

Cairo University

Egyptian Informatics Journal

[www.elsevier.com/locate/eij](http://www.elsevier.com/locate/eij) [www.sciencedirect.com](http://www.sciencedirect.com/science/journal/11108665)



Detection of Bundle Branch Block using Adaptive Bacterial Foraging Optimization and Neural Network



# Padmavthi Kora [a](#_bookmark0),[\*](#_bookmark2), Sri Rama Krishna Kalva [b](#_bookmark1)

a *Dept of ECE, Gokraju Rangaraju Institute of Engineering and Technology, Hyderabad, India*

b *Dept. of ECE, V R Siddhartha Engineering College, Vijayawada, India*

Received 7 April 2015; revised 23 March 2016; accepted 24 April 2016

Available online 6 June 2016

Abstract The medical practitioners analyze the electrical activity of the human heart so as to pre- dict various ailments by studying the data collected from the Electrocardiogram (ECG). A Bundle Branch Block (BBB) is a type of heart disease which occurs when there is an obstruction along the pathway of an electrical impulse. This abnormality makes the heart beat irregular as there is an obstruction in the branches of heart, this results in pulses to travel slower than the usual. Our cur- rent study involved is to diagnose this heart problem using Adaptive Bacterial Foraging Optimiza- tion (ABFO) Algorithm. The Data collected from MIT/BIH arrhythmia BBB database applied to an ABFO Algorithm for obtaining best(important) feature from each ECG beat. These features later fed to Levenberg Marquardt Neural Network (LMNN) based classifier. The results show the proposed classification using ABFO is better than some recent algorithms reported in the liter- ature.

© 2016 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information,

KEYWORDS

ECG;

Bundle Branch Block; ABFO;

LMNN;

MIT–BIH Arrhythmia database

Cairo University. This is an open access article under the CC BY-NC-ND license ([http://creativecommons.](http://creativecommons.org/licenses/by-nc-nd/4.0/)

[org/licenses/by-nc-nd/4.0/](http://creativecommons.org/licenses/by-nc-nd/4.0/)).

1. Introduction

Globally heart diseases are the most prevalent cause for human mortality. Every year, 9.4 million deaths are attributed

\* Corresponding author.

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

**Production and hosting by Elsevier**

to high Blood Pressure (BP) including 51% deaths due to strokes and 45% deaths due to the coronary heart diseases. Most cardiac diseases are due to risk factors, such as unbalanced diet, high blood pressure, tobacco usage, obesity, diabetes and physical inactivity.

BBB developed when there was a block along the conduc- tion path of electrical pulses in the heart. BBB makes it diffi- cult for the heart to pump blood effectively through the heart circulatory system because the impulse deviates from the preferred path. This delay may be observed through the changes in the ECG. There are two types of BBB : Left Bundle Branch Block (LBBB) and Right Bundle Branch Block

<http://dx.doi.org/10.1016/j.eij.2016.04.004>

1110-8665 © 2016 Production and hosting by Elsevier B.V. on behalf of Faculty of Computers and Information, Cairo University. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

(RBBB). ECG changes in Left Bundle Branch Block (LBBB) are

* Increased QRS complex duration (>0.12 s).
* Increased Q wave amplitude.
* Abnormal T wave.
* ECG changes in Right Bundle Branch Block (RBBB):
* Increased QRS complex duration (>0.12 s).
* RSR’ format
* T wave inversion

as depicted in [Figs. 1–3](#_bookmark3). ECG is the cost effective tool for ana-

lyzing the cardiac abnormalities. The diagnosis of the heart diseases by the physicians done by following a standard rule set (changes). In this project, our aim was to automate the above procedure so that it leads to correct diagnosis of the ail- ment of BBB. Good performance depends on the efficient and accurate detection of ECG features. Here in this paper ABFO technique used as the feature extraction (optimization) technique.

In recent years, many models are developed based on the evolutionary behaviors of living beings and have been applied for solving the practical real world issues. Among them, Bac- terial Foraging Optimization (BFO) [[27–29]](#_bookmark28) may be a popula- tion based search optimization technique. Bacterial forage activity of *Escherichia coli (E. coli)* bacteria is used extensively as a model to solve many engineering applications. In Recent years, BFO has been applied with success to some engineering concepts such as, harmonic estimation [[5]](#_bookmark19), optimum manage- ment [[7]](#_bookmark19), reduction machine learning and transmission loss [[6,25,26,34]](#_bookmark19) and so on.

