CHAPTER 5

BACTERIAL FORAGING OPTIMIZATION ALGORITHM (BFOA)

5.1 INTRODUCTION

Nowadays, the term efficiency is necessarily present in every engineer's vocabulary. Concepts such as performance and cost cannot be neglected in a competitive society such as ours. Minimizing cost and/or maximizing performance can be considered as an optimization problem. So that, to optimize is to find the best solution to a certain designated problem. Every method has a set of problems to which it is more indicated. This depends on a series of problem characteristics, specially the function describing it, is not easily obtainable. Therefore, a good general understanding of the problem and of optimization method is needed.

Deterministic methods are based on the calculation of functions, derivatives or approximations of it. The objective is then to find where the function gradient is null using the direction to where it points. Stochastic methods search for the optimal solution of a certain problem using an "oriented random" approach. In a large number of problems, there exists a need to find optimal solutions due to more than one characteristic. In this case, a multi-objective approach must be employed. Engineering problems demanding low cost, high performance and low loss are examples of application where this approach is needed. It becomes clear that the knowledge on the nature of the problem is a requirement, allowing the choice of a more suitable optimization tool for power system optimization.

There are many optimization algorithms for finding the optimal rating of FACTS devices. They are categorized as follows:

- Genetic Algorithms
- Simulated Annealing
- Ant Colony Optimization
- Particle Swarm Optimization
- Differential Evolutionary Algorithm
- Tabu Search
- Fuzzy Logic Control
- Artificial Neural Networks and
- Bacterial Foraging Optimization Algorithm

Among the above mentioned optimization algorithms, Bacterial Foraging Optimization Algorithm (BFOA) (Kevin Passino 2002) is proposed in this work to find the optimal rating of FACTS devices to achieve minimization of real power loss, FVSI, L-index, minimization of total cost, voltage profile improvement and enhancement of voltage stability. The rationale behind selecting Bacterial Foraging Algorithm is that this algorithm is not largely affected by the size and non-linearity of the problem. Also this algorithm has converged to the optimal solution in many problems where the most analytical methods have failed to converge. This algorithm also has advantages such as less computational burden, global convergence, less computational time requirement and can handle more number of objective function when compared to the other evolutionary algorithms.

5.2 BACTERIAL FORAGING OPTIMIZATION ALGORITHM (BFOA)

Bacterial Foraging Optimization Algorithm (BFOA) is proposed by Kevin Passino (2002), is a new comer to the family of nature inspired optimization algorithms. Application of group foraging strategy of a swarm of *E.*coli bacteria in multi-optimal function optimization is the key idea of this new algorithm. Bacteria search for nutrients is a manner to maximize energy obtained per unit time. Individual bacterium also communicates with others by sending signals. A bacterium takes foraging decisions after considering two previous factors. The process, in which a bacterium moves by taking small steps while searching for nutrients, is called chemotaxis. The key idea of BFOA is mimicking chemotactic movement of virtual bacteria in the problem search space.

p : Dimension of the search space,

S : Total number of bacteria in the population,

N_c : The number of chemotactic steps,

N_s : The swimming length.

N_{re} : The number of reproduction steps,

N_{ed} : The number of elimination-dispersal events,

P_{ed} : Elimination-dispersal probability,

C(i): The size of the step taken in the random direction specified by the tumble.

Foraging theory is based on the assumption that animals search for and obtain nutrients in a way that maximizes their energy intake *E* per unit time T spent foraging. Hence, they try to maximize a function like E/T (or they maximize their long-term average rate of energy intake). Maximization of such a function provides nutrient sources to survive and additional time for other important activities (e.g., fighting, fleeing, mating, reproducing, sleeping, or shelter building). Shelter building and mate finding activities sometimes bear similarities to foraging. Clearly, foraging is very different for different species. Herbivores generally find food easily but must eat a lot of it. Carnivores generally find it difficult to locate food but do not have to eat as much since their food is of high energy value. The "environment" establishes the pattern of nutrients that are available (e.g., via what other organisms are nutrients available, geological constraints such as rivers and mountains and weather patterns) and it places constraints on obtaining that food (e.g., small portions of food may be separated by large distances). During foraging there can be risks due to predators, the prey may be mobile so it must be chased and the physiological characteristics of the forager constrain its capabilities and ultimate success. Bacterial Foraging optimization theory is explained by following steps.

