown in Tables 5–7, EFO outperforms the other algorithms for optimization of the CEC 2014 problems. Convergence comparisons between EFO, CLPSO, ABC, GSO, GSA and CoDE for optimization of a subset of CEC 2014 functions are shown in Figs. 9–13. Due

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

to the large number of functions in the CEC 2014 problem set, we selected one function from each category to illustrate the convergence of the compared algorithms. The list of selected functions and their characteristics are presented in Table 8.

As we mentioned earlier, the electromagnetic particles in EFO are highly cooperative; this characteristic helps EFO to converge effectively by obtaining sufficient knowledge about the search

Table 13
Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for hybrid function 6 (dimension = 30)

P_field/N_field	0.3	0.4	0.5	0.6
0.05	2.62E+02 (1.00E+02)	**1.70E+02 (8.95E+01)**	3.58E+02 (1.13E+02)	3.40E+02 (2.25E+02)
0.1	5.63E+02 (3.37E+02)	2.89E+02 (2.60E+02)	3.41E+02 (2.56E+02)	4.34E+02 (2.49E+02)
0.15	3.77E+02 (2.30E+02)	4.55E+02 (2.73E+02)	3.52E+02 (1.80E+02)	2.23E+02 (2.11E+02)
0.2	3.60E+02 (2.30E+02)	5.16E+02 (2.40E+02)	4.60E+02 (2.76E+02)	4.50E+02 (2.68E+02)

Table 14
Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for composition function 8 (dimension = 30).

P_field/N_field	0.3	0.4	0.5	0.6
0.05	2.66E+03 (7.98E+02)	**2.50E+03 (6.18E+02)**	3.00E+03 (7.98E+02)	2.57E+03 (8.18E+02)
0.1	2.54E+03 (5.39E+02)	3.15E+03 (1.15E+03)	2.66E+03 (5.79E+02)	2.73E+03 (5.92E+02)
0.15	2.70E+03 (8.41E+02)	3.36E+03 (5.76E+02)	3.21E+03 (9.68E+02)	2.49E+03 (8.06E+02)
0.2	2.93E+03 (5.78E+02)	2.93E+03 (8.88E+02)	3.13E+03 (7.52E+02)	2.86E+03 (6.24E+02)

Table 15
Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for shifted and rotated high conditioned elliptic function (dimension = 30).

R_rate/Ps_rate	0.1	0.2	0.3	0.4
0.1	4.55E+05 (2.95E+05)	4.62E+05 (2.06E+05)	5.84E+05 (4.08E+05)	6.20E+05 (4.36E+05)
0.2	8. 16E+05 (5.07E+05)	6.53E+05 (3.50E+05)	5.49E+05 (2.43E+05)	4.46E+05 (2.72E+05)
0.3	7.06E+05 (3.34E+05)	5.36E+05 (2.25E+05)	**3.66E+05 (1.48E+05)**	6.42E+05 (3.38E+05)
0.4	6.53E+05 (4.11E+05)	8.52E+05 (5.26E+05)	5.52E+05 (4.25E+05)	9.67E+05 (4.67E+05)

Table 16
Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for shifted and rotated Rosenbrock's function (dimension = 30).

R_rate/Ps_rate	0.1	0.2	0.3	0.4
0.1	4.26E+01 (4.63E+01)	5.05E+01 (4.70E+01)	**2.37E+01 (3.46E+01)**	3.84E+01 (3.58E+01)
0.2	3.61E+01 (4.02E+01)	3.44E+01 (5.34E+01)	3.60E+01 (4.78E+01)	3.50E+01 (4.55E+01)
0.3	6.18E+01 (4.85E+01)	3.41E+01 (4.27E+01)	6.15E+01 (5.02E+01)	7.46E+01 (4.71E+01)
0.4	3.78E+01 (3.66E+01)	6.36E+01 (3.76E+01)	5.49E+01 (4.12E+01)	7.21E+01 (3.39E+01)

Table 17
Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for shifted Rastrigin's function (dimension = 30).

R_rate/Ps_rate	0.1	0.2	0.3	0.4
0.1	6.49E+00 (1.72E+00)	5.07E+00 (2.27E+00)	4.90E+00 (1.41E+00)	4.34E+00 (1.61E+00)
0.2	2.36E+00 (1.16E+00)	1.49E+00 (1.07E+00)	1.89E+00 (1.45E+00)	1.56E+00 (1.04E+00)
0.3	1.19E+00 (1.47E+00)	5.97E−01 (9.61E−01)	3.98E−01 (5.14E−01)	2.44E−01 (4.30E−01)
0.4	1.19E+00 (1.03E+00)	1.99E−01 (4.20E−01)	4.00E−01 (5.12E−01)	**1.16E−01 (3.10E−01)**

Table 18
Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for hybrid function 6 (dimension = 30).