A scientific analysis of the simulated chemotaxis by the classical gradient descent search algorithm is explained in [[9,10]](#_bookmark19). The analysis shows that varying the chemotaxis step- size can lead to better convergence as compared to a fixed step-size. The adaptation schemes, proposed for automatic adjustment of the step-size, are simple and do not impose any additional burden on the BFO algorithm regarding an excess number of functions. Several researchers have investi- gated the adaptation of step size in both deterministic and stochastic gradient descent optimization algorithms [[11–16]](#_bookmark19)

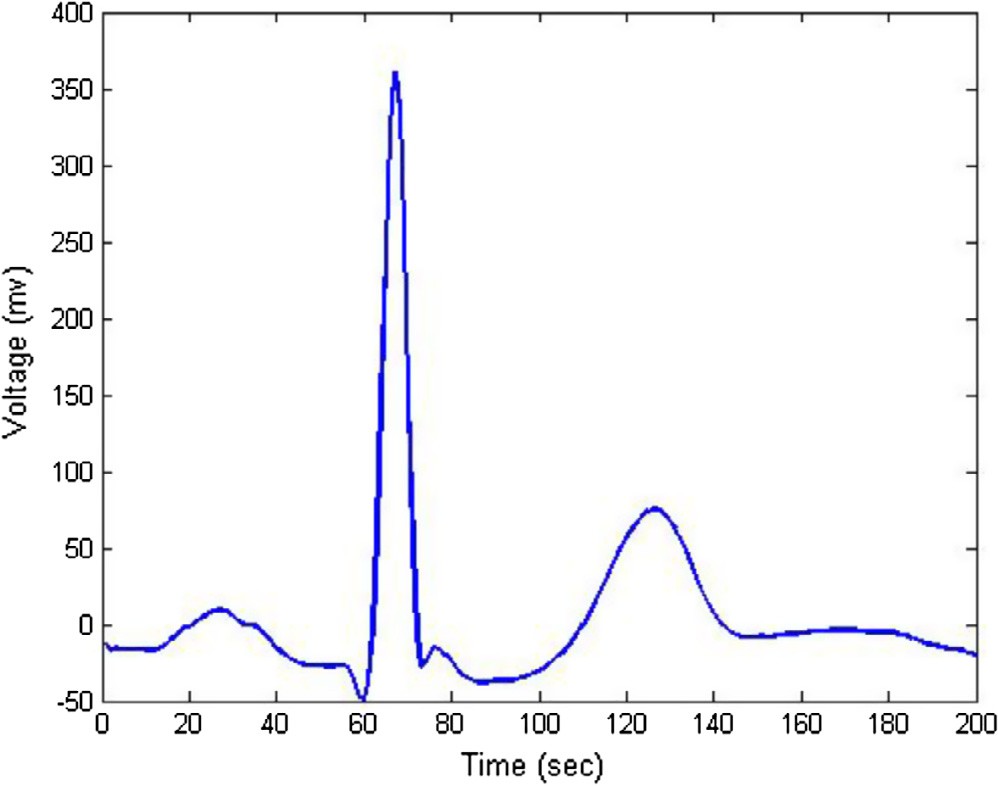


Figure 1 Normal beat.

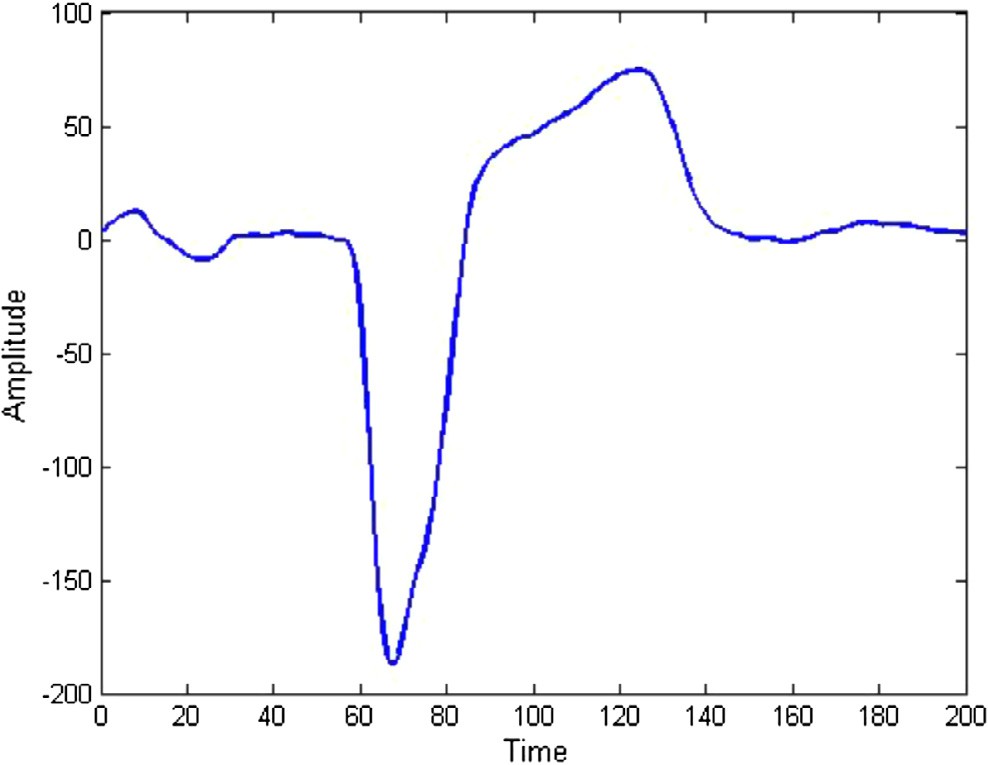


Figure 2 Left Bundle Branch Block.