- Chemotaxis
- Swarming
- Reproduction and
- Eliminational-Dispersal

5.2.1 Chemotaxis

This process simulates the movement of an E.coli cell through swimming and tumbling via flagella. Biologically an E.coli bacterium can move in two different ways. It can swim for a period of time in the same direction or it may tumble and alternate between these two modes of operation for the entire lifetime. Suppose $\theta^i(j, k, l)$ represents i^{th} bacterium at

j^{th} chemotactic, k^{th} reproductive and l^{th} elimination-dispersal step. $C(i)$ is the size of the step taken in the random direction specified by the tumble (run length unit). Then in computational chemotaxis the movement of the bacterium may be represented by

$$\theta^{i}(j+1,k,l) = \theta^{i}(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^{T}(i)\Delta(i)}} \qquad (5.1)$$

Where Δ indicates a vector in the random direction whose elements lie in [-1, 1].

5.2.2 Swarming

An interesting group behavior has been observed for several motile species of bacteria including E.coli and S. Typhimurium, where intricate and stable spatio-temporal patterns (swarms) are formed in semisolid nutrient medium. A group of E.coli cells arrange themselves in a traveling ring by moving up the nutrient gradient when placed amidst a semisolid matrix with a single nutrient chemo-effecter. The cells when stimulated by a high level of succinate, release an attractant aspertate, which helps them to aggregate into groups and thus move as concentric patterns of swarms with high bacterial density.

5.2.3 Reproduction

The least healthy bacteria eventually die when each of the healthier bacteria (which yielding lower value of the objective function) asexually split into two bacteria, which are then placed in the same location. This keeps the swarm size constant.

5.2.4 Elimination and Dispersal

Gradual or sudden changes in the local environment where a bacterium population lives may occur due to various reasons. Events can occur such that all the bacteria in a region are killed or a group is dispersed into a new part of the environment. For example, a significant local rise of temperature may kill a group of bacteria that are currently in a region with a high concentration of nutrient gradients. Events can take place in such a fashion that all the bacteria in a region are killed or a group is dispersed into a new location. Over long periods of time, such events had spread various types of bacteria into every part of our environment from our intestines to hot springs and underground environments. To simulate this phenomenon in BFOA some bacteria are liquidated at random with a very small probability while the new replacements are randomly initialized over the search space.

Elimination and dispersal events have the effect of possibly destroying chemotactic progress, but they also have the effect of assisting in chemotaxis, since dispersal may place the bacteria near good food sources. From a broad perspective, elimination and dispersal are parts of the population-level long-distance motile behavior.

5.3 APPLICATION OF BACTERIAL FORAGING OPTIMIZATION ALGORITHM FOR PROPOSED METHOD

5.3.1 Algorithm

Step 1: Initialize the parameters S, N_c, N_s, N_{re}, N_{ed}, P_{ed} and the C (i), (i =1, 2,…, S).

Choose the initial value for the θ^i, i=1,2….S. These must be done in areas where an optimum value is likely to exist. The control variables (θ^i) are reactive power (for SVC) and line reactance (for TCSC). They are randomly distributed across the domain of the optimization space. After computation of θ is completed, the value of P (position of each member in the population of the S bacteria) is updated automatically and termination test is done for maximum number of specified iterations.

Step 2: Elimination-Dispersal loop: $l= l+1$

Step 3: Reproduction loop: $k= k+1$

Step 4: Chemo taxis loop: $j= j+1$

(i) For $i= 1, 2,\ldots,S$ take a chemo tactic step for bacterium 'i' as follows:

(ii) Compute cost $J(i,j,k,l)$.

(iii) Let $J(i,j,k,l)=J(i,j,k,l)+J_{cc}(\theta^i (j,k,l),\mathrm{P}(j,k,l))$

(iv) Let $J_{last} = J\ (i,j,k,l)$ to save this value since find better cost via a run

(v) 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]. Where R is a real number.

(vi) Move let

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

This results in a step of size C (i) in a direction of the tumble for bacterium i

(vii) Compute $J(i,j+1,k,l)$.

The load flow analysis using N-R method is carried out. The values of FVSI or L-index, real power loss and the total cost are calculated. If the cost function or loss is minimum then next step can be carried out else go to step (iii)

(viii) Swim.

(a) Let m=0 (counter for swim length)

(b) While $m<N_s$

Let m=m+1

If $J(i,j+1,k,l)<J_{last}$ (if there is improvement), let $J_{last} = J(i,j+1,k,l)$

and let $\theta^i(j+1,k,l) = \theta^i(j+1,k,l) + C(i)\dfrac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}$ and use this

$\theta^i(j+1,k,l)$ to compute the new $J(i,j+1,k,l)$.

Else, let $m=N_s$. End of while statement

(ix) Go to next bacterium $(i+1)$ if $i \neq S$

Step 5: If $j<N_c$ go to step 3. In this case, continue chemo taxis, since the life of the bacteria is not over.

Step 6: Reproduction

a) For the given k and l, and for each i=1,2,….,S, let $J^i_{health} = \sum_{j=1}^{Nc+1} J(i,j,k,l)$ be the health of bacterium i. Sort bacteria and chemo tactic parameter C(i) in order of ascending cost J_{health}.

