A Novel Modified Electromagnetism-like Algorithm for Solving Constrained Optimization Problem

Wenming Cao*, Jianwen Fang and Gan Huang

College of Electrical and Information Engineering, Qzhou University, Qzhou, Zhejiang, China

*Corresponding author e-mail: 357838461@qq.com

Abstract. The original Electromagnetism-like mechanism (EM) is a meta-heuristic algorithm utilizing an attraction-repulsion mechanism (called as force F) to move sample points towards optimality in global optimization problems. To solve the potential problem of stagnation happening in the original algorithm, new searching procedures were proposed in modified algorithm to determine the next search direction and step length of points in the paper. Compared to the original algorithm, tri-factors of the best historical visited positions of each point, best point and total force F were considered in the improved searching procedures. Moreover, the feasibility and dominance (FAD) rules were incorporated to extend the proposed algorithm to solve constrained optimization problem. Finally, Preliminary experiments verified the effectiveness of the proposed algorithm.

1. Introduction
As shown in following formula (1), the problem that is addressed in this paper considers finding a global optimal solution of a nonlinear constrain optimization problem without constrains violation.

\[ \min f(x) \]
\[ \text{s.t. } g_i(x) \leq 0, i = 1, \cdots, m, \quad h_j(x) = 0, j = 1, \cdots, l \]
\[ x \in \Omega \] (1)

2. Another section of your paper
Where \( f : \mathbb{R}^n \rightarrow \mathbb{R} \), \( g : \mathbb{R}^n \rightarrow \mathbb{R}^m \) and \( h : \mathbb{R}^n \rightarrow \mathbb{R}^m \) are nonlinear continuous functions and the closed set \( \Omega \) is defined by \( \Omega = \{ x \in \mathbb{R}^n : l \leq x \leq u \} \), where \( l \) and \( u \) are, respectively, the lower and upper bounds of \( x \). In real life, this class of global optimization problems with constrains is very important and frequently encountered in engineering applications. In the last decades, many algorithms have been proposed to solve problem (1). Probably the most extensively used in practice are stochastic-type algorithms. In this paper, we are interested in the electromagnetism-like (EM) algorithm proposed in [1,2]. This is a population-based algorithm that simulates the electromagnetism theory of physics by considering each point in the population as an electrical charge. The method uses an attraction-repulsion mechanism to move points towards the optimum.
Although the original EM has shown powerful capabilities of solving unconstrained optimization problem, it cannot be used directly to solve optimization problems with constraints like (1). Many modified versions of EM are proposed aimed to solve this problem. One common approach is to convert the problem (1) into an unconstrained problem via a penalty function [3]. However, the inherent difficulty is that the choice of penalty parameter in the penalty function-based methods. Other effective method is the use of the feasibility and dominance (FAD) rules for the constraints handling in constrained global optimization [4]. These rules can be easily incorporated into the EM algorithm. Based on this, EM can directly handle the constrained optimization problem without any changes. By introducing a new formula of charge calculation based on both the function value and the total constraint violations, the performance of the above hybrid EM algorithm combined with FAD rules can be improved further [5].

Many other revised versions have been developed to improve the original EM. The modifications in [6] are concerned with the charges associated with each point in the population, and a new formula for calculating charge of point is proposed. A local search based on the original pattern search method of Hooke and Jeeves and a shrinking strategy that aims to reduce the population size are proposed in [7]. A new modified EM algorithm is proposed in [8] using a linear combination of the total force exerted on a point, computed at the current iteration, with that of the previous iteration to define the force vector to move that point in the population. In [9], a hybrid IEM algorithm combining the advantages of the EM algorithm and the genetic algorithm is proposed for recurrent fuzzy neural controller design. Another hybrid approaches combining an electromagnetism-like method with a strong local search method, known as Solis and Wets and modified DFP, are proposed in [10,11]. A hybrid technique incorporating concepts of PSO and EM is proposed in [12], which creates individuals in a new generation not only by features of PSO, but also by attraction-repulsion mechanism of EM.

