Analysis of Chemotaxis in Bacterial Foraging Optimization Algorithm

Livjeet Kaur
Department of ECE
SLIET, Longowal
Punjab (India)

Mohinder Pal Joshi
Department of ECE
SLIET, Longowal
Punjab (India)

ABSTRACT
For the last few decades, algorithms like Genetic Algorithms, Evolutionary Programming, and Evolutionary Strategies etc. are being used for optimization of various problems. Nowadays various swarm inspired algorithms have replaced them. Bacterial Foraging Optimization (BFO) is the latest among these algorithms. It has been widely accepted as global optimization technique due to its ease of implementation. In this paper we analyzed chemotactic behavior of bacteria by minimizing various mathematical benchmark functions. MATLAB simulations of these functions for different step sizes are shown in graphical form. Work is concluded by discussing the effect of varying step size on chemotactic movement of bacteria.

General Terms
Bacterial Foraging, Chemotaxis, Swarm Intelligence, Global Optimization

Keywords
ASO, PSO, BFO, BFOA.

1. INTRODUCTION
Nature has always taught the living beings to adapt according to its ways for their well being. It guides various species in different ways. Homo sapiens, being the most intelligent species, has always tried and proved himself successful to exploit the nature and its ways for his self motives. From last few decades, scientists have been imitating various natural procedures like evolution, natural genetics or group behavior of various creatures to attain a desired objective. These nature inspired methods can be used to optimize many real world problems.

Various methods have been defined for optimization which can be broadly classified into three methods namely deterministic, stochastic and heuristic methods. Deterministic optimization methods depend on mathematical properties for finding the optimum solution. These techniques are dependent on gradient, local optimums and are inefficient in large scale search space [1]. Commonly used deterministic methods are Inner approximations, Outer approximations, Cutting methods, Branch and Bound methods and many more. Stochastic optimization methods are the methods that use random variables and random iterates to solve stochastic problems. Randomness is introduced in the search-process to accelerate search progress. This type of randomness can also make the method less sensitive to modeling errors. These optimization methods include Simulated annealing, Stochastic tunneling, Parallel tampering, Monte Carlo sampling etc. Some of the stochastic methods like Stochastic Gradient Descent and Finite difference Stochastic Approximation use statistical inference tools to estimate the true values of the function and/or make statistically optimal decisions about the next steps. The third optimization method i.e. heuristic is a computational method that uses iterations to improve an optimal solution of any problem. This method makes very less assumptions about the problem being optimized and can search very large spaces of candidate solutions. Many heuristics implement some form of stochastic optimization. Due to this reason, these methods are also called as combinatorial methods. These include Genetic Algorithms, Swarm based optimization algorithms, Memetic Algorithms etc. Genetic algorithm (GA), which was originally conceived by J. Holland, is based on Darwinian evolution and biological genetics [2]; Swarm Intelligence uses the collective behavior of animals to achieve the desired goal. There are two popular swarm inspired methods: Ant Colony optimization (ACO) and Particle Swarm Optimization (PSO). Proposed by Marco Dorigo et al., ACO is based on foraging behavior of ant colonies [3]. PSO, proposed by Eberhart Kennedy, is inspired by social behavior of flocks of birds and schools of fish [4]. Currently, these nature inspired techniques are being used for finding better quality solutions in optimization problems and formulate better decision making mechanisms. Bacterial Foraging Optimization Algorithm (BFOA) proposed by Passino, is a newcomer in this field. BFO is inspired by social foraging behavior of Escherichia coli bacteria [5]. BFO has been already applied to a number of problems like adaptive control (Kim & Cho, 2005b), harmonic estimation (Mishra, 2005), machine learning (Kim & Cho, 2005a), and optimal power flow scheduling (Tang et al., 2006) [6]. The objective of this paper is to analyze the effect of varying step size of bacterium on its chemotactic behavior.

The rest of the paper is organized as follows. Section 2 provides a review of BFO Algorithm. Section 3 describes graphical analysis of chemotactic movement of bacteria using various mathematical functions. Section 4 presents discussion of results. Finally, in last section, we give conclusion of the work.