(b) The S_r bacterium with the highest J_{health} values die and the other S_r bacteria with the best values split.

Step 7: If $k< N_{re}$, go to step 2. In this case we have not reached the number of specified reproduction steps.

Step 8: Elimination-Dispersal

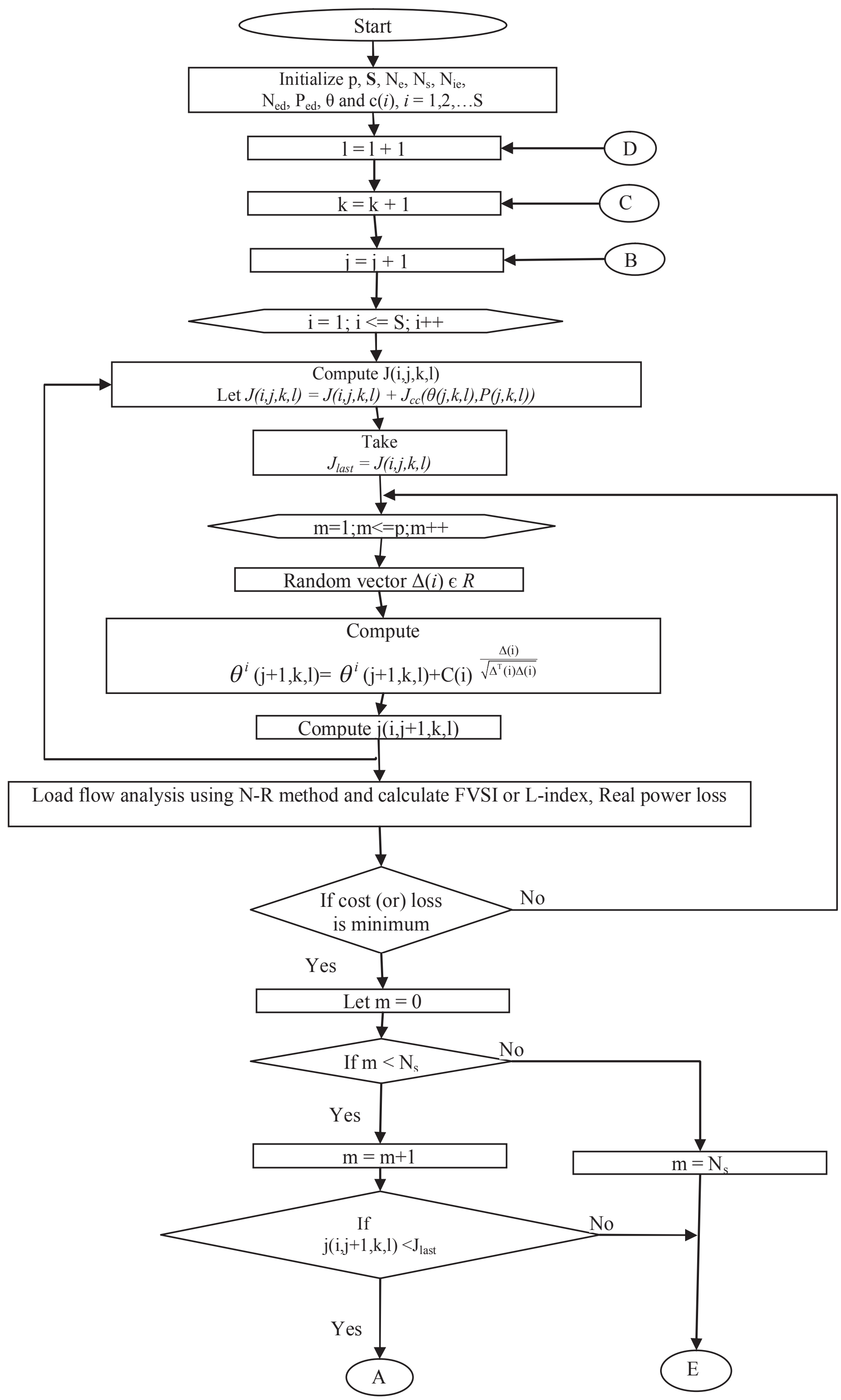
For i=1,2,…..,S with probability $P_{ed,}$ eliminate and disperse each bacterium. Eliminate a bacterium and disperse one to a random location on the optimization domain. If $l < N_{ed,}$ then go to step 1, otherwise end.

The FVSI, L-index, real power loss, total cost and bus voltages are also obtained separately. The parameter of the Bacterial Foraging Optimization Algorithm is given in Table 5.1 and the flowchart for the proposed algorithm is shown in Figure 5.1.

