A Review of Bacterial Foraging Optimization and Its Applications

Vipul Sharma
E&C Engg Dept, Gurukul Kangri University,Haridwar, India

S.S. Pattnaik
NITTTR, Chandigarh

Tanuj Garg
E&C Engg Dept, Gurukul

ABSTRACT
Recently, germ intelligence has grabbed prime focus of research fraternity working on optimization and many such powerful algorithms have been reported till date. Of this, Bacterial foraging optimization algorithm (BFOA) has attracted a lot of attention as a high performance optimizer because of its faster convergence and global search approach. Since its inception in 2001, many variants of BFOA have come up leading to even faster convergence with higher accuracy. This paper presents an application based review of such variants and will be useful for new researchers exploring its use in their research problems.

Keywords
Optimization, CAD, germ intelligence, BFOA.

1. INTRODUCTION
In a pioneering paper, written in 2002, Prof. K.M. Passino introduced an optimization technique known as Bacterial Foraging Optimization Algorithm (BFOA) based on the social foraging behaviour of Escherichia Coli (E. Coli) bacteria present in human intestine(Passino, 2002). Since inception, the BFOA has drawn attention of researchers as a high performance optimizer and many successful applications of BFOA in optimal control engineering (Passino, 2002;Korani, 2008), image processing [8], network scheduling [11], electric load forecast [13] etc have been reported till date. BFOA coupled with method of moment (MOM) has also been used in antenna applications [2-5]. Many improved application based variants of BFO [1,4,6,7,10] have also come up leading to drastic reduction in convergence time and with higher accuracy. But complete potential of BFOA optimizer is yet to be explored. The main objective of this review paper is to discuss basic BFOA, different variants of it and their applicability in engineering problems.

2. THE BASIC BACTERIA FORAGING ALGORITHM
The Bacteria Foraging is an evolutionary algorithm which estimates cost function after each iterative step of the program as the program execution proceeds and leads to progressively better fitness (less cost function). The parameters to be optimized represent coordinates (position) of the bacteria. The parameters are discretized in the desirable range, each set of these discrete values represent a point in the space coordinates. Then one bacteria is positioned (created) at each point. After each progressive step the bacteria move to new positions (new coordinate values) and at each position cost function is calculated and then, with this calculated value of cost function, further movement of bacteria is decided by decreasing direction of cost function. This finally leads the bacteria to a position (set of optimization parameters) with highest fitness.

The foraging strategy of E. Coli. Bacteria is governed by four processes. These are chemotaxis, swarming, reproduction and elimination and dispersal. Chemotaxis is achieved by swimming and tumbling. When the bacterium meets favourable environment (rich in nutrients and noxious free), it continues swimming in the same direction. Decrease in cost function represents favourable environment, while increase in cost function represents unfavourable environment. When it meets unfavourable environment it tumbles (changes direction). In swarming, the bacteria move out from their respective places in ring of cells by bringing mean square error to the minimal value.

Chemo taxis and Swarming
Suppose m1,m2,m3 are the parameters to be optimized. They will represent axis of space coordinates (like x,y,z axis in rectangular coordinate system). Now let fi,j,k,l(m1,m2,m3) represents position of ith bacterium at a point in m1,m2,m3 coordinate system, in jth Chemotaxis, kth reproduction and lth elimination and dispersal step. Also let C(i) represents unit run-length of a bacterium. Then movement of ith bacterium in jth chemotaxis step can be represented by following equation.

$$\mathrm{f}^{i,j+1,k,l}(\mathrm{m1},\mathrm{m2},\mathrm{m3}) = \mathrm{f}^{i,j,k,l}(\mathrm{m1},\mathrm{m2},\mathrm{m3}) + \mathrm{C}(i)\frac{\mathrm{del}(i)}{\sqrt{\mathrm{del}^{\mathrm{T}}(i)\ \mathrm{del}(i)}} \quad \ldots\ldots\ldots\ldots\ldots..(1)$$

where del(i) is three elements direction vector (because position of bacteria being represented by three coordinates which are optimization parameters). Each element of del (i) is a random number lying between [-1, 1]. If ith bacterium meets favourable environment which is represented by less value of cost function at that point in space coordinates, it swims which means direction vector will remain same as was in previous (j-1th) chemotaxis step. Otherwise, del (i) is assigned a new value which is a random number lying between [-1,1].