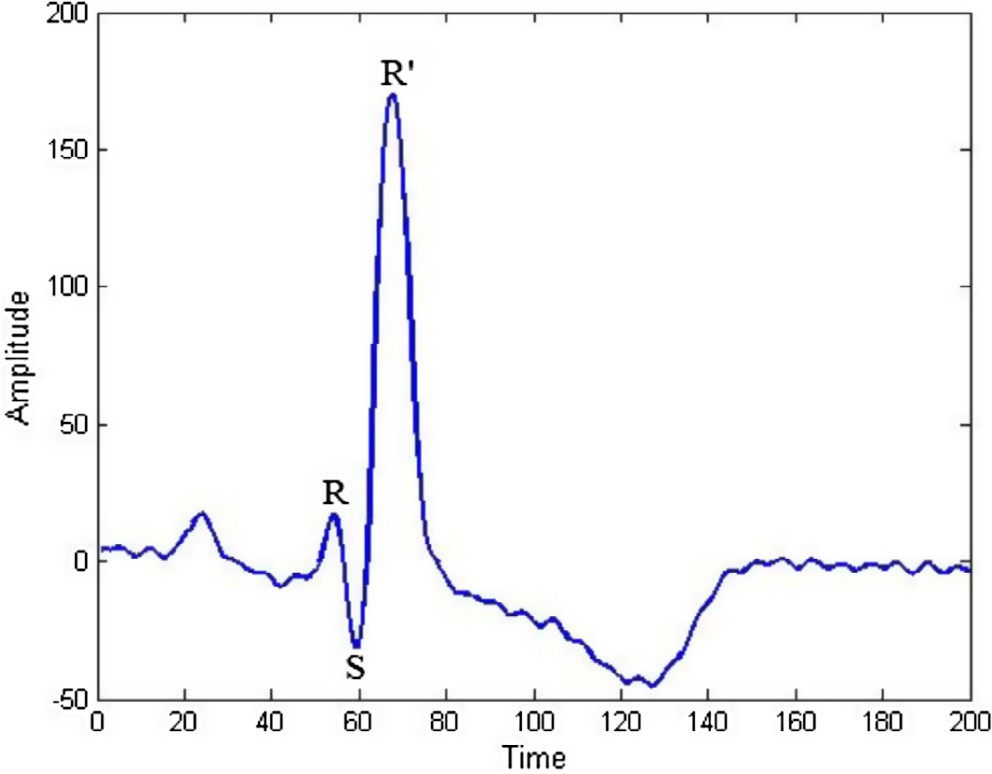


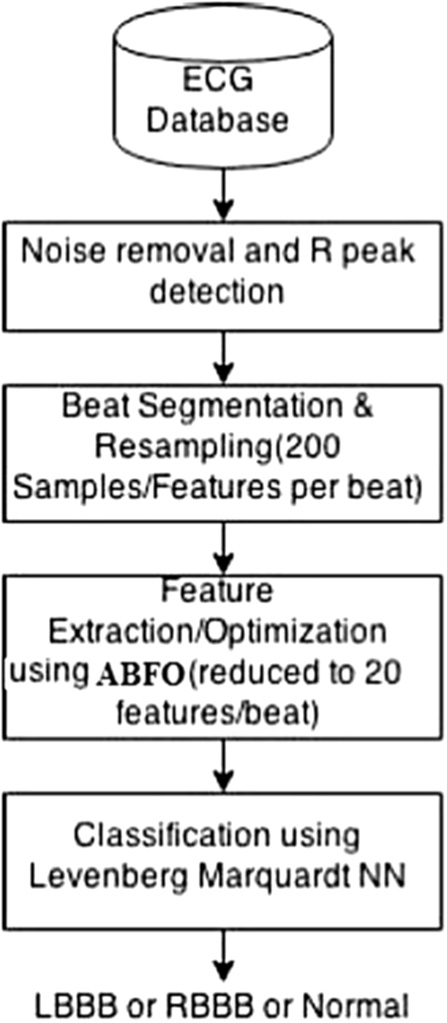
Figure 3 Right Bundle Branch Block.

in the context of training neural networks. Unlike the works cited in [[11–16]](#_bookmark19) adaptation schemes implemented in this paper for the chemotaxis step-size are not based on complex calculus techniques, such as Hessian matrix evaluation [[17,18]](#_bookmark21). They are solely based on the fitness information of individual avoiding any oscillatory behavior around the optimum and accelerate the convergence of the bacterium toward an optimum.

The proposed ABFO compared with the Genetic Algorithm (GA) [[20,21,4,19]](#_bookmark22) which is a traditional algorithm for optimization of ECG features on the following performance measures such as convergence speed, and the accuracy in the final output. Bacterial foraging feature classification using neural network fuzzy learning implemented in [[8]](#_bookmark19).