R_rate/Ps_rate	0.1	0.2	0.3	0.4
0.1	4.08E+02 (1.96E+02)	3.51E+02 (2.92E+02)	3.54E+02 (1.91E+02)	4.15E+02 (1.44E+02)
0.2	4.42E+02 (2.52E+02)	**2.78E+02 (2.23E+02)**	3.79E+02 (2.18E+02)	4.14E+02 (2.08E+02)
0.3	4.49E+02 (2.56E+02)	3.43E+02 (2.18E+02)	3.93E+02 (2.22E+02)	4.23E+02 (1.87E+02)
0.4	4.27E+02 (2.24E+02)	2.94E+02	4.39E+02 (2.24E+02)	3.25E+02 (1.60E+02)

Table 19
Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for composition function 8 (dimension = 30).

R_rate/Ps_rate	0.1	0.2	0.3	0.4
0.1	2.39E+03 (2.39E+03)	2.54E+03 (2.54E+03)	3.19E+03 (3.19E+03)	3.47E+03 (3.47E+03)
0.2	2.71E+03 (2.71E+03)	2.96E+03 (2.96E+03)	3.07E+03 (3.07E+03)	2.72E+03 (2.72E+03)
0.3	3.14E+03 (3.14E+03)	2.96E+03 (2.96E+03)	2.81E+03 (2.81E+03)	2.89E+03 (2.89E+03)
0.4	**2.32E+03 (2.32E+03)**	2.88E+03 (2.88E+03)	2.89E+03 (2.89E+03)	2.56E+03 (2.56E+03)

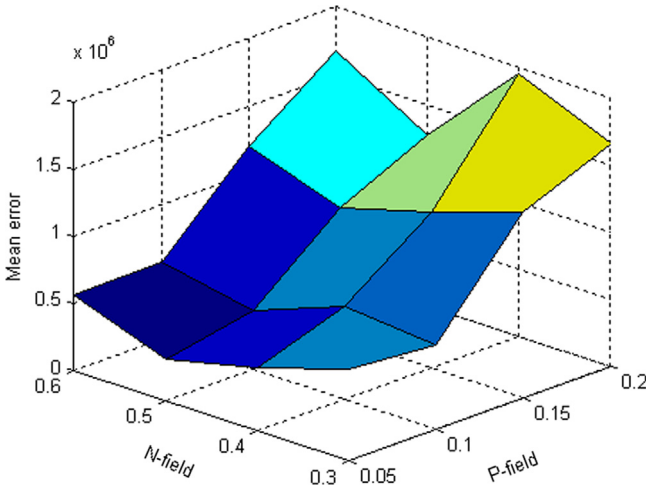

Fig. 14. Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for shifted and rotated high conditioned elliptic function (dimension = 30).

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

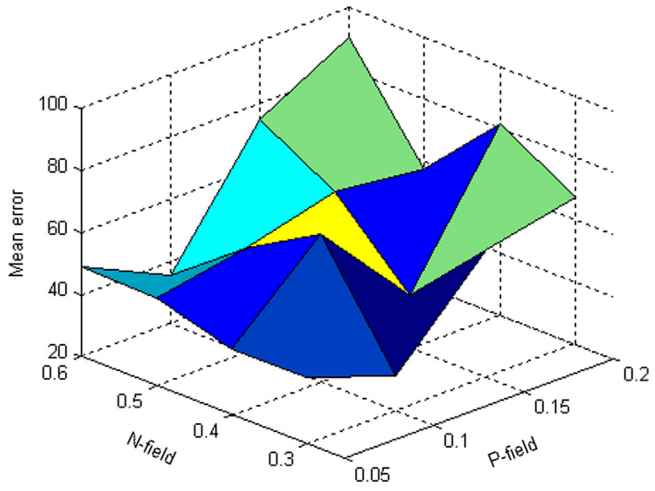

Fig. 15. Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for shifted and rotated Rosenbrock's function (dimension = 30).