After each chemotaxis step, the bacteria move and reach new points in space (whose coordinate axis are optimization parameters). Each point represents a set of optimization parameters. Here, at these present locations, fitness of each bacterium is evaluated which further decides next movement of the bacterium. Fitness of ith bacterium is represented by

Cost function Pi,j,k,l. Better fitness mean less value of Cost function.

Reproduction

Health status (fitness) of each bacterium is calculated after each complete chemotaxis process. It is overall sum of cost function$\sum_{j=1}^{Nc} \mathrm{P}^{i,j,k,l}$, where NC is total number of steps in a complete chemotaxis process. Locations of healthier bacteria represent better sets of optimization parameters. Then, to further speed up and refine the search, more number of bacteria is required to be placed at these locations in the optimization domain. This is done in reproduction step. Healthiest half of bacteria (with minimum value of cost function) are let to survive, while the other half of them die. Each surviving bacterium splits up into two and these two are placed at the same location. In this way population of bacteria remains constant.

Elimination and Dispersal Event

The chemotaxis process performs local search and reproduction speeds up convergence of search parameters. But, chemotaxis and reproduction may not be enough to reach the global minimum point (best optimized set of parameters). The bacteria may also get trapped in local minima assuming it to be the best fitness position in the surrounding patch. To avoid this to happen, elimination and dispersal event is performed. The bacterium having probability Ped (probability of elimination and dispersion) is eliminated from present location and one bacterium is placed (dispersion) at a random location so as to realize global search. The population of bacteria still remains constant.

Following is the step by step procedure of Bacterial Foraging Algorithm:

Initialize parameters

D = Dimension of search. It is number of parameters to be optimized. If you have three parameters to be optimized, say m1,m2,m3, then D will be equal to three.

B= Number of bacteria in the population. It should be equal to number of sets of points obtained by discretizing the optimization parameter. Suppose m1,m2,m3 each parameter is discretized to give ten values in range [1, 2]. Then each set will represent a point in space (m1,m2,m3-coordinates). Hence there will be ten points (locations) in the optimization domain. So, ten bacteria are required to be placed at these points to start the search.

NC = Number of chemotaxis steps a bacterium has to move in a complete chemotaxis procedure before going for reproduction.

Ns = Number of swimming steps

Nre = Number of reproduction steps

Ned = Number of elimination and dispersal steps

Ped = Elimination and dispersal probability

C(i) = Unit run-length

fi,j,k,l (m1,m2,m3)= Position vector of ith bacterium, in jth chemotaxis step, in kth reproduction step and lth elimination and dispersal step at a point in m1,m2,m3-coordinates for assumption given above.

Step 1: Elimination and dispersal loop l = l+1

Step 2: Reproduction loop l = l+1

Step 3: Chemotaxis loop j = j+1

For i= 1,2,3.....B, a chemotaxis step for ith bacterium will be as follows:

Calculate fitness function Pi,j,k,l.

Save this value in Plast= Pi,j,k,l so that we can find better fitness (cost) via run.

Tumble: Generate direction vector del (i) is assigned a new value which is a random number lying between [-1,1].

Move using equation (1)…

$$\mathrm{f}^{i,j+1,k,l}(\mathrm{m1},\mathrm{m2},\mathrm{m3}) = \mathrm{f}^{i,j,k,l}(\mathrm{m1},\mathrm{m2},\mathrm{m3}) + \mathrm{C}(i)\ \frac{\mathrm{del}\ (i)}{\sqrt{\mathrm{del}^{\mathrm{T}}\ (i)\ \ \mathrm{del}\ (i)}}$$

Calculate fitness function Pi,j,k,l

Swim : (i) Initialize swim counter sc = 0.