2. BFO REVIEW
BFO is based on foraging strategy of bacteria Escherichia coli. After many generations, bacteria with poor foraging strategies are eliminated while; the individuals with good foraging strategy survive signifying survival of the fittest. The whole process can be divided into three sections, namely, chemotaxis, reproduction, and elimination and dispersal [5].

2.1 Chemotaxis
Chemotaxis can be defined as foraging behavior of bacteria in which it try to avoid noxious substances and search for nutrient rich substances by climbing up the nutrient

concentration. This process involves two actions; either a run (in the same direction as the previous step) or tumble (in an absolutely different direction from the previous one). In order to explore whole search space there is a limit on run steps in a particular direction. So, bacteria tumble after some run steps. Suppose $\theta^i(j,k,l)$ be the position of i^{th} bacterium at j^{th} chemotactic, k^{th} reproductive and l^{th} elimination & dispersal loop. Then chemotactic movement of the bacterium may be mathematically represented by following equation.

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}},$$

In the above expression, $C(i)$ is the size of the step taken in random direction and $\Delta(i)$ indicates a vector in the arbitrary direction. When the bacterial movement is run, $\Delta(i)$ remains unchanged; otherwise, $\Delta(i)$ is a random vector whose elements lie in [−1, 1]. Fitness function, denoted as $J(i, j, k, l)$, will be evaluated for each step of run or tumble in the chemotactic process.

2.2 Reproduction

The health of each bacterium is calculated as the sum of the step fitness during its life, namely,

$$J^i_{health} = \sum_{j=1}^{N_c+1} J(i,j,k,l)$$

where N_c is number of chemotactic steps. All bacteria are sorted in increasing order according to health status. In the reproduction step, only the first half of population survives and a surviving bacterium reproduces by splitting into two daughter bacteria, which are then placed at the same locations. Thus, the population of bacteria keeps constant.

2.3 Elimination & Dispersal

The chemotaxis provides a basis for local search, and the reproduction process speeds up the convergence of the algorithm. However, only chemotaxis and reproduction are not enough for global optima searching. Since bacteria may get stuck around the initial positions or local optima, it is possible for the diversity of BFO to change either gradually or suddenly to eliminate the accidents of being trapped into the local optima. In BFO, according to a preset probability P_{ed}, bacteria are eliminated and dispersed after a certain number of reproduction steps. After elimination, they are moved to another position within the environment.

Algorithm:

Step 1. Initialize parameters p, S, N_c, N_s, N_{re}, N_{ed}, P_{ed}, C (i) (i = 1, 2, . . ., S), θ^i,
where
p: dimension of the search space,
S: the number of bacterium,
N_c: chemotactic steps,
N_s: swimsteps,
N_{re}: reproductive steps,
N_{ed}: elimination and dispersal steps,
P_{ed}: probability of elimination,
$C(i)$: the run-length unit (i.e., the chemotactic step size during each run or tumble).

Step 2. Elimination-dispersal loop: $l = l + 1$.
Step 3. Reproduction loop: $k = k + 1$.
Step 4. Chemotaxis loop: $j = j + 1$.
Substep 4.1. For i = 1, 2…, S, take a chemotactic step for bacteria i as follows.
Substep 4.2. Compute fitness function, $J(i,j,k,l)$.
Substep 4.3. Let $J_{last} = J(i,j,k,l)$ to save this value since we may find better value via a run.
Substep 4.4. Tumble: generate a random vector $\Delta(i) \in R^p$ with each element $\Delta m(i)$, m = 1, 2, . . .,p, a random number on [−1, 1]
Substep 4.5. Move: Compute $\theta^i(j+1,k,l)$.This results in a step of size $C(i)$ in the direction of the tumble for bacteria i.
Substep 4.6. Compute $J(i, j+1, k, l)$ with $\theta^i(j+1, k, l)$.
Substep 4.7. Swim:
(i) Let m =0 (counter for swim length)
(ii) While $m < N_s$ (if not climbed down too long)
a) Let $m = m + 1$
b) If $J(i, j+1, k, l) < J_{last}$, Let $J_{last} = J(i, j+1, k, l)$, Then, another step of size $C(i)$ in the same direction will be taken as Eq.(1) and use the new generated $\theta^i(j+1, k, l)$ to compute the new $J(i, j+1, k, l)$.
c) Else Let $m = N_s$.
Substep 4.8. Go to next bacterium (i+1): if $i \neq S$ go to 4.2 to process the next bacteria.