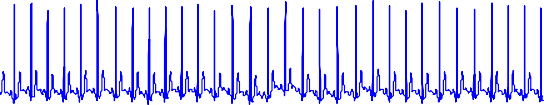
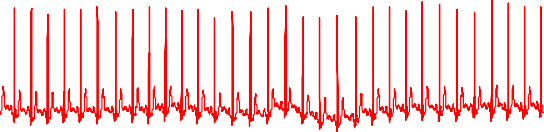
The layout of the paper organized as follows. In Section [2](#_bookmark4), we outline the Preprocessing ECG such as Data collection, Noise removal and Segmentation of ECG into beats. In Section [3](#_bookmark8), we explained the Algorithm of classical BFO and modification to the BFO algorithm. Section [4](#_bookmark12) provides a classification of ABFO features. Section [5](#_bookmark16) contains results and Sections6 and 7 provide discussion and conclusions. The classification flow diagram shown in [Fig. 4](#_bookmark5).

|  |  |  |  |
| --- | --- | --- | --- |
| Table 1 | MIT–BIH record numbers. |  | |
| Record | NB | LBBB | RBBB |
| 100 | 2237 | 0 | 0 |
| 01 | 1858 | 0 | 0 |
| 103 | 2080 | 0 | 0 |
| 106 | 1505 | 0 | 0 |
| 109 | 0 | 2490 | 0 |
| 111 | 0 | 2121 | 0 |
| 118 | 0 | 0 | 2164 |
| 123 | 1513 | 0 | 0 |
| 124 | 0 | 0 | 1529 |
| 207 | 0 | 1457 | 85 |
| Total | 3778 | 6068 | 9193 |
|  |  |  |  |

Figure 4 ECG classification flow diagram.

1600

1400



1200

1000

800

Amplitude

600

1. Preprocessing of ECG data
   1. *Data acquisition*

The data for the classification were collected from the MIT– BIH Arrhythmia Database [[30]](#_bookmark29), which consist of 5 normal, 3 LBBB and 3 RBBB patients data at 360 Hz sampling rate of one-hour duration. The total number of beats in each class is as shown in [Table 1](#_bookmark6).

* 1. *Noise removal and beat segmentation*

To remove the baseline wander present in the signal Sgolay FIR smoothing filter was used as shown in [Fig. 5](#_bookmark7). RR interval is the distance between two R peaks. 1/3 of the RR interval samples to the left of R peak and 2/3 of the RR interval sam- ples to the right of R peak were considered as one beat. Differ- ent patients have different RR intervals. Each ECG beat was re-sampled to 200 samples so that it is easy to process them.

1/3 of RR interval: (R peak): 2/3 of RR interval.

1. Bacterial foraging optimization

The bacterial foraging activity of *E. coli* bacteria [[24]](#_bookmark23) is used as the inspiration for extracting (optimizing) the features of ECG. Feature selection may be an international optimization prob- lem in machine learning. It reduces the number of features, which are redundant and noisy leading to acceptable accuracy. The Bacterial Foraging Optimization (BFO) projected by Passino [[2,3]](#_bookmark24) in 2002 relies on the selection that tends to get rid of an organism with low search methods. Several genera- tions with poor foraging methods have been eliminated, whereas only the organisms with good search strategy are sur- viving since they are the fittest. The BFO formulates the search

400

200

0

-200

0 1000 2000 3000 4000 5000 6000 7000 8000 9000 10000

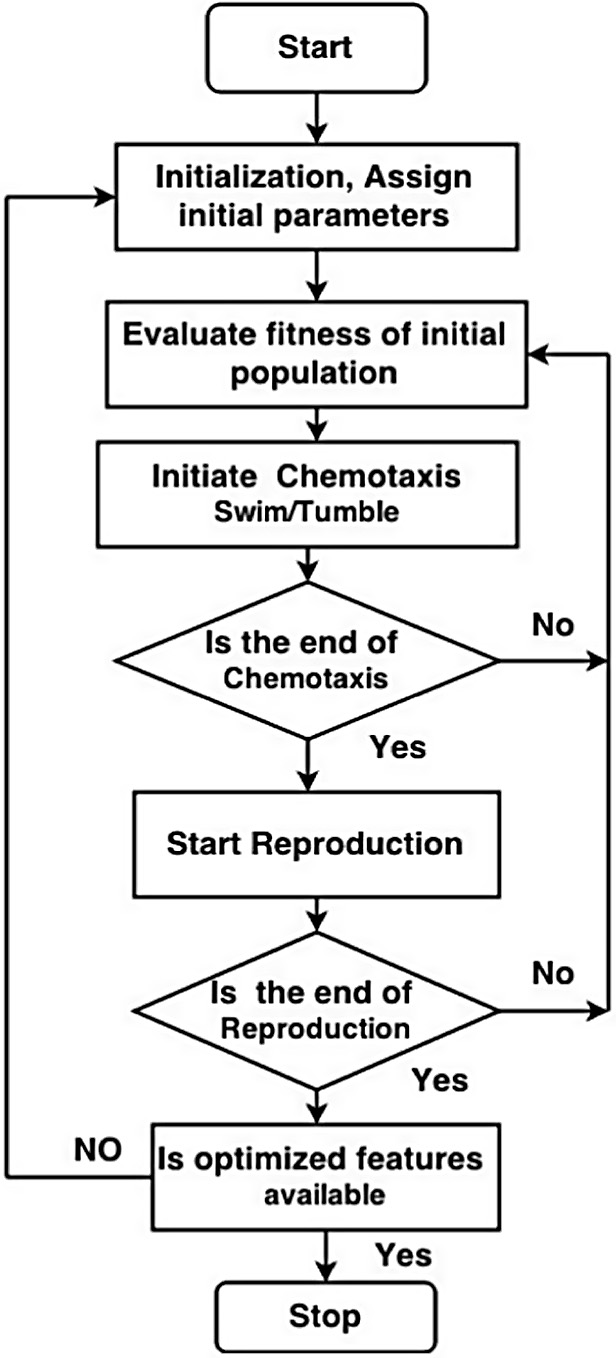
Time

Figure 5 ECG baseline wander removal, up signal: original signal, down signal: baseline wander re-moved signal.

behavior as exhibited by *E. coli* to solve the optimization prob- lem. Certain real-world optimization problems [[33]](#_bookmark29), BFO have been reportedly outperformed several powerful optimization algorithms regarding final accuracy.

