Journal of Computational Information Systems 9: 18 (2013) 7397–7405
Available at http://www.Jofcis.com

# An Activity Improved Bacterial foraging Optimization and Its Performance Analysis ⋆

Fuqing ZHAO [1,2,*], Xin JIANG [1], Junbiao WANG [2]

[1] *School of Computer and Communication Technology, Lanzhou University of Technology, Lanzhou 730050, China*

[2] *Key Laboratory of Contemporary Design, Integrated Manufacturing Technology, Ministry of Education, Northwestern Polytechnical University, Xi'an 710072, China*

**Abstract**

This paper proposes an activity improved bacterial optimization (AIBFO) aims to solve the stationary problem that cause by the failure tumbling in chemotaxis of the original Bacterial Foraging Optimization. The novel algorithm introduces a differential evolution operator into the chemotactic operation to optimize the stationary bacterium after their failing tumble in order to improve the effectiveness of chemotaxis. The evolution method and convergence of the algorithm have been proved theoretically in the paper. Typical example experiments show that the novel algorithm has great ability of avoiding the local optimum and performs a faster convergent speed and search accuracy in solving high dimensional problems.

*Keywords*: Bacteria Foraging Optimization; Chemotaxis; Stationary Problem; Activity Improved; Difference Evolution

## 1 Introduction

Biologically inspired computing is an important branch of natural computing which imitates the nature of biological evolution process to search for optimal solution. Inspired by the nature biological communities, many intelligent computing methods proposed before were based on animal behaviors such as Ant Colony Optimization(ACO) [1], Shuffled Frog Leaping Algorithm(SFLA) [2], Particle Swam Optimization(PSO) [3] etc. The Bacterial Foraging Optimization(BFO) [4] is proposed by K. Passion in 2002, which was a novel biological computing method inspired by the global cooperation mechanisms of Ecoli bacteria searching for food in human intestinal.

Due to complex processes and coupled parameters that involved in the BFO may sensitively affect the solution, it is difficult to optimize this algorithm. In order to improve the BFO's performance, researches and optimizations in many aspects of the original BFO had been done

*Project supported by the National Nature Science Foundation of China (No. 61064011).

*Corresponding author.
*Email address:* fzhao2000@hotmail.com (Fuqing ZHAO).

1553–9105 / Copyright © 2013 Binary Information Press
DOI: 10.12733/jcisP0202
September 15, 2013

[5, 6, 7]. M.S. Lia proposed Bacterial foraging algorithm with varying population, which was based on the physiological characteristics of the natural bacterium. Pandit proposed an improved bacterial foraging algorithm (IBFA) which using a parameter automation strategy and crossover operation in the original micro BFO to improve computational efficiency. Verma Om Prakash used a combination of bacterial foraging optimization (BFO) and probabilistic derivative technique derived from Ant Colony Systems for edge detection.Then the novel approach obtains an ideal result.

Although the performance of BFO has been improved in many ways by the methods above, there are still some intrinsic problems in the original BFO that need to be solved. This paper proposes a Activity Improved Bacterial Foraging Optimization (AIBFO) which introduce a differential mutation operator from DE Algorithm aims to solve the problem of tumble failure in the chemotatic step of the original BFO.

# 2 The Differential Evolution Operator

The DE algorithm [9, 10] is proposed by Storn R and Price K at 1995 which use the mutual cooperation and competition among individuals within the group for the optimization searching. Considering the parameters setting in the BFO, the differential mutation operator is defined as in Eq. (1)

$$v_d^i = \theta_d^i(j,k,l) + F \cdot (\theta_d^{gb}(j,k,l) - \theta_d^{R2}(j,k,l) + \theta_d^{R1}(j,k,l) - \theta_d^{R3}(j,k,l)) \tag{1}$$

Where $d$ refers to the dimension of solution, $\theta_d^i(j,k,l)$ represents the $i^{th}$ bacterium after $j^{th}$ chemotactic step, $v_d^i$ represents the $i^{th}$ bacterium during differential mutation, $\theta_d^i$ is the global best, $\theta_d^{R1}, \theta_d^{R2}, \theta_d^{R3}$ are three individuals that are randomly chosen from the whole group, $F \in [0.2, 0.9]$ refers to the differential factor which controls the degree of mutation.