Step 5. If $j < N_c$, go to Step 4. In this case, continue chemotaxis since the life of the bacteria is not over.
Step 6. Reproduction.
Substep 6.1. For the given k and l, and for each i =1, 2, .. ., S, let

$$J^i_{health} = \sum_{j=1}^{N_c+1} J(i,j,k,l)$$

be the health of the bacteria. Sort bacterium in order of ascending values (J_{health}).
Substep 6.2. The S_r bacteria with the highest J_{health} values die, and the other S_r bacteria with the best values split, and the copies that are made are placed at the same location as their parent.
Step 7. If $k < N_{re}$ go to Step 2. In this case, the number of specified reproduction steps is not reached; start the next generation in the chemotactic loop.
Step 8. Elimination-dispersal: for i =1, 2, .. ,S, with probability P_{ed}, eliminate and disperse each bacteria, which results in keeping the number of bacteria in the population constant. To do this, if a bacterium is eliminated, simply disperse one to a random location on the optimization domain. If $l < N_{ed}$, then go to Step 2, otherwise, end [7].

3. MATHEMATICAL ANALYSIS OF CHEMOTAXIS

3.1 Mathematical Functions and BFO parameters

To analyze the chemotactic movement of bacteria, we consider a set of mathematical benchmark functions of varying complexity and dimensions. The considered mathematical functions are

1) Simple 1-dimensional function $f_1(x) = x(x-8)$
2) Two dimensional Sphere function $f_2(x, y) = x^2 + y^2$
3) Three dimensional Rastrigin function
$f_3(x, y, z) = 30 + x^2 + y^2 + z^2 - 10 (\cos(2\pi x) + \cos(2\pi y) + \cos(2\pi z))$

Our objective is to minimize these functions and to see the effect of varying step size on behavior of bacteria in a given search space.
Analysis is restricted within specified search ranges as given in table 1 where p is the dimension of search space.

Table 1. Detail of functions under consideration

Function	Range of Search Space	Minima
f_1	$(-10\ 10)^p$	$f_1(4) = -16$
f_2	$(-20\ 20)^p$	$f_2(0,0) = 0$
f_3	$(-20\ 20)^p$	$f_3(0,0,0) = 0$

The initialization values of various parameters for BFO are given in the following table:

Table 2. Parameters for BFO

Parameter	Initial Values
S	20
N_c	50
N_s	4
N_{re}	4
N_{ed}	2
S_r	S/2
P_{ed}	0.25

3.2 Results of MATLAB simulations

MATLAB simulations were done on considered benchmark functions by varying the step size in each simulation. Various graphs depicting the results are shown for all the functions.

1) $f_1(x) = x(x-8)$

In Fig 1 the function f_1 is shown followed by the subsequent Fig 2 to 4 showing the graphs of optimal solution provided by different bacteria for varying step sizes.