Bacteria move into a random direction to search for favor- able direction of increasing nutrients. Hence this optimization technique is useful when the gradient of the cost function is not known. BFO is good because of its less mathematical complex- ity. The BFO is a non-gradient optimization problem inspired by the search mechanism used by *E. coli* microorganism, as it maximizes its energy intake per unit time spent in search. The three operating steps in bacteria per area unit are

1. Chemotaxis
2. Swarming
3. Reproduction
4. *Chemotaxis:* The movement of *E. coli* bacterium can be explained via two steps, Swimming and Tumbling, through the flagella. Basically, the *E. coli* bacteria will move in 2 alternative ways. It will swim for an amount of time within the same direction, or it will tumble (change direction). It will alternate between these 2

modes of operation for its entire life period. Suppose *x*(*i*) represents *i*th bacteria and *C* is the size of the step taken in the random direction specified by the run length, in the process of chemotaxis *x*(*i* + 1) of the bacteria could also be given by

*x*(*i* + 1)= *x*(*i*)+ *C*(*i*) *Del*(*i*)

*Del*(*i*)*DelT*(*i*)

(1)

whose elements lie in —1 to 1. The simulated chemotac- where ‘*Del*’ indicates a vector in the random direction tic movement of *E. coli* bacterium may be viewed as a

random hill climbing.

1. *Swarming:* In *E. coli* bacteria group behavior is observed as in several species, where complex and stable spa- tiotemporal groups are formed in a semisolid nutrient medium. The *E. coli* bacteria form themselves like a traveling ring and moving down toward the nutrient food. The cells in the *E. coli*, excited by a high level of succinate, release an attractant aspartate, which helps them to arrange into groups and thus move as coaxial patterns of swarms with high density. The cell to cell fit- ness of each bacterium calculated by the following objec- tive function.

*d*

*f*(*x*)=

Xh

*i*=1

*i*+1 *i*

100 *x* — *x*2 2

+ (*xi* — 1)2i

(2)

where *d* is the dimension and *xi* is the *i*th bacterium.

1. *Reproduction:* The unhealthy bacteria finally die while the remaining healthy bacteria (those yielding a higher value of the cost function) asexually split into 2 bacteria, then placed in the same location, and kept the swarm size constant.
   1. *Algorithm for BFO*

The complete pseudo-code for feature optimization using BFO given below, and the flow chart for the BFO algorithm is shown in [Fig. 6](#_bookmark9).

Step 1 Set the BFO parameters.

*N*: Number of bacteria in the population.

*Nc*: Count of Chemotaxis steps. *Nre*: Total reproductive steps. *n*: Dimension of the problem.

*Ned*: Total number of elimination dispersal events.

*Ped*: Probability of elimination dispersal.

*C*(*i*): Step-size taken by tumble.

Step 2 Begin Elimination dispersal loop.

Step 3 For every reproduction step perform the following. Step 4 For every chemotaxis step perform the following.

* + 1. Calculate the fitness function (*J*) of the initial population using the Eq. [(2)](#_bookmark10).
    2. Set *J* last = *J* Hold this value.
    3. Tumble: create a random vector delta from —1 to 1.
    4. Move: Let the bacterium move to a position with step size *C*(*i*) using the Eq. [(1)](#_bookmark11) called Tumble.
    5. Again Swim
       1. *m* = 0.

Figure 6 BFO flow chart.

* + - 1. While *m* < *Ns* (if have not climbed too long).
      2. *m* = *m* + 1.
      3. If *J*(*i*)> *J* last (if doing well).
      4. Again move the bacteria using Eq. [(1)](#_bookmark11). Use this *J* to calculate the new *J* as in point (iv). Let *J* last = *J*(*i*).
      5. Else, let *m* = *Ns*; the end of the while statement.
      6. Go to the next bacteria i.e. go to (i) to calcu- late the next bacteria.

Step 5 End of Chemotaxis loop? If NO, repeat Step 4. Step 6 Begin Reproduction loop.

* + - * 1. Calculate the health of each bacterium by find- ing the maximum cost value of each bacterium.
        2. The bacteria with the lowest *J* health values die and the remaining bacteria with the best values are split into two bacteria thus making the pop- ulation of bacteria constant.
  1. *Adaptive BFO: Our contribution*

BFO with fixed step size *C*(*i*) suffers from two main problems [[1]](#_bookmark25):

* If step-size is very large, then the precision gets down although the bacterium reaches the vicinity of optimum

point rapidly. It moves around the maximum for the remaining chemo-taxis steps.

have taken the transpose of the above data so that the rows of the matrix represent features.

