A Systematic way for Image Segmentation based on Bacteria Foraging Optimization Technique
(Its implementation and analysis for image segmentation)

Amrinder Singh

Assistant Professor
Akal Degree College,Mastuana,
Sangrur(Punjab,India)

Sonika Jindal

Assistant Professor
Shaheed Bhagat Singh College of Engg. & Tech,
Ferozepur(Punjab,India)

Abstract- **Bacterial foraging optimization algorithm (BFOA) has been widely accepted as a global optimization algorithm of current interest for distributed optimization and control. BFOA is inspired by the social foraging behavior of Escherichia coli. BFOA has already drawn the attention of researchers because of its efficiency in solving real-world optimization problems arising in several application domains. The purpose of this paper is to detect the image segmentation in an efficient way using Bacterial Foraging Optimization which is an optimization technique inspired from E. coli bacteria and does not require any threshold values in image segmentation. Hence, this technique helps to reduce the computational complexity and time consuming. It is also demonstrated in this study that this method can rectify the shortage perfectly, and obtain a more ideal edge image.**

I. INTRODUCTION

Image segmentation is often described as the process that subdivides an image into its constituent parts and extracts those parts of interest (objects). It is one of the most critical tasks in automatic image analysis because the segmentation results will affect all the subsequent processes of image analysis, such as object representation and description, feature measurement and even the following higher level tasks such as object classification and scene interpretation[3] .It basically implies partitioning a digital image into different parts or objects regions corresponding to individual surfaces, objects, or natural parts of objects bacteria foraging optimization technique for segmentation is an active area of research and new optimization methods are being developed continuously. Genetic Algorithm (GA), since 1960, has been playing a dominant role in the world of optimization. However, in the year 1995, a population-based optimization method was developed by Eberhart and Kennedy, which is called Particle Swarm Optimization (PSO). PSO [7] has been inspired by social behavior of bird flocking or fish schooling and can efficiently handle any arbitrary optimization problem. PSO has been successfully applied in many areas like function optimization, training of artificial neural network, fuzzy system control, image denoising. Researchers have proposed new PSO variants to improve speed and accuracy. Some of the PSO variants are linear decreasing weight particle swarm optimization (LDWPSO) , gregarious

particle swarm optimization (GPSO) and self - organizing hierarchical particle optimizer with time varying acceleration coefficients (HPSO-TVAC) . In 2002, bacterial foraging optimization (BFO) was proposed by K. M. Passino. Very few researchers have tried to improve the quality of optimal solution over multimodal and high dimensional functions using BFO. Researchers have tried to improve quality of solution by fusion of BFO with PSO and GA. The BFO was hybridized with PSO (known as BSO) for multimodal and high dimensional functions.

II. LITERATURE SURVEY

In 2002, Passino [2] proposed Bacterial Foraging Optimization Algorithm (BFOA) for distributed optimization and control. BFA is based on the foraging behaviour of Escherichia Coli (*E. Coli*) bacteria present in the human intestine and already been in use to many engineering problems including multiple robot co-ordination. According to paper, BFA is better than Particle Swarm Optimisation in terms of convergence, robustness and precision.

Dhariwal et al. (2004) presented an approach, inspired by bacterial chemo-taxis, for robots to navigate to sources using gradient measurements and a simple actuation strategy (biasing a random walk). They have showed the efficacy of the approach in varied conditions including multiple sources, dissipative sources, and noisy sensors and actuators through extensive simulations.

Datta et al. (2008) proposed an improved adaptive approach involving Bacterial Foraging Algorithm (BFA) to optimize both the amplitude and phase of the weights of a linear array of antennas for maximum array factor at any desired direction and nulls in specific directions.

Chen et al. (2009) in the paper entitled "An Improved Bacterial Foraging Optimization", presented a modified method to improve the classic BFO, called as **iBFO** in which both of search scope and chemotaxis step varies dynamically, which can markably accelerate the convergence and enhance the searching precision.

Zhang et al.(2010) [4] in the paper entitled "Bacterial Foraging Optimization Based Neural Network for Short-term Load Forecasting", purposed bacterial foraging optimization technique to train the neural networks. BFO is a novel and powerful global search technique, and it can find the weights/biases of the neural network quickly and accurately.

Sunil Kumar, Vijay Kumar Sharma, Somya Saxena et al.(2013) [8] Edge detection is a primary operation of most of the image processing applications such as image detection, boundary detection, image classification, image registration. Edge detection filters out less important information and preserve the structural properties of image. The Proposed technique uses ANFIS edge detector for edge detection on digital images. It involves a neuro fuzzy system with the learning capability of neural network and the advantages of rule-based fuzzy system. This work follows hybrid algorithm to resolve the edge detection issues with the help of least square method and gradient descent method.

III. OBJECTIVE OF WORK
1. To study the bacterial forging optimization technique.
2. The algorithm for optimization technique Bacterial foraging optimization has been implemented using MATLAB software .
3. This algorithm is then applied on the image.