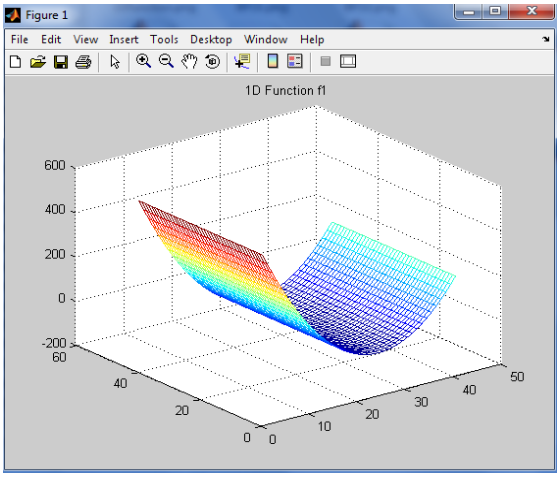

Fig 1. 1D Simple function

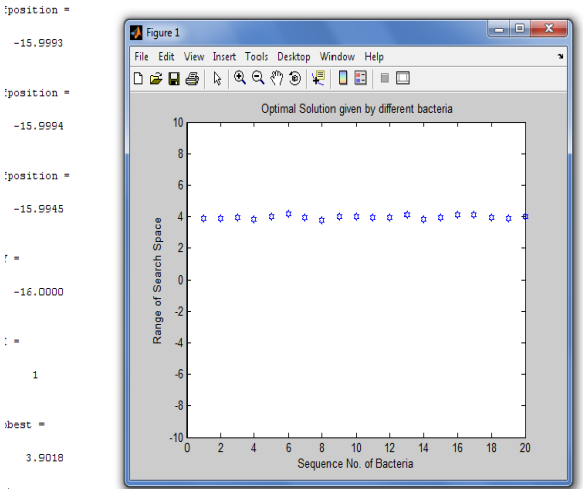

Fig 2. For step size, C(i) = 0.05

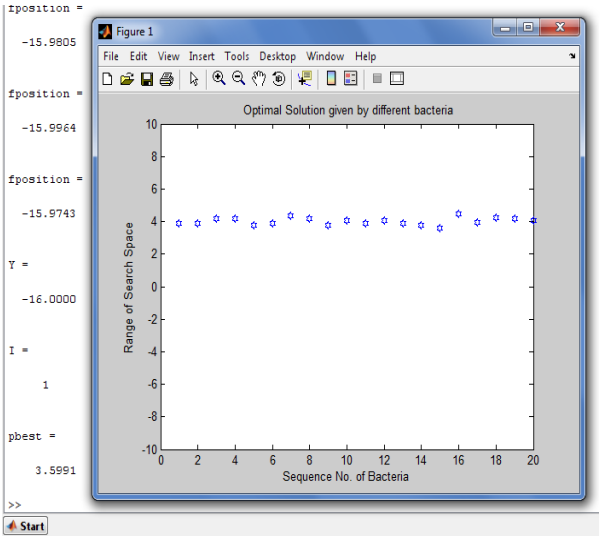

Fig 3. For step size, C(i) = 0.1

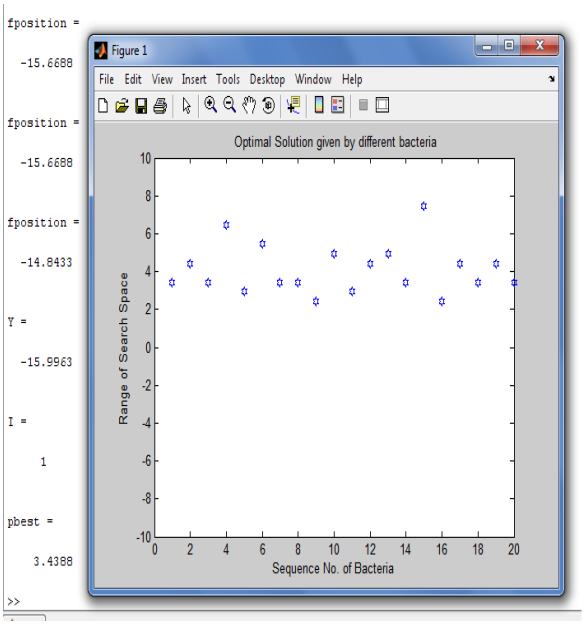

Fig 4. For step size, C(i) = 0.5

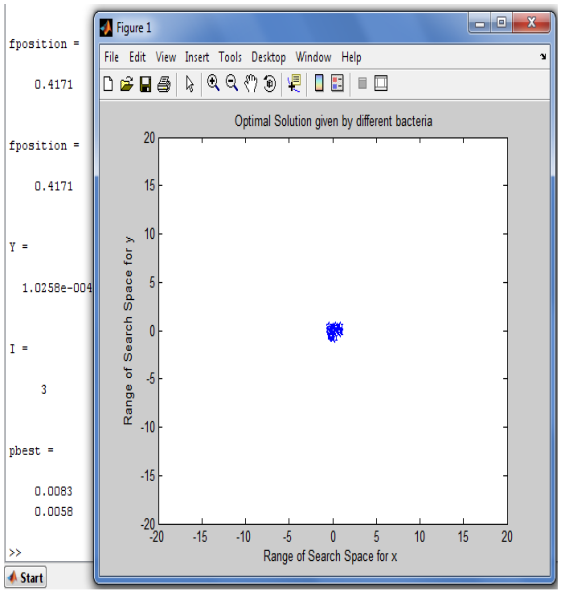

Fig 6. For step size C(i) = 0.05

2) $f_2(x, y) = x^2 + y^2$

In Fig 5 the sphere function is shown followed by the subsequent Fig 6 to 8 showing the graphs of optimal solution provided by different bacteria for varying step sizes.