In this paper, a modified attraction-repulsion mechanism of EM combined with FED rules is presented. In the proposed algorithm, the best previously visited position of each point is added into the search process, and search direction and step length for point from one generation to next are synthetically determined by following factors: current point’s best previously visited position, the best particle in current swarm and total force F. Introduction of the best historical positions can help to improve the property of convergence. More importantly, this improved search mechanism can help solve the problem of stagnate of the original algorithm caused by too small values of the total force F, especially during the latter stage of iteration.

The rest of this paper is organized as follows. In Section 2, a review of the original EM and motivation of this paper are introduced. The general scheme of the proposed algorithm is demonstrated in Section 3. In Section 4, computational results on a set of test problems are presented. Some conclusions are given in Section 5.

3. A brief review of EM

3.1. The principle of EM

EM algorithm is a stochastic search method for global optimization. Similar to GA, population-based algorithm starts with randomly sampling points from the feasible region. According to the objective function values of these sample points, the regions of attraction are determined. Then a mechanism, similar to the attraction-repulsion mechanism of the electromagnetism theory, is invoked for further exploitation of these candidate regions. In EM, the charge of each point relates to the objective function value. This charge also determines the magnitude of attraction or repulsion of the point over the sample population – the better the objective function value, the higher the magnitude of attraction. The direction for each point to move in subsequent iterations can be determined by evaluating a combination force exerted on the point via other points. Like the electromagnetic forces, this force is calculated by adding the forces from each of the other points calculated separately. Then this attraction-repulsion mechanism encourages the points to converge to the optimal regions rapidly.
Take the following figure 1 for example. There are three particles and their own objective values are 15, 10 and 5, respectively. Because particle 1 is worse than particle 3 while particle 2 is better than particle 3, particle 1 represents a repulsion force which is the $F_{13}$ and particle 2 encourages particle 3 that moves to the neighborhood region of particle 2. Consequently, particle 3 moves along with the total force $F$. More details about EM can be available in [1].

![Figure 1. An example of attract-repulse effect](image)

3.2. Motivation

In original EM, all particles except current best particle will be updated and previous positions will not be reserved. Although it has been proved that EM can converge to the vicinity of global optimum with probability one [2], it does not guarantee that the newly generated particles are all better than the previous per iteration. This means that a lot of useful historical information conducive to find the optimal solution may lose during the search process. To solve this problem, the best previously visited positions of all particles will be recorded and applied in the revised method.

There is another issue we need to consider about EM. As discussed above, the EM utilizes an attraction-repulsion mechanism to move particles towards the optimality. The moving direction and step of particles are determined by evaluating a combination force $F$ exerted on the point via other points. Obviously, if force $F$ is too weak, the movement of particle will fall into stagnation. It isn’t conductive to the exploration of the optimality. Unfortunately, this situation is more likely to occur in the search process of EM, especially during the later stage. The problem of stagnation can be illustrated by using figure 2 below.

![Figure 2. An illustration of stagnation problem](image)

As shown in figure 2, most particles are concentrated around the position of global optimum. According to the work mechanism of EM, the situation showed in figure 2 often occurs during the latter stage of iteration [1]. Because positions of all particles are close to each other, the objective function values of the corresponding particles in figure 2 are also close. In this situation, it is most likely to happen that the combination forces $F$ exerted on the points are close to 0. Consequently, the particles...
will stagnate according to the search mechanism of EM. In order to avoid this problem, we will add an independent force exerted on the point from the current best point. More detailed discussions are available in 3.5.

4. The revised algorithm
The revised algorithm consists of five steps: preprocessing, initialization, local search, calculation of charge and total force vector and movement of point.

4.1. Preprocessing
Firstly, all equality constraints in problem (1) should be converted into inequality constraints. Then the problem (1) is rewritten as following formula:

\[ \text{min } f(x) \]
\[ \text{s.t. } c(x) \leq 0 \quad x \in \Omega \]

(2)

Where the vector of inequality constraints of problem is define by:

\[ c(x) = (g_1(x), \ldots, g_m(x), |h_1(x)| - \epsilon, \ldots, |h_l(x)| - \epsilon) \]

(3)

Where \( \epsilon \) is a smaller constant, for example, \( \epsilon = 0.001 \). The constraints violation is measured by \( l_2 \) norm of a vector, \( \| \pi(x) \|_2 \), where

\[ \pi_j(x) = \max \{0, c_j(x)\} \quad j = 1, \ldots, m, l \]

(4)

Obviously, if a point \( x^i \) makes \( \| \pi(x) \|_2 = 0 \), it means that this point meets all constraints of problem according to equation (3) and (4). Then it is a feasible point, whereas the point is infeasible. Therefore, the fitness of points can be determined by using following rules [4]:

Rule 1: If one point is infeasible and other is feasible, the feasible point is better;
Rule 2: If both points are infeasible, the one with lower constraints violation is better;
Rule 3: If both points are feasible, the one with lower function value is better.