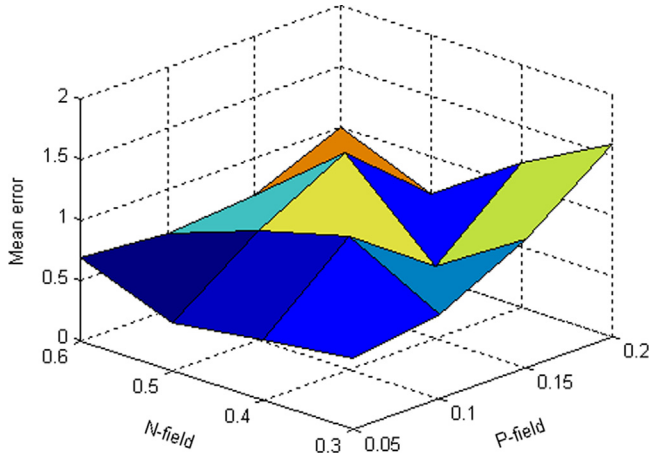

Fig. 16. Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for shifted Rastrigin's function (dimension = 30).

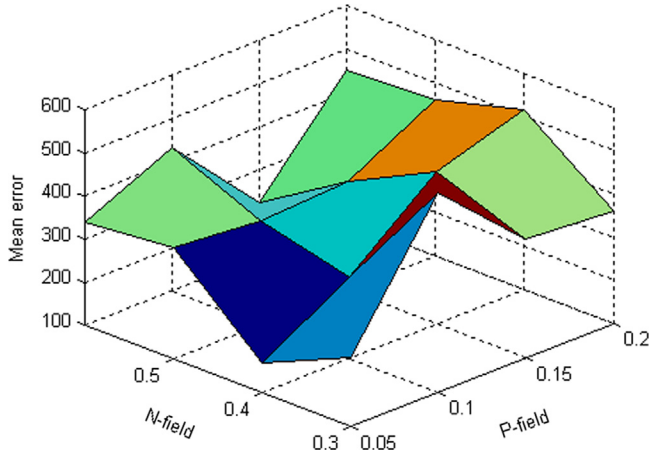

Fig. 17. Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for Hybrid Function 6 (dimension = 30).

space from a large subset of particles for the generation of new particles. Therefore, as seen in Figs. 9–13, EFO converges faster than the other compared algorithms in the optimization of all types of functions (unimodal, multimodal, separable, non-separable).

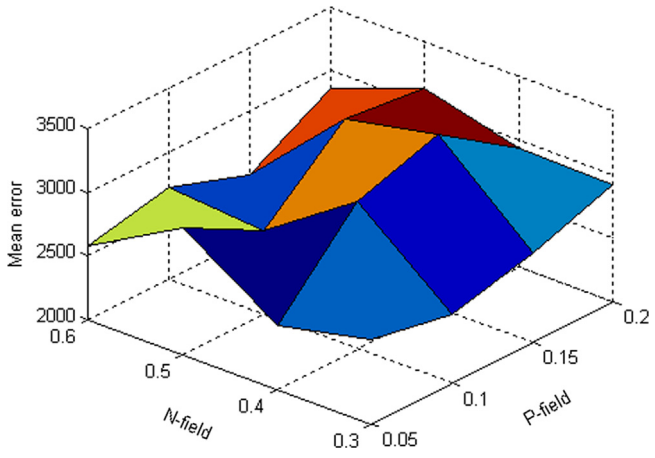

Fig. 18. Variation of mean and standard deviation (± SD) of error with variation of P_field and N_field values for composition function 8 (dimension = 30).

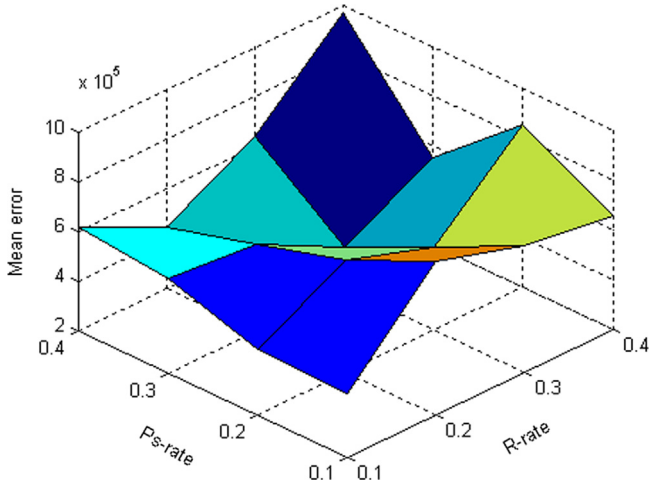

Fig. 19. Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for shifted and rotated high conditioned elliptic Function (dimension = 30).