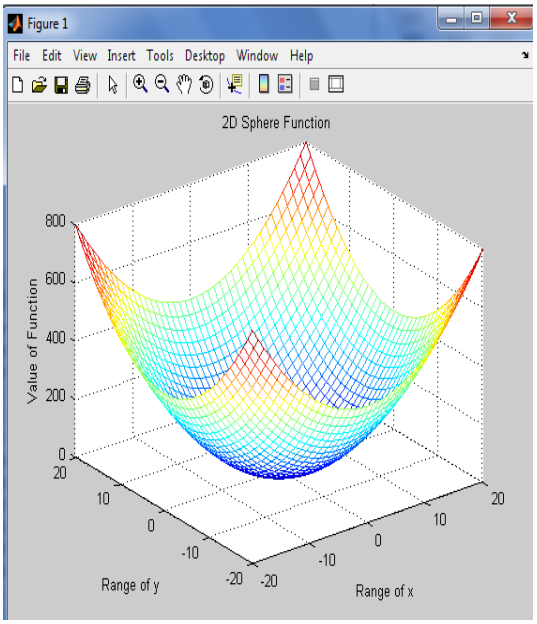

Fig 5. Sphere function

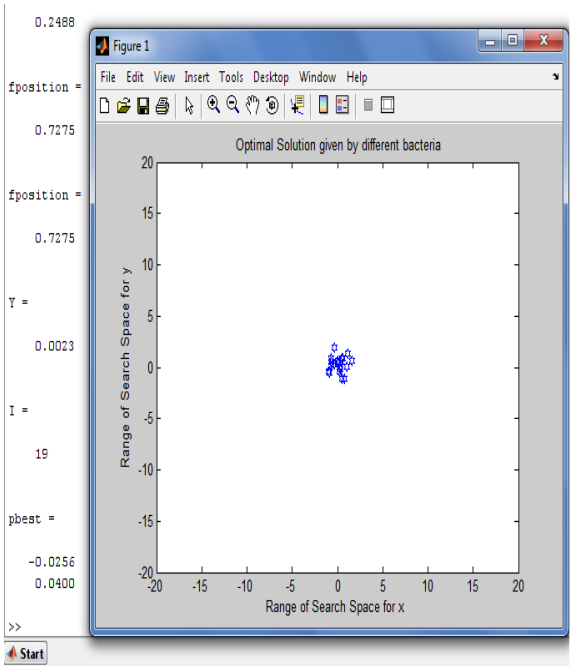

Fig 7. For step size C(i) = 0.1

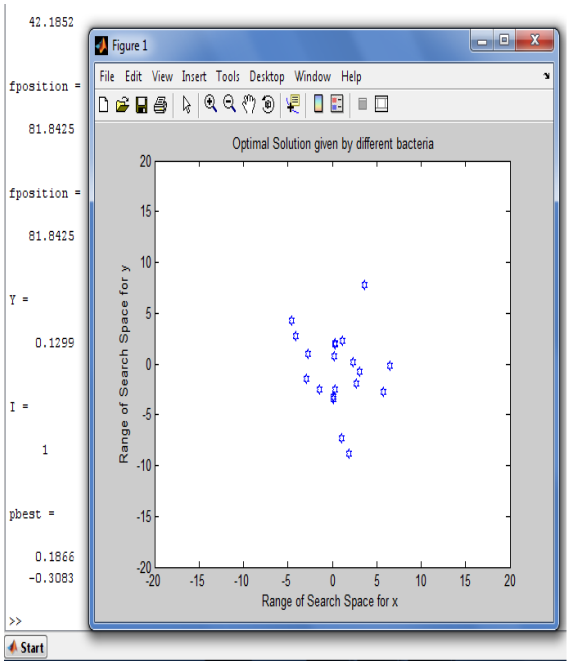

Fig 8. For step size, C(i) = 0.5

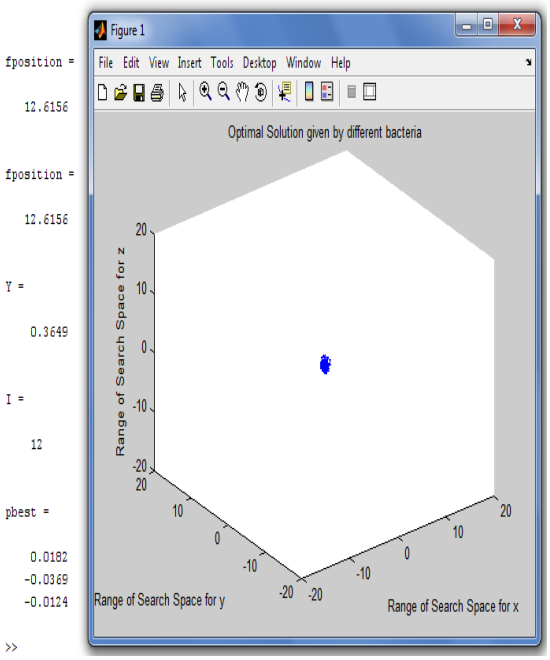

Fig 10. For step size C(i) = 0.05

3) $f_3(x, y, z) = 30 + x^2 + y^2 + z^2 - 10\,(\cos(2\pi x) + \cos(2\pi y) + \cos(2\pi z))$

In Fig 9 the Rastrigin function is shown followed by the subsequent Fig 10 to 12 showing the graphs of optimal solution provided by different bacteria for varying step sizes.