In Eq. (1), the global best and three other individuals from the current population are introduced into the mutating bacterium. The individual after mutated is compared with the original one by Eq. (2). Only if the new fitness value after mutation is better than the previous fitness value, the original individual will be replaced.

$$\widetilde{\theta^i} = \begin{cases} v^i & \text{if } f(u^i) < f(\theta^i(j+1,k,l)) \\ \theta^i(j,k,l) & \text{if } otherwise \end{cases} \tag{2}$$

Eq. (2) is the greedy selection operation which always retains the outstanding individuals of the previous generation.

# 3 The Optimization of BFO

## 3.1 The insufficient in chemotaxis of BFO

The original BFO makes the evolution and global optimum searching by three basic operations: Chemotaxis, Reproduction and Elimination-dispersal [4]. The original chemotaxis step can be

defined as a tumble followed by a run. The tumble process determines the direction and makes a tentative step by Eq. (3), then start the running process if the fitness value becomes better.

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i) \cdot \Delta \tag{3}$$

Where $\theta^i(j,k,l)$ denotes the location of the $i^{th}$ bacterium at $j^{th}$ chemotactic step, $k^{th}$ reproduction step, and $l^{th}$ elimination-dispersal step. $C(i)$ is the size of the step taken in the random direction. $\Delta = \frac{\Delta(i)}{\sqrt{\Delta(i)^T \cdot \Delta(i)}}$ represents a unit direction, $\Delta(i)$ is a random vector. During the run process, the bacterium will calculate its fitness value after each moving step. Bacterium will keep moving only if the fitness value at the new place is better than the formal one, or skip running and start next tumble.

However, in the original chemotaxis, even though the bacterium wont move forward after it finds out the fitness value becomes worse, the probability of run process is decreased and then convergence speed of algorithm will be slowed down if this king of failing tumble happens frequently.

In order to decrease the stationary probability of bacterium, a correction should be given to make sure that the bacterium can fix its position. The $f(.)$ in this paper represents the objective function.

## 3.2 The process of AIBFO Algorithm

Define parameter $Nc$, $Nre$ and $Ned$ as the maximum number of chemotactic, reproduction and elimination-dispersal step, $j, k, l$ as the counting index of three operations. Set $Ns$ as the longest swimming distance, $m$ as the index of the swimming step.

**Step 1** Initialize parameters $Ned = 1, Nre = 1, Nc = 1$, go to Step 2.

**Step 2** If $j < Nc$, $j = j + 1$. For each bacterium, do the following processes:

**a.** Tumble: Generating a random direction then make a move by Eq. (3) and calculate the fitness value at current position. If the value is better than previous one, go to b, otherwise go to c.

**b.** Run: When $m < Ns, m = m + 1$. Calculating the fitness value at each step. If better than last step, keep running until $m = Ns$ and go to Step 3, otherwise stop running and go to Step 3.

**c.** Differential mutation: For the tumble failure cases, making differential mutation to fix bacteriums position through Eq. (1), Eq. (2)and Eq. (3), then go to Step 3.

**Step 3** If $k < Nre$, $k = k + 1$, start the reproduction as the original BFO, then go to Step 2, otherwise go to Step 4.

**Step 4** If $l < Ned$, $l = l + 1$, start the elimination-dispersal step as the original BFO, then go to Step 2, otherwise end.

### 3.3 Analysis of the convergence in AIBFO

**Definition 1** *For random sequence* $\{\overrightarrow{X}(t), \forall X(t) \in S\}_{t=0}^{+\infty}$. *S is the state space. For* $t \geq 1$ *and* $i_{k+1} \in S(k \leq t+1)$, *if* $P\{X(t+1) = i_{t+1} \mid X(t) = i_t, \cdots, X(0) = i_0\} = P\{X(t+1) = i_{t+1} \mid X(t) = i_t\}$ $\{\overrightarrow{X}(t), \forall X(t) \in S\}_{t=0}^{+\infty}$ *is a Markov chain.*

**Definition 2** *Described the global optimal solution sets as* $M = \{X^*; \forall X^* \in S\}$. *M refers to the feasible solution sets.*

**Theorem 1** *The populations fitness value of AIBFO is monotonous nonincreasing.*

**Proof** Reproduction and elimination operations in AIBFO are following the survival of the fittest rule which ensures the retaining of optimal solutions in each generation and eliminating poor solutions that makes the overall fitness of the current population is heading to $minf(x)$ and no increasing. Theorem 1 has proven.