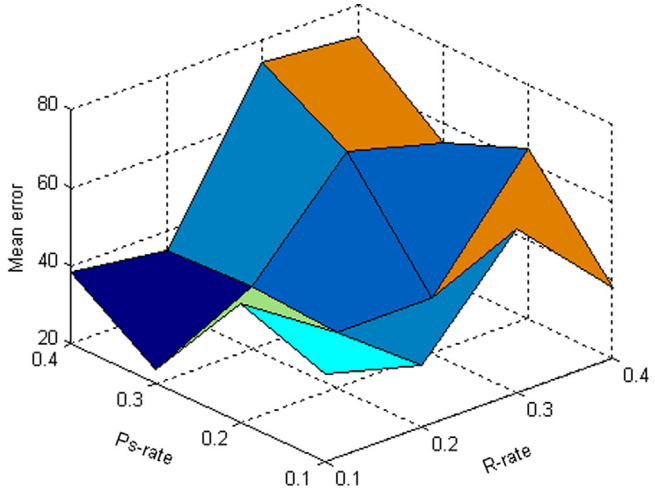

Fig. 20. Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for shifted and rotated Rosenbrock's Function (dimension = 30).

4. Impact of parameters on the performance of EFO

In Section 2.1, we briefly discussed the EFO parameters. In this section, we comprehensively study the impact of EFO parameters

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

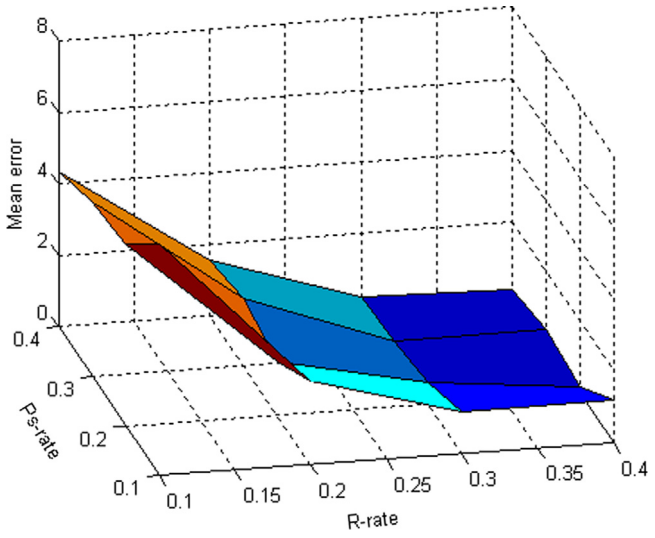

Fig. 21. Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for shifted Rastrigin's function (dimension = 30).

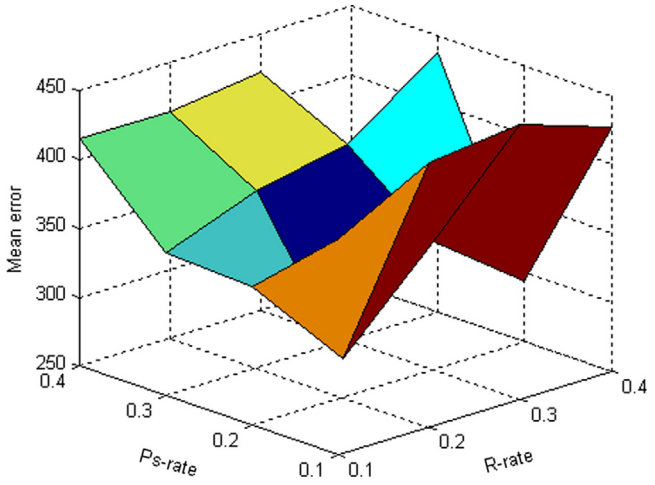

Fig. 22. Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for hybrid function 6 (dimension = 30).