(ii) If sc< Ns

If Pi,j,k,l < Plast , Let Plast= Pi,j,k,l, and use equation (1) given in step e) to move in the same direction.

Use the new generated location fi,j,k,l for new values of m1,m2,m3 to calculate Pi,j,k,l and continue in the loop.

Else sc=Ns

Do the same process for next bacterium i=i+1, go step b) if i≠S.

Step 4: If j< Nc, go to step 3 for next chemotaxis step as the chemotaxis process not complete.

Step 5: Reproduction. With current values of k,l, compute overall fitness (cost function) $\sum_{j=1}^{Nc} \mathrm{P}^{i,j,k,l}$ for each ith bacterium and sort the fitness in descending order. Higher value of cost function means less fitness.

Step 6: Half of the bacteria with less fitness will die and the other half will reproduce. They will split into two and placed at the same locations of their parents. So, population remains constant.

Step 7: If k< Nre, go to step 2. Increment the reproduction counter and start new chemotaxis process.

Step 8: Elimination-dispersion. Eliminate the bacterium with probability Ped and disperse one at a random location in the optimization space.

Step 9: If l< Ned, go to step1. Otherwise end.

3. VARIANTS OF BFO WITH APPLICATIONS

Liu et al. in [11], applied BFO to RFID (radio frequency identification) network scheduling. A variant of BFO, self-adaptive bacterial foraging optimization (SABFO) has been developed in which swim length of individual bacterium adjusts dynamically during search to balance the exploration/ exploitation trade off. During search if bacterium discovers a promising domain (better fitness), swim length of bacteria is adapted to smaller one (exploitation state). If a bacterium’s fitness is unchanged for a user defined number of number of steps (food exhausted), the swim length adjusts to larger one and this bacterium enters in exploration state. The simulation results when compared to GA, PSO and BFO, show that SABFO obtains superior solutions than the other methods.

In paper [13], Zhang et al. presented application of bacteria foraging optimized neural network (BFO NN) for short term electric load forecast. Global search feature of BFO lead to faster convergence of neural network. For training of neural network, BFO has been applied in feedback path. Mean square error (MSE) of the neural network has been taken as cost function to BFO. BFO is used to find optimized weights of neural network while minimizing the MSE. The authors applied this model for load forecast of NewYork City and obtained very accurate results. Simulation results also showed BFONN converges more quickly than Genetic algorithm optimized neural network (GANN).

Mangraj et al. in [2] describe application of BFO in antenna problem. It presents use of BFO for optimization of included angle of V-dipole antenna for higher directivity. MOM code has been coupled with BFO to get best included angle. Directivity of V-dipole which is function of included angle has been taken as cost function to be maximized. Since BFO is usually used to minimize cost function, a fitness function reciprocal of directivity has been taken as cost function to be minimized. Comparative results between straight line dipole and V-dipole with optimized included angle using BFO have been provided.

T Datta et al in [10], present an improved adaptive approach involving BFO (ABFO) to optimize both amplitude and phase of weights of a linear array of antennas for maximum factor at any desired direction and null in a specific direction. Principle of adaptive delta modulation has been used to make the swim step size of BFO adaptive. The method is applied to six elements linear array. It is found that ABFO is capable of synthesizing patterns with multiple nulls at any desired directions and that too with faster convergence.

In [3] BFO is used to calculate resonant frequency and feed point of a microstrip antennas with thick substrates ($h/\lambda > 0.0815$). Changes in widths of patch, for given values of substrate thickness, length, width and feed point of patch, are calculated using standard expressions and are then randomly placed in search space. Then optimization is done to meet experimental value of resonant frequency of the patch. The optimized width is then used to calculate resonant frequency of the patch. To obtain feed point location using BFO, equation input impedance patch antenna has been taken as cost function of optimization.

In [1], the paper introduces a hybrid approach involving PSO and BFO for optimization of multi-modal and high dimensional functions. The algorithm combines PSO based mutation operator with bacterial chemotaxis in order to make judicious use of exploration and exploitation abilities of search space and to avoid false and premature convergence. The algorithm is tested on five standard functions Rosenbrock, Rastrigin, Griewank, Ackley, Shekel's Foxholes and also on spread spectrum radar poly phase code design. It is found that the overall performance of the hybrid algorithm is better than stand alone BFO and at least comparable to PSO and its variants.