The above rules will be used to determine the best point in the swarm, and the better one between two points.

4.2. Initialization
The procedure initialization is firstly used to sample \( m \) points, \( \{ x^1, \ldots, x^m \} \), randomly from the feasible domain of the variables, where \( x^i = [x^i_1, \ldots, x^i_n] \), \( i = 1, \ldots, m \), where \( n \) represents the dimension of problem. The procedure uniform sampling can be determined by following:

\[ x^i_k = l_k + (u_k - l_k). \text{rand}(1, n) \quad k = 1, \ldots, n \]

(5)

The procedure ends with \( m \) points identified, and the objective function value \( f(x^i) \) can be determined for each point. Compared to the original algorithm, the constraints violation \( \| \pi(x^i) \|_2 \) is required to be computed for each point in the population additionally. This pair of information \( (f(x^i), \| \pi(x^i) \|_2) \) defines the point fitness.
4.3. Local search
In the original EM, the local search procedure is used to gather local information and improve the current solutions. Application of the local search procedure might produce better results. Notice that the performance of the algorithm improves but at the cost of the number of function evaluations required by the Local procedures. Therefore, the combined local search algorithm should be as simple as possible. In this paper, a simple random line search is proposed. For example, a small random disturbance $\delta$ is added to the current point $x_i$, if the value of $f(x_i+\delta)$ is better than $f(x_i)$, then replacing original $x_i$ with $x_i+\delta$.

4.4. Calculating of charge and force vector
In the original EM, the charges of the points are calculated according to their objective function values, and the charge of each point is not constant and changes from iteration to iteration. In this way, the points that have better objective values possess higher charges. It is noted that it is not necessarily the best one for the point with better objective value in solving of constrained optimization problems. Consequently, a modified formula (6) used to calculate the charge is proposed.

$$q_i = \exp \left( -n \left[ \pi(x_i)^\frac{1}{n} + \sum_{k=1}^{m} \left| f(x_i) - f(x^{k\text{best}}) \right| \right] \right), \quad i = 1, 2, \cdots, m$$  \hspace{1cm} (6)

The charge of each point $i$, denoted by $q_i$, determines point $i$’s power of attraction or repulsion. In this way, the points that both have better objective values and lower constraints violation possess higher charges. From formula (6), the total force $F_i$ exerted on point $i$ is computed by the formula (7).

$$F_i = \begin{cases} \sum_{j \neq i} x^j - x^i - \frac{q^j q^i}{\| x^j - x^i \|}, & \text{if } x^j \text{ is better than } x^i \\ \sum_{j \neq i} x^j - x^i - \frac{q^j q^i}{\| x^j - x^i \|}, & \text{if } x^j \text{ is better than } x^i \\ \end{cases}, \quad \forall i$$  \hspace{1cm} (7)

According to formula (7), the direction of force between two points is decided after comparing their fitness. The point that has a better fitness attracts the other one. Contrarily, the point with worse fitness repels the other.