0 Bacterium 1 – – – Bacterium *N* 1

* If the step-size is small, then it takes many chemotaxis steps to reach out to the optimal point.

So the converging rate decreases. It may not reach optimum point using a small number of iterations. Hence for increasing convergence speed and decreasing the error in the final out the step-size plays a major role. So it is required to adjust the step-

Data*T* = B

BB@

B

*f*1 *f*1 *f*1 *f*1 *f*1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *f*2 | *f*2 | *f*2 | *f*2 | *f*2 |
| *f*3 | *f*3 | *f*3 | *f*3 | *f*3 |
| *f*4 | *f*4 | *f*4 | *f*4 | *f*4 |
| *f*5 | *f*5 | *f*5 | *f*5 | *f*5 |
| *f*6 | *f*6 | *f*6 | *f*6 | *f*6 |

C

C CCA

C

size depending on the distance between the bacteria from the optimal point. If the variation is very high, then the step-size is to be increased and if the deviation is small indicating that the bacterium is near to the optimal point, then the step-size must be reduced. Here, the principle of adaptive delta modula-

Here our aim is to reduce features (rows) to keep the size of population same as above. After applying the ABFO algo- rithm, the features are reduced as shown in the matrix below:

0 Bacterium 1 – – – Bacterium *N* 1

tion is used to control the step-size. In adaptive delta modula- tion, the error between the actual signal and the predicted value of the signal is integrated and then the output of the inte- grator is given as the input to the voltage controlled oscillator

Data*T* =

@

B

*f*2 *f*2 *f*2 *f*2 *f*2

*f*1 *f*1 *f*1 *f*1 *f*1 C

A

*f*3 *f*3 *f*3 *f*3 *f*3

(VCO) that adjusts the step-size. The procedure for Adaptive step-size shown in [Fig. 7](#_bookmark13).

* *X*: The parameter to be optimized.
* *E*(*j*): Deviation from the desired value.
* *C*(*i*, *j*): Step size to be modified in each chemotaxis step depending on the deviations in the previous steps.

Here we have taken the deviation in the last chemotaxis step and is multiplied with the previous step-size. The multi- plier increases or reduces the step-size accordingly. *X* for the next step, is obtained by adding the step size to the previous value.

* 1. *Implementation for ECG feature extraction*

In the year 2002 Passino designed BFO algorithm to reduce the size of population by mimicking the behavioral model (forag- ing strategy) of *E. coli* bacteria present in our inter-stain. The matrix below shows that there are *N* bacteria each having six features. By using ABFO algorithm [[31,32,35]](#_bookmark29) the size of fea- tures can be reduced by taking the features with good fitness strategy only.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *f*1 | *f*2 | *f*3 | *f*4 | *f*5 |
| *f*1 | *f*2 | *f*3 | *f*4 | *f*5 |
| *f*1 | *f*2 | *f*3 | *f*4 | *f*5 |
| *f*1 | *f*2 | *f*3 | *f*4 | *f*5 |
| *f*1 | *f*2 | *f*3 | *f*4 | *f*5 |

The features with the lowest health values die and the

remaining features are placed in the descending order of their cost value (health status). Again find the transpose of the above matrix as

0 Bacterium 1 *f*2 *f*1 *f*3 1

* *f*2 *f*1 *f*3

Data*T* = B – *f*2 *f*1 *f*3 C

B C

* *f*2 *f*1 *f*3

B@ Bacterium *N f*2 *f*1 *f*3 CA

ECG beat features optimized to 20 features. The ABFO gives optimized features (best features) for the classification. ECG beat features before optimization = [123 ... 200]. The optimized ECG features (20 features) using ABFO algorithm are [676866696570716472 63736274616075597658 77].

1. Classification by Back Propagation Neural Network

Back Propagation Neural Network (BPNN) [[23]](#_bookmark26) is widely used in the machine learning applications. BPNN structure made up of interconnected layers: The input layer, hidden layers (one or

0 Bacterium 1

*f*6 1

more) and output layer. The input to the input layer is fed by

Data = B

B@

–

–

–

Bacterium *N*

*f*6 C

*f*6 *f*6

*f*6 C

A

the external source. The internal link between input and out- put layers is provided by the hidden layer. The output results of the Neural Network can be taken from the output layer as shown in [Fig. 8](#_bookmark14).

where *f*1, *f*2, *f*3, *f*4, and *f*5 are the features of each bacterium. In this study, the ABFO algorithm is used for reducing the features (not for reducing the bacterium population). So we