In paper [12], the Korani has applied BFO oriented by particle swarm optimization (PSO), called BFPSO, for tuning of PID (proportional derivative integral) controller kp, kI and kd. Conventional BFO depends on random search directions which may lead to delay in reaching global solution while PSO is prone to be trapped in local minima. In order to get better optimization, the new algorithm combines advantages of both the algorithms i.e. PSO's ability to exchange social information and BFO's ability in finding new solutions by elimination and dispersal. Simulation results demonstrated that overshoots of PID controller are reduced considerably with faster convergence and thus the hybrid algorithm outperforms conventional BFO and PSO.

The paper [5] presents an hybrid approach involving BFO and PSO. The authors utilized faster convergence property of PSO for finding corresponding positions of bacteria in the predefined domain i.e. search space. By doing so they made the search space narrowed and thus reducing computational time. The change in resonant frequencies of different patches are calculated first and placed randomly in search space of PSO and then searching starts using PSO to nearest experimental values. Then theses values are placed in search space of BFO. Root mean square value (RMSE) is taken as fitness function in BFO. The results show promising improvement in accuracy and drastic reduction in time.

Mahmoud in [7] used bacterial foraging oriented by particle swarm optimization (BF PSO or simply BSO) hybridized with Nelder-Mead algorithm to design bow-tie antenna for RFID readers. The BSO-NM algorithm is integrated with Method of Moments (MoM) to optimize the antenna. The algorithm optimizes bow tie antenna parameters i.e. height, neck width and flare angle to make it resonant at desired frequency. Reflection coefficient has been taken as cost function to be minimized. It is found that BFO-NM algorithm produced results better than those generated by individual BFO or BSO.

Chen et al. in [6] have introduced adaptive bacterial foraging optimization (ABFO) by changing swim step length dynamically during execution of the algorithm. Two models ABFO0 and ABFO1 have been reported. In ABFO0, the evolution process is divided into two phases. The algorithm starts with exploration phase, in which a large swim length is assigned to all the bacteria. The larger swim length permits them to explore the whole space fast to locate global optimum and avoid being trapped in local optima. These positions of global optimum are input to the exploitation phase. In exploitation phase, the bacteria join the resources found in previous phase and are assigned smaller swim step length so that they can better exploitation of the resources. In ABFO1 each bacterium individually performs focused and deeper exploitation of promising regions and wider exploration of other regions of search space. When a bacterium enters in a nutrients richer region with higher fitness swim length decreases to have better exploitation of the resources. When it enters in a food exhausted region, swim length increases to have faster exploration. Simulation results show that both ABFOs are definitely better than the original BFO.

BFO has been used as an image enhancement tool in [8]. The paper presents use of BFO with adaptive median filter to improve peak signal to noise ratio of a highly corrupted image in absence of original image. In first stage the image corrupted with salt and pepper noise of varied density is applied to adaptive median filter. Then, in second stage both noisy image and adaptive median filter output image are passed through BFO to minimize the error due to differences in filtered image and noisy image. Number of bacteria has been taken same as the number of pixels in image. The mean square error between noisy image and filtered image has been taken as cost function. Original image is reconstructed from image corrupted with noise as high as 90%. This quality and accuracy has been achieved with simultaneous reduction in computational time.

In [4], Sastri et al have introduced a velocity modulated bacterial foraging optimization technique (VMBFO). The

VMBFO has been obtained from hybridization of BFO and PSO to reduce convergence time. Larger population in BFO requires more convergence time. But if less population is taken, they may be insufficient to explore entire search space and there is always a possibility of being trapped in local minima. Taking this issue, the authors took minimum number of bacteria and defined contour prior to start of optimization process and thus avoiding random search. In the algorithm, initially, the entire bacteria are used in PSO and are treated as particles. After completion of PSO, all the particles are redesignated as bacteria and are allowed to search randomly the exact locations by means of BFO. The algorithm has been used to calculate the resonant frequency of rectangular microstrip patch antenna.