**Theorem 2** *The AIBFO generates new individual with Markov character, its solutions sequence* $\{\overrightarrow{X}(t), \forall X(t) \in S\}_{t=0}^{+\infty}$ *is a Markov chain.*

**Proof** Set $\xi(t)$ as the transient between $X(t)$ and $X(t+1)$ ,the new individual generating process can be described as follow:
In chemotaxis:
If tumbling success, the bacteriums position is changed by Eq. (1):

$$\xi(t) = X(t) + m \cdot C \cdot \Delta \tag{4}$$

If tumble failure, the bacteriums position is changed through Eq. (1), Eq. (2) and Eq. (3):

$$\xi(t) = \begin{cases} u(t) & \text{if } f(u(t)) \leq f(X(t)) \\ X\,(t) & \text{owherwise} \end{cases} \tag{5}$$

Where $u(t) = X(t) + \psi(X^{gb}, X^{R1}, X^{R2}, X^{R3})$ represents the simplification of Eq. (1), $\psi(.)$ is the differential variable.

Set $Sr \in S$ to be the advantage solution sets in current population. The new individual is generated by the advantage individual from the current population or its copying. So the process of new individual producing can be defined as:

$$X(t+1) = \begin{cases} \xi(t) & \text{if } \xi(t) \in Sr \\ X'(t) & \text{otherwise} \end{cases} \tag{6}$$

Where $X'(t)$ represents the complete copying of the advantage individual in current population, while $X'(t)$ and $\xi(t)$ are all generated by Eq. (4) and Eq. (5). Therefore the relation between the new individual and the current one can be abstracted as a linear relationship as shown in Eq. (7):

$$X(t+1) = X(t) + \alpha(t) \cdot \beta(t) \tag{7}$$

Where $\alpha(t)$ represents the linear coefficient of variation, $\beta(t)$ represents the linear amplitude of variation. Then

$$\begin{aligned}
&P\{X_{t+1} = i_{t+1} \mid X_t = i_t, X_{t-1} = i_{t-1}, \cdots, X_0 = i_0\} \\
&= P\{X_t + \alpha(t) \cdot \beta(t) = i_{t+1} \mid X_t = i_t, \cdots, X_0 = i_0\} \\
&= P\{X_{t-1} = i_{t+1} - \alpha(t) \cdot \beta(t) - \alpha(t) \cdot \beta(t) \mid X_t = i_t, \cdots, X_0 = i_0\} \\
&\cdots \\
&= P\{X_0 = i_{t+1} - \sum_{k}^{t} \alpha(k) \cdot \beta(k) \mid X_t = i_t, \cdots, X_0 = i_0\} \\
&= P\{X_{t+1} \mid X_t = i_t\}
\end{aligned}$$

From the reasoning above, it can be concluded that the production of $X(t+1)$ in AIBFO is only related with $X(t)$. According to definition 1, the population sequence $\{\overrightarrow{X}(t), \forall X(t) \in S\}_{t=0}^{+\infty}$ in AIBFO is a Markov chain. Theorem 2 has proven.

**Theorem 3** *For* $\{\overrightarrow{X}(t), \forall X(t) \in S\}_{t=0}^{+\infty}$ *produced by AIBFO that satisfies the Theorem 1 and Theorem2, if* $P\{X(t) \in M \mid X(t-1) \notin M\} \geq d(t-1) \geq 0$, $\lim\limits_{t \to +\infty} \prod_{i=0}^{t} [1 - d(i)] = 0$, *then* $\lim\limits_{t \to +\infty} \delta(t) = 1$, *where* $\delta = \{X(t) \in M\}$.

**Proof** According to the whole probability formula, when $\forall t = 1, 2, \cdots$. We get

$$\delta(t) = [1 - \delta(t)] P\{X(t) \mid X(t-1) \notin M\} + \delta(t-1) P\{X(t) \in M \mid X(t-1) \in M\} \tag{8}$$