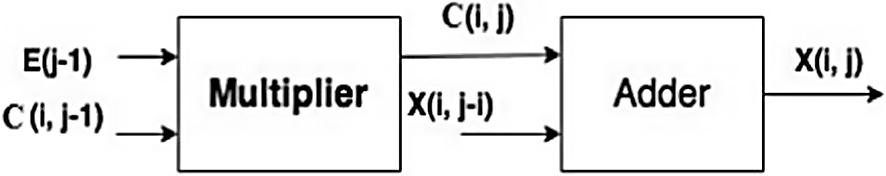
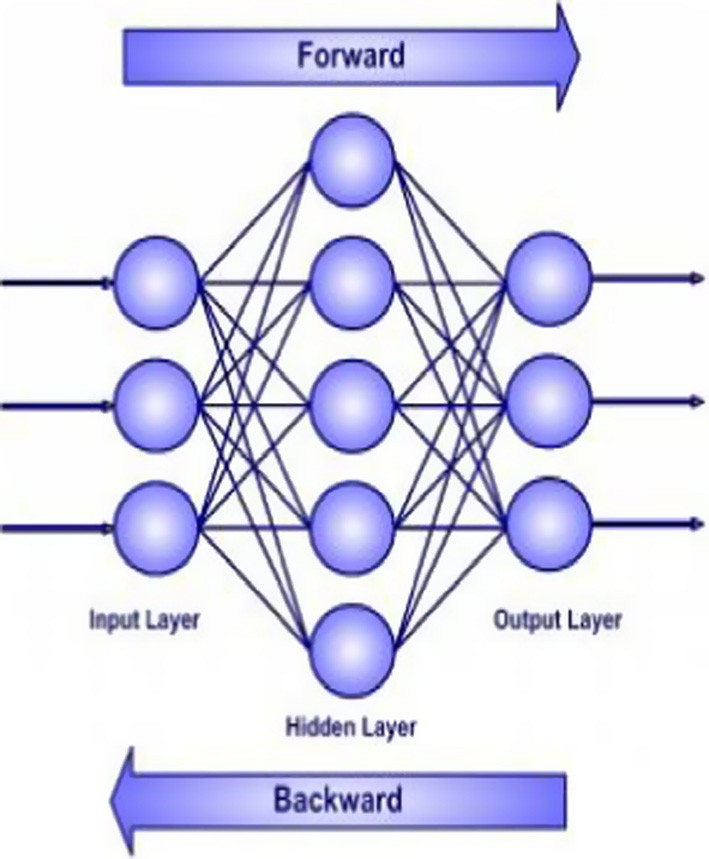


Figure 7 Adaptive step-size process.

* 1. *Levenberg–Marquardt Neural Network (LMNN)*

In this work for the detection of BBB, back propagation Levenberg–Marquardt Neural Network (LMNN) was used. This NN provides rapid execution of the network to be trained. To test the performance of this algorithm, Scalar Con- jugate Gradient (SCG) NN and LMNN were used. The LMNN algorithm is a robust, and very simple method for approximating a function [[36]](#_bookmark30). The LMNN [[22]](#_bookmark27) gives a numer- ical solution for minimizing a nonlinear function, over a space of parameters. The LMNN is an alternative to the Gauss– Newton technique for minimizing a function. SCG NN

|  |  |  |  |
| --- | --- | --- | --- |
| Table 3 Accuracy Comparison of GA, BFO and ABFO with SCG NN classifier. | | | |
| Classifier | Sen (%) | Spe (%) | Accuracy (%) |
| GA + SCG NN | 98.2 | 97.2 | 97.9 |
| BFO + SCG NN | 96.5 | 95.5 | 95.6 |
| ABFO + SCG NN | 96.7 | 96.7 | 98.1 |
|  |  |  |  |

Sensitivity = True Positive (TP)

True Positive (TP)+ False Positive (FN)

(4)

Accuracy = TP + TN

TP + TN + FP + FN

× 100 (5)

Figure 8 Feed forward back propagation NN.

method provides conjugate directions of search instead of per- forming linear search. The network training and testing are performed using 12,692 and 6347 ECG beats correspondingly by setting the total number of iterations to 1000 and mean square error less than 0.001 with the minimum time requirement.

1. Results

The performance of ABFO is compared with Genetic Algo- rithm (GA), and BFO Optimization techniques and the results are shown in [Table 2](#_bookmark17). The ABFO and GA features are classi- fied using SCG NN, LM NN as in [Table 3](#_bookmark15):

* Count of Normal beats used for classification – 9193.
* Count of RBBB beats used for classification – 3778.
* Count of LBBB beats used for classification – 6068.
* Total number of beats used for classification – 19,039.
* Count of correctly classified beats – 18,800.
* Total misclassified beats – 239.

For measuring accuracy two parameters sensitivity (Sen)

and specificity (Spe) are calculated using the following equations:

* TP (True Positive) = Correctly classified Normal beats.
* TN (True Negative) = Correctly classified Abnormal beats.
* FP (False Positive) = Count of Normal beats classified as Abnormal.
* FN (False Negative) = Count of Abnormal beats classified as Normal.

In the training, we applied multilayer NN, and checked the network performance and decided whether any changes to be made to the training process, or the data sets, the network architecture. First, check the training record, ‘trainlm’ Matlab function.

The property training indicates the iteration is up to the point, where the performance of the validation reached a min- imum. The training continued for 16 iterations before the stop. The next step is validating the network, a plot of epochs versus Mean Squared Error (MSE), which shows the relationship between the number of epochs of the network to the MSE as shown in [Fig. 9](#_bookmark18). If the training is perfect the network outputs and the targets are exactly equal, but that is rare in practice.