4.5. Moving points
After evaluating the total force vector $F_i$, the point $i$ is moved by a step given by following formula:

$$x^i_{j+1} = x^i_{j+1} + \omega_1 (p^i_{j+1} - x^i_{j+1}) + \omega_2 (g_j^{j+1} - x^i_{j+1}) + F^i_j$$  \hspace{1cm} (8)

Where, $p^i_j$ is the $j$-dimensional component of the best historical position of point $i$. $g_j$ is the $j$-dimensional component of the best point of the current the swarm. $\omega_1$ and $\omega_2$ are weighting factors, such that:

$$\omega_1 + \omega_2 = 1$$

The value of above weighting factors are randomly generated from $[0,1]$. Note that the new positions should be checked (and modified if necessary) to ensure the allowed feasible movement toward the upper bound, or the lower bound, that:
Finally, after updating all points, the termination criteria will be checked to determine to stop calculation or modify (if necessary) the $pi$ and $g$, and repeat the search process. The general scheme of the proposed algorithm can be given as following:

step 1. Preprocessing

step 2. Initialization

2.1. Sampling $m$ points randomly from the feasible domain according to (5), and calculating objective function value $f(x_i)$ and constraints violation $||\pi(x')||_2$.

2.2. Set $pi$ equal to be $xi$ obtained by step 2.1 (As there is not previously visited position in the first iteration).

2.3. Find the best point $g$ in the swarm.

step 3. Check the stop criteria, if the stop criterion is not satisfied, go to step 4.

step 4. Calculation of charge and total force $F$

4.1. Calculate the charge by formula (6).

4.2. Calculate the force $F$ by formula (7).

step 5. Movement of points

5.1. Update the positions of points by formula (8).

5.2. Check and modify the new positions by formula (9).

5.3. Recalculate the objective function and constraints violation, update $p$ and $g$, go to step 3.

Where $m$ is the number of sample points, and $xi$ is the coordinate of point $i$. $g$ and $pi$ is respectively the coordinate of best point in swarm and the best historical position of point $i$. The formula (8) plays a very important role in the process of searching optimal solution. According to formula (8), search direction and step length are synthetically determined by the best previously visited position of point $i$, the best point in current swarm and total force $F$. Let us consider again the question shown in figure 2. According to formula (8), all particles in figure 2 except the current best still can move encouraged by its best previous position and the best position of the current swarm even if the total force $F$ is equal to 0. Obviously, the new search mechanism can help solve the stagnate problem of the original algorithm caused by too small values of the total force $F$. Moreover, introduction of the best historical positions can help improve the property of convergence.

5. Computational experiments

In this section, different types of benchmark functions are used to certify the effectiveness and efficiency of proposed algorithm. To avoid attributing optimization results to choose of a particular initial population and to conduct fair comparisons, we demonstrate results for test functions in terms of the average number of function evaluations over 30 runs. The average and best objective function values are both reported. All the computations are conducted on an Intel(R) i5-4210M CPU 2.60GHz Pc. The algorithm is coded in Matlab R2012b.

5.1. Compared with FED-EM

In this case, we shall compare the proposed algorithm (denoted by FED-MEM) with FED-EM algorithm since both use the same Feasibility and dominance rules. The selected parameters are shown in Table 1, and $\epsilon$ is set to be 0.001. In order to facilitate the comparisons, as shown in Table 2, the same 13 general test functions are used in test.
The results in Table 3 show that the performances of the improved method are the same as or better than that of the FED-EM, except for function g05 and g13. It is known from table 1 that the swarm size and iteration number of the improved algorithm are both greatly less than the FED-EM. Meanwhile, the number of evaluations of local search drastically decreases for the proposed method with local search only applied to the current best point and performed one-time per iteration. Obviously, the proposed algorithm shows better performance in exploitation of better solutions.

However, function g05 and g13 with three equality constraints appear to be exception. Our parts of results are worse than that of the FED-EM. This might be because this type of algorithm is not good at solving the problem with much equality constrains. Hence, the effects of performance improvement are not obvious and the accuracy of the average function values is not good enough. According to equal (3), all equality constrains must be converted into un-equality constrains by subtracting a small instant $\varepsilon$. Only result of subtraction just equal to $\varepsilon$, the original equality constrain could be met. Unfortunately, the desired solution of meeting the conditions in practice is hard to be obtained. As shown in figure 3,
the distributions of constraints violation $\pi$ of 30 solutions for g05 are depicted. It can be seen that all values are not equal to 0. It demonstrates that not all equality constraints are met. Finally, an approximate but not optimum solution can be obtained. The same conclusion can be derived from convergence curve of constraint violation of g05, as shown in figure 4.