According to Theorem 1 and Theorem 2, the solution sequence of AIBFO is a decreasing monotonous Markov chain. If $X(t-1)$ has already entered the global optimum sets $M$, obviously that $X(t)$ is bound to be concluded in $M$ from Eq. (6). Thus $P\{X(t) \in M \mid X(t-1)\} = 1$, then Eq. (8) can be transformed as

$$\begin{aligned}
&\delta(t) = [1 - \delta(t)] P\{X(t) \in M \mid X(t-1) \notin M\} + \delta(t-1) \\
&\Rightarrow 1 - \delta(t) = [1 - P\{X(t) \in M \mid X(t-1) \notin M\}] \cdot [1 - \delta(t-1)] \\
&\Rightarrow 1 - \delta(t) \leq [1 - d(t-1)][1 - \delta(t-1)] \\
&\Rightarrow \lim_{t \to +\infty} \delta(t) \geq 1 - [\delta(0)] \lim_{t \to +\infty} \prod_{i=0}^{t-1} [1 - d(i)] \\
&\Rightarrow \lim_{t \to +\infty} \delta(t) \geq 1 (because \lim_{t \to +\infty} \prod_{i=0}^{t-1} [1 - d(i)] = 0)
\end{aligned}$$

and $\delta \leq 1$, thus $\lim\limits_{t \to +\infty} \delta(t) = 1$. Theorem 3 has proven. According to Theorem 3, $\lim\limits_{t \to +\infty} \delta(t) = \lim\limits_{t \to +\infty} P\{X(t) \in M\} = 1$, which means the solution sequence $\{\overrightarrow{X}(t), \forall X(t) \in S\}_{t=0}^{+\infty}$ of AIBFO will converges to global optimal M sets with probability of 1.

# 4 Stimulation Studies

## 4.1 Experimental condition and benchmark functions

The experiment is a MATLAB simulation test, taken on the computer of Windows7 operation system, Core i5 CPU and 2G RAM. In order to test the performance of AIBFO, eight benchmark functions are used, as shown in Table1, the dimensions of the functions are set at 30-dimensional. Where $f_1 \sim f_3$ are unimodal functions, $f_4 \sim f_8$ are multimodal functions.

Table 1: Benchmark functions

| **Funtion** | **Mathematical Representation** |
|---|---|
| *Sphere* | $f_1 = \sum_{i=1}^{30} x_i^2, \;\; min(f_1) = f_1(0, ..., 0) = 0$ |
| *Quartic* | $f_2 = \sum_{i=1}^{30} i x_i^4 + random[0, 1) \;\; min(f_2) = f_2(0, ..., 0) = 0$ |
| *Rosenbrock* | $f_3 = \sum_{i=1}^{29} [100(x_{i+1} - x_i^2) - (x_i - 1)^2] \;\; min(f_3) = f_3(0, ..., 0) = 0$ |
| *Schewefel* | $f_4 = 418.9282 * 30 - \sum_{i=1}^{30} [x_i sin\sqrt{\lvert x_i \rvert}] \;\; min(f_4) = f_4(420.9687, ..., 420.9687) = 0$ |
| *Rastrigin* | $f_5 = \sum_{i=1}^{30} [x_i^2 - 10\cos 2\pi x_i + 10] \;\; min(f_5) = f_5(0, ..., 0) = 0$ |
| *Ackley* | $f_6 = -20exp(-0.2\sqrt{\frac{1}{30}\sum_{i=1}^{30} x_i^2}) - exp(\frac{1}{30}\sum_{i=1}^{30} cos2\pi x_i) + 20 + e$<br>$min(f_6) = f_6(0, ..., 0) = 0$ |
| *Griewank* | $f_7 = \frac{1}{4000}\sum_{i=1}^{30} x_i^2 - \prod_i^{30} \cos(\frac{x_i}{\sqrt{i}}), \;\; min(f_7) = f_7(0, ..., 0) = 0$ |
| *Penalized* | $f_8 = \frac{\pi}{30}\{10\sin^2(\pi y_1) + \sum_{i=1}^{29} y_i - 1)^2 \cdot [1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\}$<br>$+ \sum_{i=1}^{30} u(x_i, 10, 100, 4), \;\; min(f_8) = f_8(0, ..., 0) = 0$<br>$y_i = 1 + \frac{1}{4}(x_i + 1)$ ,<br>$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x > a \\ 0 & -a \le x_i \le a \\ k(-x_i - a)^m & x < -a \end{cases}$ |

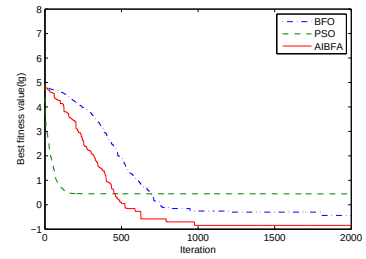
## 4.2 Simulation result and analysis

In the simulation studies, AIBFO is evaluated on the functions in comparison with BFO and PSO respectively. Considering the parameter setting in references [3, 7, 9, 11, 12], the population size is set to be 50. In AIBFO, $CR$ is set to be 0.9 and $F$ is set to be 0.5. For the parameters of PSO the inertia weight $\omega$ is set to be 0.729 and $c_1 = c_2 = 1.8$. The results listed in Table 2 are the average mean solution of each algorithm obtained from 40 runs of the algorithm on each benchmark function.