IV. THE BACTERIA FORAGING OPTIMIZATION ALGORITHM

During foraging of the real bacteria, locomotion is achieved by a set of tensile flagella. Flagella help an *E.coli* bacterium to tumble or swim, which are two basic operations performed by a bacterium at the time of foraging. When they rotate the flagella in the clockwise direction, each flagellum pulls on the cell. That results in the moving of flagella independently and finally the bacterium tumbles with lesser number of tumbling whereas in a harmful place it tumbles frequently to find a nutrient gradient. Moving the flagella in the counterclockwise direction helps the bacterium to swim at a very fast rate. In the above-mentioned algorithm the bacteria undergoes chemotaxis, where they like to move towards a nutrient gradient and avoid noxious environment. Generally the bacteria move for a longer distance in a friendly environment. Figure 1 depicts how clockwise and counter clockwise movement of a bacterium take place in a nutrient solution.
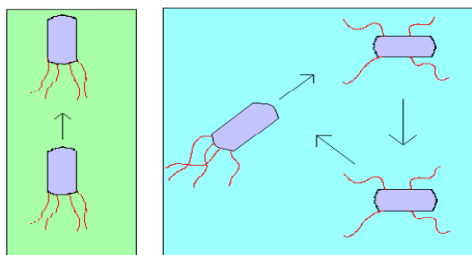

Figure 1. Swim and tumble of a bacterium

When they get food in sufficient, they are increased in length and in presence of suitable temperature they break in the middle to from an exact replica of itself. This phenomenon inspired Passino to introduce an event of reproduction in BFOA. Due to the occurrence of sudden environmental changes or attack, the chemotactic progress may be destroyed and a group of bacteria may move to some other places or some other may be introduced in the swarm of concern. This constitutes the event of elimination-dispersal in the real bacterial population, where all the bacteria in a region are killed or a group is dispersed into a new part of the environment.

V. E. COLI BACTERIAL SWARM FORAGING
- Chemotaxis
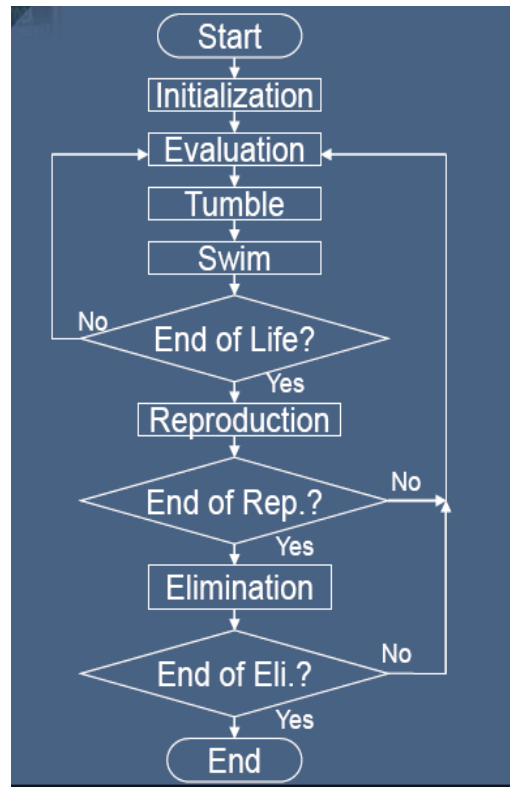- Swarming
- Reproduction
- Elimination and dispersal

Fig.2 Flowchart of bacterial foraging

- p: Dimension
- S: Population size
- Nc: Chemotaxis steps
- NS: Swim length limitation
- Nre: Reproduction steps
- Ned: Elimination-dispersal steps
- Ped: Elimination-dispersalrate
- random number on [-1,1],where i from 1 to p.
- $c(i)$: Step size

VI. PROPOSED ALGORITHM FOR THE IMPLEMENTATION OF THE BFOA

We have used bacterial forging optimization algorithm for segmentation of the image. This algorithm shows how the bacterial based theory is used to detect the image. natural selection tends to eliminate animals with poor "foraging strategies" (methods for locating, handling, and ingesting food) and favor the propagation of genes of those animals that have successful foraging strategies since they are more likely to enjoy reproductive success (they obtain enough food to enable them to reproduce). After many generations, poor foraging strategies are either eliminated or shaped into good ones (redesigned). Logically, such evolutionary principles have led scientists in the field of "foraging theory" to hypothesize that it is appropriate to model the activity of foraging as an optimization process:

A. Steps:

Initialize Parameters: *p, S, Nc, Ns, Nre* and *C (i)*, *i* = 1, 2… S

Where,
p = Dimension of search space
S = Number of bacteria in the population
Nc = Number of chemotaxis steps
Ns= Number of swimming steps
Nre = Number of reproduction steps
C (i) = Step size taken in the random direction specified by the tumble
J (i, j, k) = Fitness value or cost of *i-th* bacteria in the *j-th* chemotaxis and *k-th* reproduction steps
θ (i, j, k)= Position vector of *i-th* bacterium in *j-th* chemotactic step and *k-th* reproduction steps
Jbest (j, k) = Fitness of best position in the *j-th* chemotaxis and *k-th* reproduction steps
Jglobal= Fitness value or cost of the global best position in the entire search space