![Figure 3](image)

**Figure 3.** The distributions of constraints violation of 30 solutions for g05

![Figure 4](image)

**Figure 4.** A typical convergence curve of constraint violation in solving g05

Overall, we observe that the performance of the improved method is better than the FED-EM for solving the most general test functions.

5.2. Further test set

In this section, we further compare the performances among the proposed algorithm and other classical heuristic search algorithms such as [4] PSO (standard particle swarm optimization), PESO (particle evolutionary swarm optimization), TCPSO (Toscano and Coello’s PSO), SR (stochastic ranking), DE (differential evolution) and ABC (artificial bee colony). Amount of available test results of these algorithms on functions in table 2 can be directly accessed in paper [4]. We can use these results in comparing our method with above algorithms. The parameters setting in tests are shown in Table 4.

| Algorithm name | PSO | PESO | TCPSO | SR | DE | ABS |
|----------------|-----|------|-------|----|----|-----|
| Swarm size     | 50  | 50   | 50    | 50 | 40 | 40  |
| Maximum iteration | 350 000 | 350 000 | 350 000 | 350 000 | 240 0000 | 240 000 |

Table 5 shows the average results of testing on these algorithms. In order to facilitate comparison, we list results of comparison between above six algorithms and FED-EM, FED-MEM in table 6 and 7.
respectively, where “Y” and “N” respectively indicates FED-EM and FED-MEM with better or worse performance and, “E” denotes the same or similar test results can be obtained.

### Table 5. Average test Results of PSO, PESO, TCPSO, SR, DE, ABC

|       | $f_{global}$ | PSO     | PESO    | TCPSO   | SR      | DE      | ABC     |
|-------|--------------|---------|---------|---------|---------|---------|---------|
| g01   | -15.00       | -15.00  | -15.00  | -15.00  | -15.00  | -14.55  | -15.00  |
| g02   | 0.8036       | 0.419960| 0.721749| 0.790406| 0.78197 | 0.665   | 0.79241 |
| g03   | 1.0005       | 0.764813| 1.005006| 1.003814| 1.00000 | 1.0000  | 1.0000  |
| g04   | -30665.50    | -30665.53| -30665.50| -30665.53| -30665.53| -30665.53| -30665.53|
| g05   | 5126.4       | 5135.973| 5129.178| 5461.081| 5128.880| 5264.270| 5185.714|
| g06   | -6961.814    | -6961.814| -6961.810| -6875.94 | -     | -6961.81 | -     |
| g07   | 7049.2       | 7205.500| 7099.101| 7560.048| 7559.192| 7147.334| 7224.407|
| g08   | 0.0958       | 0.095825| 0.095825| 0.095825| 0.095825| 0.095825| 0.095825|
| g09   | 680.63       | 680.630 | 680.630 | 680.656 | 680.630 | 680.640 | 680.640 |
| g10   | 24.306       | 32.40727| 24.37125| 25.35577| 24.374  | 24.310  | 24.473  |
| g11   | 1.0000       | 0.998875| 1.000000| 1.000000| 1.000000| 1.000000| 1.000000|
| g12   | 0.0539       | 0.569358| 0.626881| 1.716426| 0.057006| 0.872   | 0.968   |
| g13   | 5126.4       | 5135.973| 5129.178| 5461.081| 5128.880| 5264.270| 5185.714|

Table 6. Results of comparison between FED-EM with PSO, PESO, TCPSO, SR, DE, ABC

| Function | PSO | PESO | TCPSO | SR | DE | ABC |
|----------|-----|------|-------|----|----|-----|
| g01      | N   | N    | N     | N  | N  | N   |
| g02      | N   | N    | N     | N  | N  | N   |
| g03      | Y   | N    | N     | Y  | N  | Y   |
| g04      | N   | N    | Y     | N  | N  | E   |
| g05      | E   | E    | Y     | N  | N  | Y   |
| g06      | N   | N    | N     | N  | N  | N   |
| g07      | Y   | N    | Y     | N  | N  | Y   |
| g08      | N   | N    | E     | N  | E  | E   |
| g09      | N   | N    | E     | N  | N  | E   |
| g10      | N   | N    | E     | N  | N  | E   |
| g11      | Y   | N    | Y     | N  | N  | E   |
| g12      | N   | N    | Y     | N  | N  | E   |
| g13      | E   | N    | E     | N  | N  | E   |