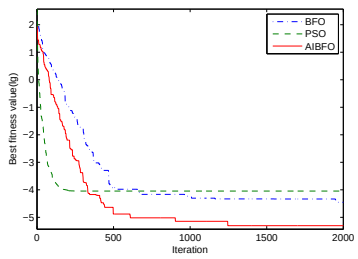
It can be seen from Table 2 that AIBFO outperforms BFO and PSO on all the benchmark functions especially for the multimodal functions $f_4 \sim f_8$. This means the optimized chemotaxis in AIBFO can effectively avoid the local optimal problem and has more advantage in solving high dimensional multimodal problems and provide better searching ability. The convergence curves of three algorithms for each benchmark function are shown in Fig. 1.

Table 2: Comparison of experiment result

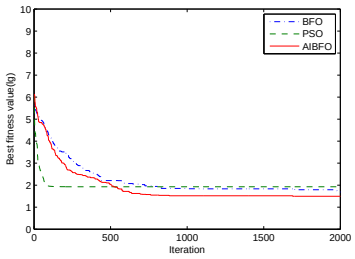
| Function | Value | BFO | PSO | AIBFO |
|---|---|---|---|---|
| $f_1(30D)$ | mean | $0.633 \pm 0.196$ | $3.881 \pm 3.624$ | $0.146 \pm 0.011$ |
| | Best | 0.434 | 3.231 | 0.078 |
| $f_2(30D)$ | mean | $1.622 \pm 2.231e-5$ | $4.1e-2 \pm 0.058$ | $3.878e-6 \pm 1.884e-6$ |
| | Best | $4.531e-4$ | $4.433e-3$ | $4.206e-5$ |
| $f_3(30D)$ | mean | $72.834 \pm 2.83$ | $78.373 \pm 8.84$ | $30.630 \pm 5.11$ |
| | Best | 49.211 | 72.251 | 25.579 |
| $f_4(30D)$ | mean | $1.037e3 \pm 6.22e2$ | $2.207e3 \pm 3.41e2$ | $8.048e1 \pm 1.01e2$ |
| | Best | $4.621e2$ | $2.007e3$ | 8.624 |
| $f_5(30D)$ | mean | $1.541e2 \pm 20.741$ | $1.257e2 \pm 20.080$ | $32.471 \pm 1.086$ |
| | Best | $1.212e2$ | $1.121e2$ | 13.992 |
| $f_6(30D)$ | mean | $16.832 \pm 1.980$ | $5.792 \pm 2.584$ | $0.931 \pm 0.584$ |
| | Best | 14.995 | 4.258 | 0.098 |
| $f_7(30D)$ | mean | $0.897 \pm 0.041$ | $4.015 \pm 1.222$ | $1.873e-2 \pm 0.012$ |
| | Best | 0.977 | 1.657 | $1.312e-3$ |
| $f_8(30D)$ | mean | $4.253e1 \pm 2.999$ | $3.369 \pm 1.281$ | $0.310 \pm 0.297$ |
| | Best | $2.416e1$ | 2.048 | $5.361e-2$ |