Table 5.1 Control Parameters of Bacterial Foraging Optimization Algorithm

Sl.No	Parameters	Values
1	Number of bacterium, S	50
2	Maximum number of steps, N_s	4
3	Number of chemo tactic steps, N_c	100
4	Number of reproduction steps, N_{re}	4
5	Number of elimination-dispersal steps, N_{ed}	2
6	Probability, P_{ed}	0.25
7	Size of the step, C(i)	0.1

5.3.2 Flowchart



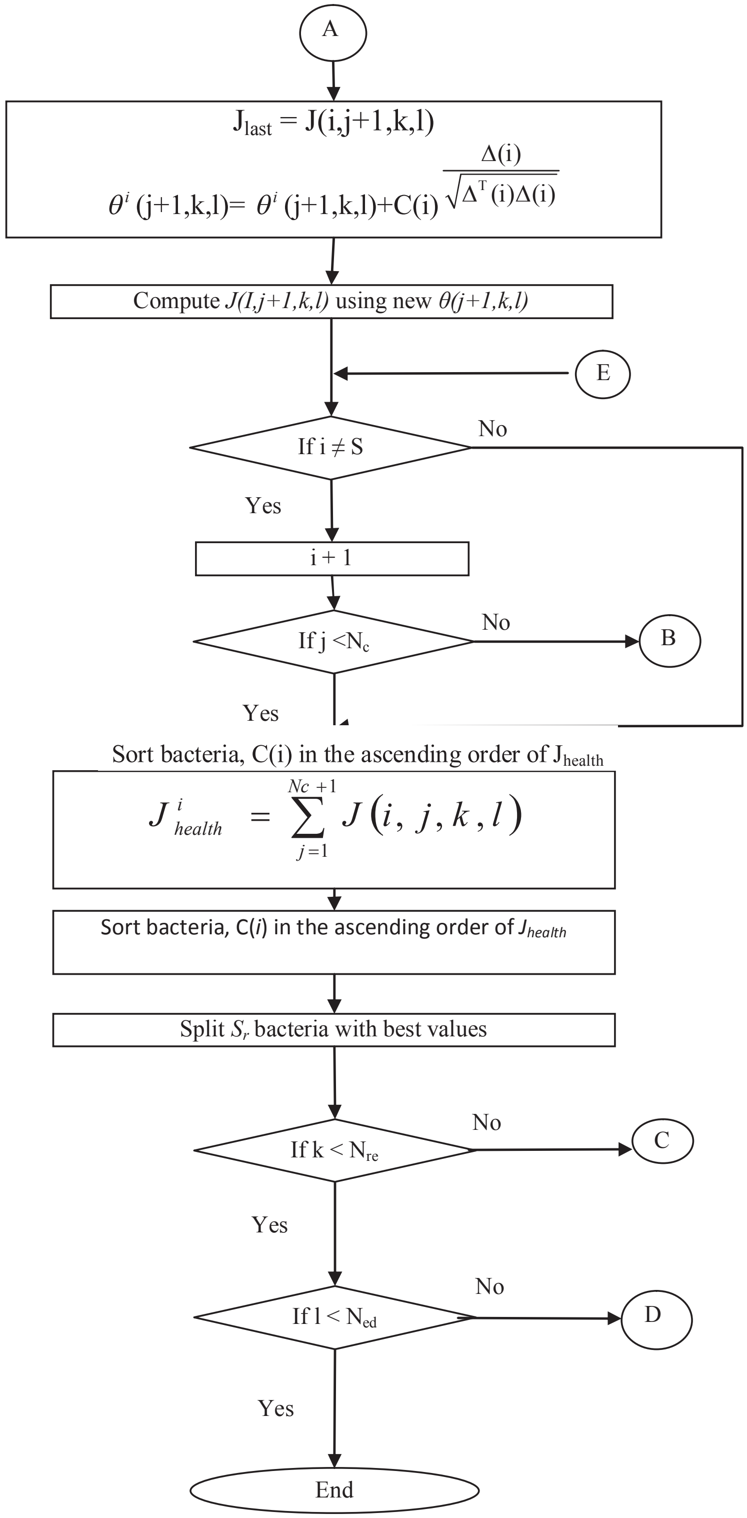


Figure 5.1 Flowchart of Proposed Bacterial Foraging Optimization Algorithm

5.4 CONCLUSION

In this chapter, Bacterial Foraging Optimization Algorithm used for finding out the optimal rating of FACTS devices is described. The Bacterial Foraging Optimization technique has gained popularity in solving optimization problems. The algorithm and flowchart for the proposed work is explained and control parameters used are tabulated. Bacterial Foraging Algorithm is based on a computational intelligence technique that is not largely affected by the size and non-linearity of the problem and has converged to the optimal solution to many problems where the most analytical methods fail to converge and also has its advantages such as less computational burden, global convergence, less computational time requirement and can handle more number of objective function.