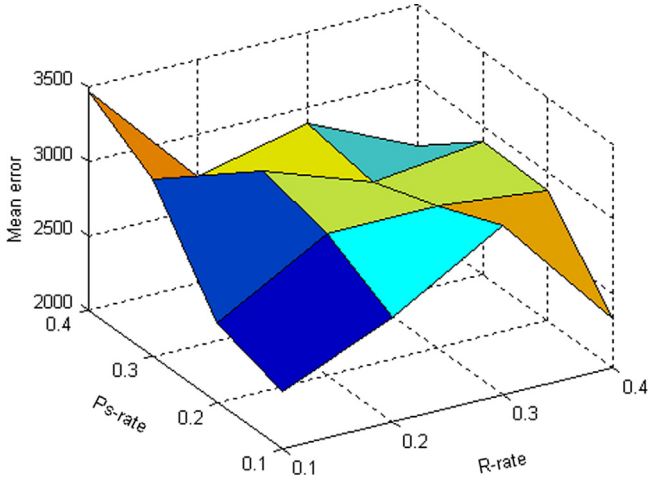

Fig. 23. Variation of mean and standard deviation (± SD) of error with variation of R_rate and Ps_rate values for composition function 8 (dimension = 30).

on the performance of the algorithm. For this experiment, we used the function set that was introduced in Table 8. Because the number of electromagnetic particles (N_emp) is the most important parameter of EFO, we first evaluated the impact of this parameter on the mean error of optimization. The results of this experiment are reported in Table 9.

As illustrated in Table 9, a population consisting of 50 electromagnetic particles outperforms greater and smaller populations. Therefore, in this experiment, we kept the population size constant (equal to 50) and studied the effects of variation in EFO parameters pairwise. The obtained results from the different parameter settings are summarized in Tables 10–19, and the corresponding surface charts are illustrated in Figs. 14–23.

In summary, a large population slows down the convergence while increasing the chances of avoiding local minima, and a small population tends to find local minima. Other important parameters of EFO are P_field and N_field; a large value for P_field increases the global search and slows down convergence, while a small value for P_field reduces the global search and increases the local search. On the other hand, the portion of the population allocated to the negative field should be equal or slightly higher than that of the neutral field to increase the global search power by repulsing electromagnets to different directions and helping them to avoid trapping in local minima. Therefore, Eq. (9) is suggested for calculation of the N_field value.

$$\text{N\_field} = \frac{1 - P\_\text{field}}{2} \tag{9}$$

The remaining parameters of EFO are Ps_rate and R_rate, High Ps_rate increases the convergence speed but reduces the global search power by selecting the existing electromagnets of the positive field for generation of new electromagnetic particles. Meanwhile, low Ps_rate slows down convergence and causes particles to bounce around the solution. On the other hand, low R_rate increases the convergence rate but reduces diversity in the population, which leads to finding local minima. While, high R_rate prevents particles from converging efficiently.

Therefore, a balance between global and local search is needed to lead the electromagnetic particles toward global minima. This balance can be achieved by the proposed parameter setting in Table 1.

5. Conclusions

This paper presented a physics-inspired and easy to implement algorithm for the optimization of continuous problems called EFO. This algorithm is inspired by the behavior of electromagnets and takes advantage of a nature-inspired ratio known as the golden ratio. Various experiments have been conducted for a better understanding of the algorithm behavior. Moreover, a comprehensive experimental investigation on the EFO parameters has been made, discussed and used to formulate an accurate and robust algorithm. Experimental results over 30 high dimensional CEC 2014 optimization functions prove that EFO generally outperforms other search-based approaches in terms of accuracy and convergence rate.

As future works, EFO should be applied to real-world problems to validate the applicability of this algorithm for real-world problems. Additionally, new search strategies and self-adaptive approaches can be proposed based on the introduced concepts in this paper to improve the search power and accuracy of EFO. Moreover, EFO works only on continuous problems. Therefore, a study can be conducted to adapt this algorithm for the optimization of discrete problems.

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002

Acknowledgements

The authors would like to thank the Research Management Centre (RMC), Universiti Teknologi Malaysia (UTM) for their support in R & D under the Special Flagship Project (Q.J130000.2428.02G50) and UTM *Big Data Centre* for the inspiration in making this study a success.