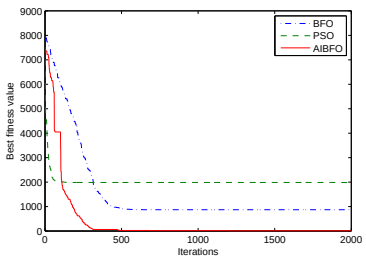

(a) Sphere



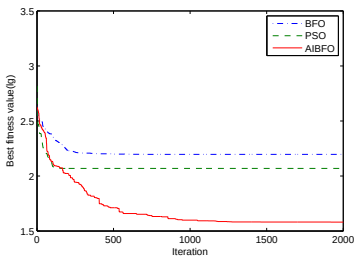

(b) Quatic



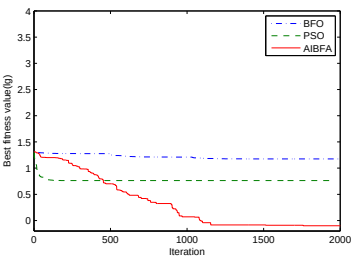

(c) Rosenbrock



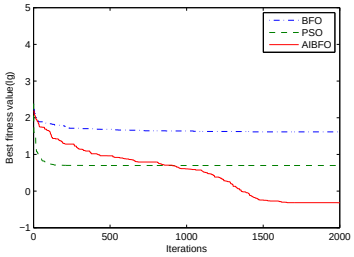

(d) Schewefel



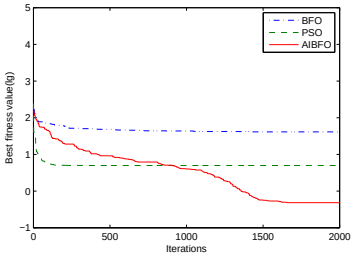

(e) Rastrin




(f) Ackley




(g) Griewank




(h) Penalized

Fig. 1: Comparisons of best solution value of three algorithms

It can be seen from Fig. 1, with the limited number of iterations, PSO has the fastest convergence speed than the other two algorithms but its curves tends to horizon soon, which means its ability of jumping out of local optimum is weak and the diversity of its population is low. In this case, the searching of PSO is easily became premature and converges to the local optimal solution. From (d)∼(h), we can see that the BFO also has the problem of falling into local optimum. With the differential evolution operator added, the AIBFO performs faster convergence speed than the original BFO. As shown in (a)∼(d) and (g), while in high dimensional unimodal problems, the AIBFO can quickly converge to the global optimal region and proceed local searching. In(d)∼(h), the AIBFO can jump out of the local optimal solution soon.

# 5 Conclusion

This paper proposes an activity improved bacterial optimization to fix the stationary problem that cause the slow convergence and easily falling into local optimum that the original BFO performs in high dimensional tests. The novel optimization algorithm introduces the evolution operator from DE algorithm into the chemotaxis operation in BFO to optimize the bacteriums position after improper tumble. The effectiveness and convergence of AIBFO are proved in theory. The simulation experiments proof that the AIBFO has better convergence speed and accuracy than the original BFO and PSO in solving high dimensional problems and can effectively avoid the condition of trapping into local optimal solution.

# Acknowledgment

This work was financially supported by the National Natural Science Foundation of China under grant numbers 61064011. It was also supported by scientific research funds from Gansu University, the General and Special Program of the Postdoctoral Science Foundation of China, the Science Foundation for Excellent Youth Scholars of Lanzhou University of Technology under grant numbers 1114ZTC139, 2012M521802, 2013T60889 and 1014ZCX017, respectively.

# References

[1] A colorni, Distributed optimization by ant colonies, in: Proc. The first European conference on artificial life, 1991, pp. 134-122.

[2] Eusuff, Optimization of water distribution network design using the shuffled frog leaping algorithm(2003) 210-225.

[3] Kennedy, J, Partcle swarm optimization, in: proc. Neural Networks, 1995, pp. 1942-1948.

[4] K.M.Passino, Biomimicry of bacterial foraging for distributed optimization and control, Control Systems, IEEE vol.22, 2002, pp. 52-67.

[5] N.pandit, An improved bacterial foraging algorithm for combined static dynamic environmental economic dispatch, Applied Soft Computing vol.12, 2012, pp. 3500-3513.

[6] Verma, O.P, A novel bacterial foraging technique for edge detection, Pattern Recognition Letters vol.32, 2011, pp. 1187-1196.

[7] Li, M. S, Bacterial foraging algorithm with varying population, Biosystems vol.100, 2010, pp. 185-197.

[8] R.Azizipanah-Abarghooee, A new hybrid bacterial foraging and simplified swarm optimization algorithm for practical optimal dynamic load dispatch, International Journal of Electrical Power and Energy Systems vol.49, 2013, pp. 416-429.

[9] R.Storn, Differential Evolution-A Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces, Technical Report TR-95-012, 1995.

[10] Price, Kenneth V, Differential evolution a practical approach to global optimization, Springer-Verlag, 2005.

[11] Biswas, Analysis of the reproduction operator in an artificial bacterial foraging system, Applied Mathematics and Computation vol.215, 2010, pp. 3343-3355.

[12] Dasgupta, S, Adaptive computational chemotaxis in bacterial foraging optimization: an analysis, Evolutionary Computation, IEEE Transactions on vol.13, 2009, pp. 919-941.