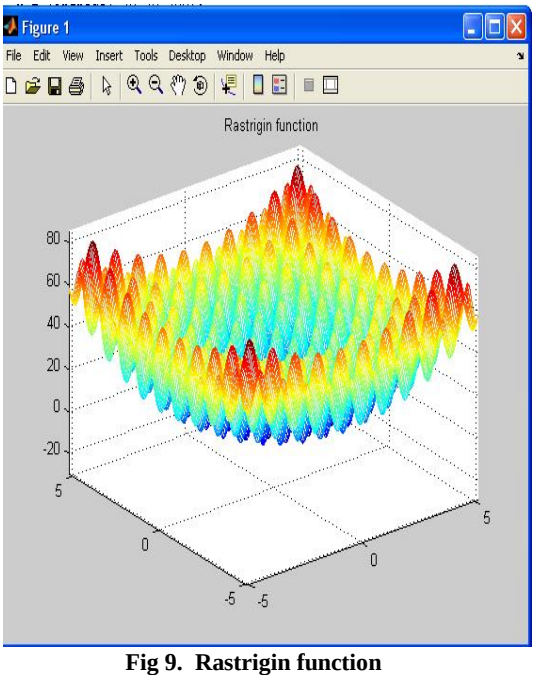

Fig 9. Rastrigin function

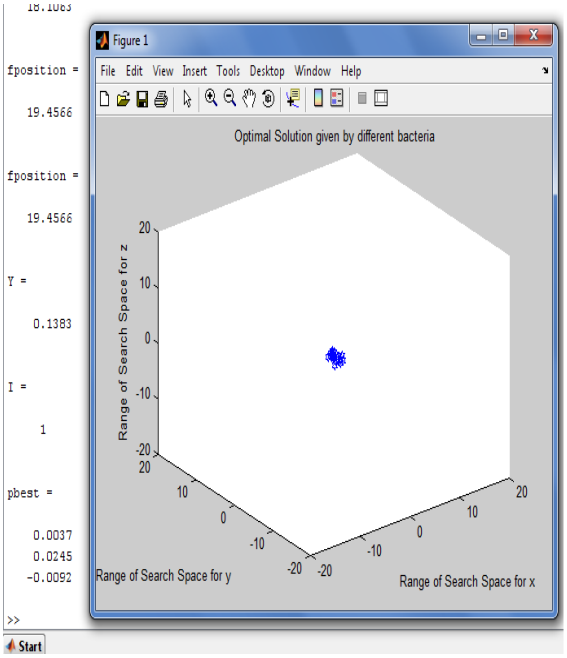

Fig 11. For step size C(i) = 0.1

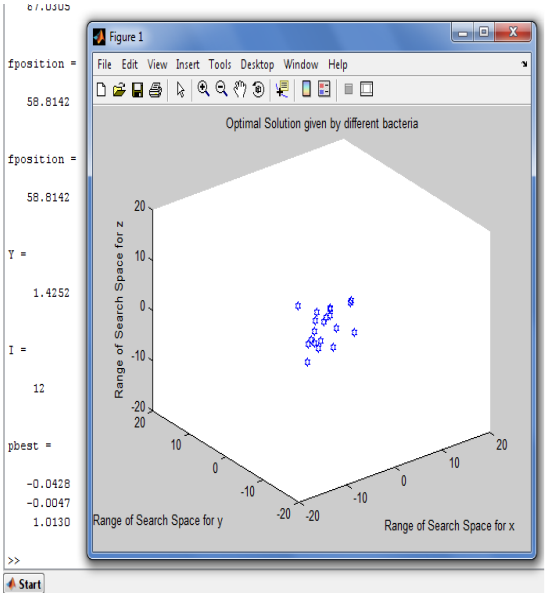

Fig 12. For step size C(i) = 0.5

4. DISCUSSION ON THE RESULTS

Above figures depicting optimal results for different step sizes and table 3 clearly show that bacteria with larger step size explore more search space as compared to bacteria with smaller step sizes. However, they can, sometimes, miss the best solution by taking very large steps which can be observed from the values of best solution for largest step size as shown in table 3. Smaller step sizes give better results by exploring the search space minutely. Limitation of taking small step size is that sometimes bacteria may get stuck into local optima as shown for function f_3 (Rastrigin Function) in table 3. In these cases, it may not be able to reach at global optima for its entire life time.

From above discussion it is clear that bacteria with larger step size would move in the entire search space while the bacteria with smaller step sizes would do fine search around local optimal solutions. Hence, chemotactic operator (i.e. the step size) should be chosen so as to allow the bacteria to explore the entire search space as well as to search effectively around the potential solutions.

Table3: Best solution for different step sizes

Function	Optimum position (Global minima) for different Step Sizes		
	C(i) = 0.05	C(i) = 0.1	C(i) = 0.5
f_1(x)	3.90	3.59	3.43
f_2(x, y)	0.0083, 0.0058	-0.0256, 0.0400	0.1866, -0.3083
f_3(x, y, z)	0.0182, -0.0369, -0.0124	0.0037, 0.0245, -0.0092	-0.0428, -0.0047, 1.0130

5. CONCLUSION