Step 1: Update the following parameters: *J (i, j, k), Jbest (j, k)* and *Jglobal= Jbest (j, k)*

Step 2: Reproduction Loop: *k = k+1*

Step 3: Chemotaxis loop: *j = j+1*

a) Compute fitness function *J (i, j, k)* for *i= 1, 2, 3...S*
b) Update *Jbest (j, k)*.
c) Tumble: Generate a random vector *Δ(i)* ∈R^p with each element Δ_m (i) *m = 1, 2...p*, a random
d) Compute *q* for *i = 1, 2 …….…..S*
e) Swim

i) Let *m* = 0 (counter for swim length)
ii) While *m < Ns* (if have not climbed down too long.

• Let *m = m+1*
• Compute fitness function *J (i, j+1, k)* for *i = 1, 2... S*
• Update *Jbest (j+1, k)*

• If *Jbest (j+1,k)< Jbest (j, k)* (if doing better),

Jbest (j, k) = Jbest (j+1, k)
Compute *θ* for *i = 1, 2…S* [Synchronous position updation]
Use this θ *(i*, j +1, *k)* to compute the new *J (i, j+1, k)*
• Else, let *m = Ns*. This is the end of the while statement

Sub Step f) Mutation Operator
Compute θ for *i = 1, 2….S* [Synchronous position updation by mutation operator]

Step 4: If *j < Nc*, go to step 3. In this case, continue chemotaxis since the life of bacteria is not over.

Step 5: Reproductions: The *Sr=S/2* bacteria with the highest cost function values die and other *Sr* bacteria with the best values split. Update *Jglobal* from *Jbest (j, k)*.
Step 6: If *k < Nre*, go to Step 2 otherwise end.

VII. EXPERIMENTAL RESULTS
VIII.

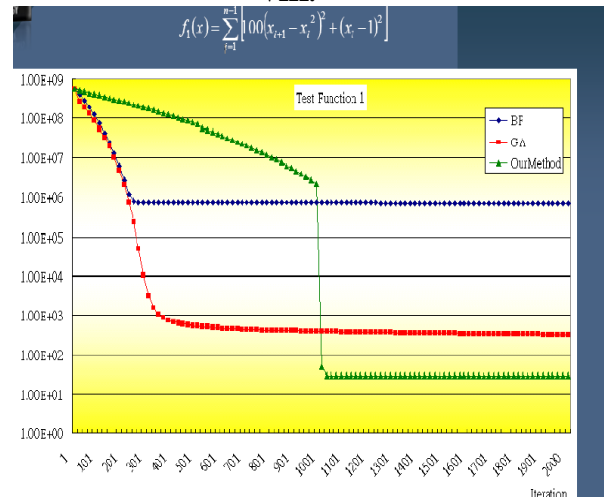

Fig.3 Experimental Results

IX. APPLICATION AREAS OF BFOA

➤ Automatic circle detection using BFOA.
➤ Automatic generation control using BFO.
➤ Calculate resonant frequency using BFO
➤ Economic emission load dispatch through fuzzy BFOA.
➤ Image compression using BFOA based ANN.

X. CONCLUSION

We review several optimization approaches based on bacterial foraging in the above.

➤ The bacterial foraging and its improved algorithm that can be applied. into plenty fields to deal with optimization problems.
➤ The exploration of the potential of BFOAs could be greatly explored in the future.

REFERENCES

1. A.D. Jepson and D.J. Fleet, "Image Segmentation" , Internet Draft 2007.
2. Passino, K. M., (2002), *"Biomimicry of bacterial foraging for distributed optimization and control,"* Control Systems Magazine, vol. 22, no. 3, pp. 52-67.
3. A.D. Jepson and D.J. Fleet, "Image Segmentation" , Internet Draft 2007.
4. Saadi, S., Kouzou, A., Guessoum, A., and Bettayeb, M., (2010), *"Bacterial Foraging Algorithm for Neutron Radiography Image Quality Improvement"* 7th International Multi-Conference on Systems, Signals and Devices.
5. W.J.Tang, Q.H.Wu, J.R.Saunders,"Bacterial Foraging Algorithm for Dynamic Environments", IEEE Congress on Evolutionary Computation, Canada, pp. 1324-1330, July 16-21, 2006.
6. Kennedy, J .and Eberhart, R. C., "Particle Swarm Optimization", Proceedings of IEEE International Conference on Neural Networks IV, Piscataway, pp. 1942 – 1948, 1995.
7. Y.Shi and R.Eberhart, "Emperical Study of Particle Swarm Optimization", Proceedings of the IEEE Congress on Evolutionary Computation(CEC'99), pp. 1945-1950, 1999.
8. Sunil Kumar, Vijay Kumar Sharma, Somya Saxena et al. ,*"Edge detection using Soft Computing in MATLAB"* International Journal of Advanced Research in Computer Science and Software Engineering(ISSN: 2277 128X) Vol. 3- Issue- 7