References

[1] I. Boussaïd, J. Lepagnot, P. Siarry, A survey on optimization metaheuristics, Inf. Sci. 237 (2013) 82–117.
[2] P.J. Van Laarhoven, E.H. Aarts, Simulated Annealing, Springer, 1987.
[3] F. Glover, Future paths for integer programming and links to artificial intelligence, Comput. Oper. Res. 13 (1986) 533–549.
[4] M.G. Omran, M. Mahdavi, Global-best harmony search, Appl. Math. Comput. 198 (2008) 643–656.
[5] M. Birattari, L. Paquete, T. Strutzle, K. Varrentrapp, Classification of Metaheuristics and Design of Experiments for the Analysis of Components Tech. Rep. AIDA-01-05, 2001.
[6] J.H. Holland, Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence, U Michigan Press, 1975.
[7] R. Rao, V. Savsani, D. Vakharia, Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems, Inf. Sci. 183 (2012) 1–15.
[8] J. Kennedy, The particle swarm: social adaptation of knowledge, in: IEEE International Conference on Evolutionary Computation, 1997, pp. 303–308.
[9] 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 (2006) 281–295.
[10] H. Wang, H. Sun, C. Li, S. Rahnamayan, J.-S. Pan, Diversity enhanced particle swarm optimization with neighborhood search, Inf. Sci. 223 (2013) 119–135.
[11] M. Tanweer, S. Suresh, N. Sundararajan, Self regulating particle swarm optimization algorithm, Inf. Sci. 294 (2015) 182–202.
[12] R. Storn, K. Price, Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces, ICSI Berkeley (1995).
[13] Z.W. Geem, J.H. Kim, G. Loganathan, A new heuristic optimization algorithm: harmony search, Simulation 76 (2001) 60–68.
[14] Z.W. Geem, State-of-the-art in the structure of harmony search algorithm, Springer (2010) 1–10.
[15] N. Tayarani, M. Akbarzadeh-T, Magnetic optimization algorithms a new synthesis, in: IEEE Congress on Evolutionary Computation, CEC 2008 (IEEE World Congress on Computational Intelligence), 2008, pp. 2659–2664.
[16] D. Karaboga, B. Basturk, A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm, J. Global Optim. 39 (2007) 459–471.
[17] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga, A comprehensive survey: artificial bee colony (ABC) algorithm and applications, Artif. Intell. Rev. 42 (2014) 21–57.
[18] S. He, Q.H. Wu, J. Saunders, Group search optimizer: an optimization algorithm inspired by animal searching behavior, IEEE Trans. Evol. Comput. 13 (2009) 973–990.
[19] M.-H. Tayarani-N, M.-R. Akbarzadeh-T, Magnetic-inspired optimization algorithms: operators and structures, Swarm Evol. Comput. 19 (2014) 82–101.
[20] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, GSA: a gravitational search algorithm, Inf. Sci. 179 (2009) 2232–2248.
[21] A. Kaveh, M.A.M. Share, M. Moslehi, Magnetic charged system search: a new meta-heuristic algorithm for optimization, Acta Mech. 224 (2013) 85–107.
[22] B. Javidy, A. Hatamlou, S. Mirjalili, Ions motion algorithm for solving optimization problems, Applied Soft Computing 32 (2015) 72–79.
[23] M. Livio, The Golden Ratio: The Story of Phi, the World's Most Astonishing Number, Random House LLC, 2008.
[24] M. Srinivas, L.M. Patnaik, Genetic algorithms: a survey, Computer 27 (1994) 17–26.
[25] Y. Wei, L. Qiqiang, Survey on particle swarm optimization algorithm, Eng. Sci. 5 (2004) 87–94.
[26] R. Akbari, R. Hedayatzadeh, K. Ziarati, B. Hassanizadeh, A multi-objective artificial bee colony algorithm, Swarm Evol. Comput. 2 (2012) 39–52.
[27] S. Yazdani, H. Nezamabadi-pour, S. Kamyab, A gravitational search algorithm for multimodal optimization, Swarm Evol. Comput. 14 (2014) 1–14.
[28] Y. Wang, Z. Cai, Q. Zhang, Differential evolution with composite trial vector generation strategies and control parameters, IEEE Trans. Evol. Comput. 15 (2011) 55–66.
[29] 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 Evol. Comput. 1 (2011) 3–18.
[30] R. Kumar, Directed bee colony optimization algorithm, Swarm Evol. Comput. 17 (2014) 60–73.
[31] S. Mirjalili, A. Lewis, S-shaped versus V-shaped transfer functions for binary particle swarm optimization, Swarm Evol. Comput. 9 (2013) 1–14.
[32] R. Tanabe, A.S. Fukunaga, Improving the search performance of SHADE using linear population size reduction, in: IEEE Congress on Evolutionary Computation (CEC), 2014, pp. 1658–1665.
[33] Z. Beheshti, S.M. Shamsuddin, Non-parametric particle swarm optimization for global optimization, Appl. Soft Comput. 28 (2015) 345–359.

Please cite this article as: H. Abedinpourshotorban, et al., Electromagnetic field optimization: A physics-inspired metaheuristic optimization algorithm, Swarm and Evolutionary Computation (2015), http://dx.doi.org/10.1016/j.swevo.2015.07.002