This paper has analyzed the effect of varying step size on chemotactic movement and hence foraging behavior of bacteria. It has been observed that faster chemotactic movement (i.e. larger step size) results in a wider search of given search space contrary to slower chemotactic movement (i.e. smaller step size) which performs comparatively narrow or minute search. From these observations, we conclude that Chemotaxis is highly dependent on step size. We further conclude that while deciding the step size of bacteria, great care should be taken such that bacteria may be able to search for the potential solution as effectively as they can. Depending on the nature of search space, bacteria can be made self adaptive in deciding the value of the step size. Effect of cell to cell attractant and repellant can also be included.

6. REFERENCES

[1] C. A. Floudas, "Deterministic Global Optimization: Theory, Methods and Applications", vol. 37 of *Nonconvex Optimization and Its Applications*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.

[2] J. H. Holland, "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to B, control, and Artificial Intelligence", *University of Michigan Press, Ann Arbor*, Michigan, USA, 1975.

[3] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics. Part B*, vol. 26, no. 1, pp. 29–41, 1996.

[4] Kennedy, J., & Eberhart, R. C., "Particle swarm optimization" in Proceedings of IEEE international conference on neural networks, Piscataway, NJ, pp. 1942–1948, (1995).

[5] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Syst. Mag.*, vol. 22, no. 3, pp. 52–67, Jun. 2002.

[6] Sotirios P. Chatzis, Spyros Koukas, "Numerical optimization using synergetic swarms of foraging bacterial populations", *Elsevier Expert Systems with Applications*, vol. 38, Issue 12, pp. 15332-15343, Nov.-Dec.,2011

[7] Hanning Chen, Yunlong Zhu, and Kunyuan Hu, "Adaptive Bacterial Foraging Optimization", Hindawi Publishing Corporation, *Abstract and Applied Analysis*, Volume 2011, Article ID 108269, 27 pages, doi:10.1155/2011/108269.