4. CONCLUSION

This paper provides a comprehensive review of the Bacterial Foraging Optimization Algorithm and its variants as it exists till date. Global search approach of BFO results in arger convergence time. This issue has addressed in details and solutions provided by various researchers have been discussed. New efficient variants of BFO like PSO-BFO and VM-BFO have been discussed in detail. Wide variety of engineering applications of BFO like electric load forecasting, RF-ID network scheduling, antenna design etc. have been presented and their optimization features have been discussed. The paper will go long way for the new researchers entering in the field of evolutionary algorithms.

5. REFERENCES

[1] Biswas A, Dasgupta S, Das S and Abraham A, " Synergy of PSO and BFO-A Comparative Study on Numerical Benchmarks", International Symposium on Hybrid Artificial Intelligent Systems (HAIS), November, 2007, Salamanca, Spain, pp. 255-263.

[2] Mangraj B B, Misra IS and Barisal AK, "Optimizing Included Angle of symmetrical V-Dipoles for Higher Directivity Using Bacteria Foraging Optimization Algorithm", PIER B, Vol. 3, 2008, pp. , 295-314

[3] Sastri GSVR, Pattnaik SS, Bajpai OP, Devi S, Sagar CV, Patra PK and Bakwad KM, "Bacterial Foraging Optimization Technique to Calculate Resonant Frequency of Rectangular Microstrip Antenna", Int. J. RF Microwave Computer Aided Eng., Vol. 18, 2008, pp. , 383-388.

[4] Sastri GSVR, Pattnaik SS, Bajpai OP, Devi S and Bakwad KM , "Velocity Modulated Bacterial Foraging Optimization Technique (VMBFO)", Applied Soft Computing Journal, Vol. 11 (1), January 2011, pp. 154-165.

[5] Sastri GSVR, Pattnaik SS, Bajpai OP, Devi S, Bakwad KM and Patra PK "Intelligent Bacterial Foraging Optimization Technique to Calculate Resonant Frequency of RMA", IJMOT, Vol 4(2), March 2009, pp.67-75.

[6] Chen H, Zu Y, and Hu K, "Adaptive Bacterial Foraging Optimization", Abstracts and Applied Analysis, Vol. 2011, 2011, pp. 1-27.

[7] Mahmoud KR, "Design Optimization of A Bow-Tie Antenna For 2.45 GHz RFID Readers Using A Hybrid BSO-NM Algorithm", PIER 100, 2010, pp. 105-117

[8] Bakwad KM, Pattnaik SS, Sohi BS, Devi S, Panigrahi BK, Sastri GSVR, "Bacterial Foraging Optimization Technique Cascaded with Adaptive Filter to Enhance Peak Signal to Noise Ratio from Single Image", IETE Journal of Research, Vol 55 (4), Jul-Aug 2009.

[9] Passino KM, "Biomimicry of Bacterial Foraging", IEEE Control System Magzine, Vol. 22, 2002, pp. 52-67.

[10] Datta T, Misra IS, Mangraj BB, Imtiaj S, "Improved Adaptive Bacteria Foraging algorithm in Optimization of Antenna Array for Faster Convergence", PIER C, Vol. 1, 2008, pp. 143-157.

[11] Liu W, Chen H, Chen H, and Chen M, "RFID Network Scheduling Using an Adaptive Bacteria Foraging Algorithm", Journal of Computer Information Systems (JCIS), Vol. 7(4), 2011, pp. 1238-1245.

[12] [12] Korani W, " Bacterial Foraging Oriented By Particle Swarm Optimization Strategy for PID Tuning", GECOO 2008, 12-16 July, 2008, Atlanta, USA, pp. 1823-1826.

[13] Zhang Y, Wu L and Wang S, " Bacterial Foraging Optimization Based Neural Network for Short term Load Forecasting", JCIS, Vol. 6(7), 2010, pp. 2099-2105