Table 7. Results of comparison between FED-MEM with PSO, PESO, TCPSO, SR, DE, ABC

| Function | PSO | PESO | TCPSO | SR | DE | ABS |
|----------|-----|------|-------|----|----|-----|
| g01      | N   | N    | E     | N  | E  | N   |
| g02      | N   | N    | E     | N  | E  | E   |
| g03      | E   | E    | N     | E  | E  | N   |
| g04      | Y   | N    | Y     | N  | N  | E   |
| g05      | E   | E    | N     | Y  | N  | E   |
| g06      | E   | E    | N     | Y  | N  | E   |
| g07      | E   | E    | N     | Y  | N  | E   |
| g08      | E   | E    | N     | Y  | N  | E   |
| g09      | E   | E    | N     | Y  | N  | E   |
| g10      | Y   | N    | Y     | N  | N  | E   |
| g11      | N   | N    | Y     | N  | N  | E   |
| g12      | E   | N    | E     | N  | N  | E   |
| g13      | E   | N    | E     | N  | N  | E   |

It is known from table 6 that the proportion of tests for FED-EM with better, equal and worse results are 8.97%, 7.69% and 83.3% respectively. These show that the convergence performance of FED-EM is worse for solving the most general test functions. By contrast, the proportion of tests for FED-MEM with better, equal and worse results are 8.97%, 51.28% and 39.74% respectively, as shown in table 7. It is noted from table 4 that the swarm size and iteration number of the proposed FED-MEM are both greatly less than those methods as a comparison. As the number of searching points and iteration increases, the test results of FED-MEM could be further developed. Obviously, the performance of the original FED-EM has been greatly improved based on the proposed modified EM.
5.3. Compared with other EM-based algorithms
In this section, the proposed method will be compared with other two constrained versions of EM which were developed for solving constrained problems.

| Function | PEM | CEM |
|----------|-----|-----|
|          | $f_{\text{best}}$ | $f_{\text{worst}}$ | $f_{\text{average}}$ | $f_{\text{best}}$ | $f_{\text{worst}}$ | $f_{\text{average}}$ |
| $g_01$   | -14.957990 | -11.028430 | -12.64385 | -15.000000 | -15.000000 | -15.000000 |
| $g_02$   | 0.516353  | 0.404182  | 0.453309 | 0.623711 | 0.452234 | 0.517221 |
| $g_03$   | 1.002150  | 0.995530  | 1.000080 | 1.001510 | 1.001670 | 1.001760 |
| $g_04$   | -30661.790 | -30627.758 | -30642.891 | -30665.513 | -30654.500 | -30660.649 |
| $g_05$   | 5126.4981  | 5384.7132  | 5191.5619 | 5126.4842 | 5136.6618 | 5128.6958 |
| $g_06$   | -6961.77   | -6961.71   | -6961.74 | -6961.813 | -6961.813 | -6961.813 |
| $g_07$   | 26.14955   | 32.74971   | 28.76543 | 25.11276 | 29.93511 | 27.75496 |
| $g_08$   | 0.095825   | 0.095825   | 0.095825 | 0.095825 | 0.095825 | 0.095825 |
| $g_09$   | 681.2211   | 685.1694   | 682.6084 | 680.8968 | 681.7960 | 681.3511 |
| $g_{10}$ | 7100.8486  | 7434.9652  | 7226.7802 | 7049.7581 | 7292.7241 | 7154.6709 |
| $g_{11}$ | 0.74900    | 0.74900    | 0.74900 | 0.7499 | 0.7499 | 0.7499 |
| $g_{12}$ | 1.000000   | 1.000000   | 1.000000 | 1.000000 | 1.000000 | 1.000000 |
| $g_{13}$ | 0.346621   | 2.134983   | 0.911530 | 0.053827 | 0.059852 | 0.056314 |

One version denoted by PEM incorporates penalty functions in the original EM [3]. Other version was proposed by using modified charge calculation of a point based on both the function value and the total constraint violations [5]. We denote this version by CEM. These algorithms use 30 independent runs and the maximum number of iterations 35000 is set as the termination criterion. Results from PEM and CEM are presented in Table 8. From Table 3 and 8 may conclude that FED-MEM is better than PEM except $g_{05}$, $g_{10}$ and $g_{13}$. Most importantly, the major advantage of our method is that it is not penalty function-based method without the difficulty with the choice of the penalty parameter value. However, the results seem to indicate that the proposed method is worse than CEM in solving most bench functions. Especially, the later show strong capabilities of exploration for the unknown variable space of $g_{05}$ and $g_{13}$. Aim to reduce the average number of evaluations, FED-MEM use fewer iteration and simpler procedure of local search than CEM. From this point of view, the performances of two algorithms should be equally matched.

6. Conclusion
This paper proposes a novel heuristic algorithm derived from electromagnetism-like mechanism for solving global optimization problems with constraints. Compared to other EM-based methods, the search mechanism of the proposed method uses the best historical visited positions of each point. Moreover, the next search direction and step length of points are determined together by its best previously visited position, the best point in current swarm and the total force. The preliminary experiments show that our method is able to show strong capabilities of exploration for the unknown variable space with less cost.

Acknowledgments
This work was supported by the projects of the Natural Science Foundation of Zhejiang Province (No.LY16F030013, No.LY15F020041) and Quzhou University of Zhejiang Province (No. BSYJ201506) and Quzhou Science and Technology Bureau (No.2015007).
References

[1] S. I. Birbil and S. C. Fang, “An Electromagnetism-like Mechanism for Global Optimization”. Journal of Global Optimization, vol.23, no.3, (2003), pp.263-282.

[2] S. I. Birbil, S. C. Fang and R. L. Sheu, “On the Convergence of a Population-based Global Optimization Algorithm”, Journal of global optimization, vol.30, no.2, (2004), pp.301-318.

[3] S. I. Birbil, “Stochastic Global Optimization Techniques”, Ph.D.diss., North Carolina State University (2002).

[4] A.C.Rocha and Edite M.G.P.Fernandes, “Feasibility and Dominance Rules in the Electromagnetism-like Algorithm for Constrained Global Optimization”, In Proceedings of the International Conference on Computational Science and Its Applications. (2008): 768-783.

[5] M.M.Ali, M.Golalikhani, “An Electromagnetism-like Method for Nonlinearly Constrained Global Optimization”, Computers and Mathematics with Applications. 60(2010):2279-2285.

[6] A.C.Rocha and M. G. P. Fernandes, “On Charge Effects to the Electromagnetism-like Algorithm”, International Conference 20th EURO Mini Conference, Neringa, LITHUANIA, (2008), May 20-23.

[7] A.C.Rocha and M.G.P.Fernandes, “Numerical Experiments with a Population Shrinking Strategy within a Electromagnetism-like Algorithm”, International journal of mathematics and computers in simulation, vol.1, no.3,(2007),pp.238-243.

[8] A.C.Rocha and M.G.P.Fernandes, “Modified Movement Force Vector in a Electromagnetism-like Mechanism for Global Optimization”, Optimization Methods and Software, vol.24, no.2,(2009), pp.253-270.

[9] C. H. Lee, F. K. Chang, and Y. C. Lee, “An Improved Electromagnetism-like Algorithm for Recurrent Neural Fuzzy Controller Design”, International Journal of Fuzzy Systems, vol.12, no.4,(2010),pp.280-290.

[10] M. Gol Alikhani , N. Javadian and R. Tavakkoli-Moghaddam, “A Novel Hybrid Approach Combining Electromagnetism-like Method with Solis and Wets Local Search for Continuous Optimization Problems”, Journal of Global Optimization, vol.44, no.2, (2009),pp.227-234.

[11] F. Yin, Y.N. Wang and W. S. Wei, “Inverse Kinematic Solution for Robot Manipulator based on Electromagnetism-like and Modified DFP Algorithms”, ACTA AUTOMATICA SINICA ,vol.37, no.1, (2011),pp.74-82.

[12] T. Ye, D. Y. Liu, X. Ma and C. S. Zhang, “A hybrid PSO with EM for Global Optimization”, International Journal of Modelling Identification&control, vol. 8, no.4, (2009),pp.327-334.