1. Discussion

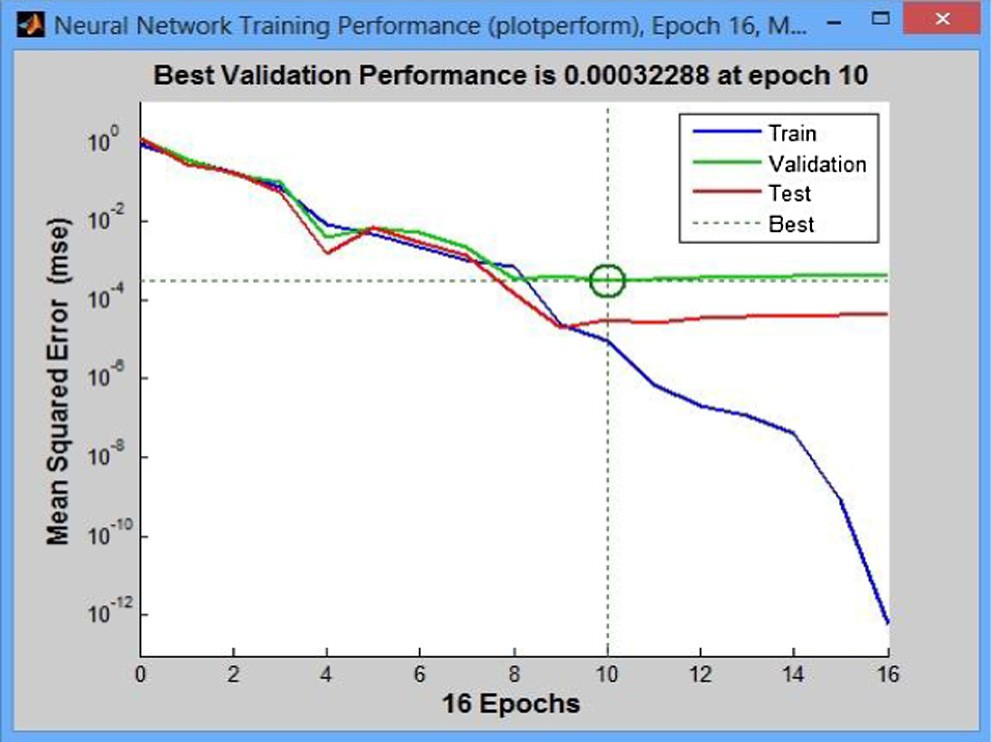
The proposed ABFO is compared against other four BBB detection algorithms such as Wavelet Transform (WT), Con- tinuous Wavelet Transform (CWT), Wavelet Transform and

Specificity = True Negative (TN)

True Negative (TN)+ False Positive (FP)

|  |  |  |  |
| --- | --- | --- | --- |
| Table 2 Accuracy comparison of GA, BFO and ABFO with LM NN classifier. | | | |
| Classifier | Sen (%) | Spe (%) | Accuracy (%) |
| GA + LM NN | 98.5 | 98.2 | 97.1 |
| BFO + LM NN | 96.5 | 96.2 | 96.1 |
| ABFO + LM NN | 98.5 | 98.9 | 98.74 |

(3)

Figure 9 Neural network training performance plot.

|  |  |  |
| --- | --- | --- |
| Table 4 Comparative study for the detection of BBB. | | |
| Studies | Approach | Accuracy (%) |
| Yu et al. [[37]](#_bookmark30) | WT and PNN | 98.39 |
| Kutlu et al. [[38]](#_bookmark30) | CWT | 97.3 |
| Ceylan et al. [[39]](#_bookmark30) | WT | 98.1 |
| Kora et al. [[36]](#_bookmark30) | BFPSO | 98.1 |
| Proposed approach | ABFO | 98.74 |
|  |  |  |

Probabilistic Neural Network (PNN) and Hybrid Bacterial Foraging and Particle Swarm Optimization (BFPSO) in terms of related features selected from the original database and clas- sification accuracy obtained from different classifiers using Matlab software.

The work in [[39]](#_bookmark30) explores an experimental study of using WT for extracting relevant features and KNN based classifier for the detection of BBB. The work presented in [[38]](#_bookmark30) used mor- phological features for classification using SVM. The work proposed in [[37]](#_bookmark30) used Arrhythmia dataset from MIT/BIH repository and 20 morphological and wavelet features are extracted then PNN is used for supervised learning and classi- fication. The work in [[36]](#_bookmark30) explores an experimental study of using Hybrid BFPSO for extracting relevant features and LMNN based classifier was used for the detection of BBB. From the experiments, it is concluded that the proposed ABFO with LMNN classifier outperformed other three algo- rithms with a selection of a minimal number of relevant fea- tures. This increases the classification accuracy as shown in [Table 4](#_bookmark20). The ABFO employed to intelligently select the most relevant features that could increase the classification accuracy while ignoring noisy and redundant features.

1. Conclusion

In the present study we developed a simple computational model for the detection of BBB using the ABFO algorithm. It also projected that ABFO algorithm can be used for ECG feature extraction in (or ‘‘intending to”) improving its conver- gence behavior and decreasing the error in the final output. The classical BFO algorithm was compared with the ABFO algorithm and the evolutionary based algorithm such as GA. In our study, we observed improved classification accuracy. The ABFO variants were shown to provide better results than their classical BFO for all the tested data. Thus, this ABFO optimization method that we applied may be useful for further such investigations.

References

1. [Nasir Ahmad NK, Tokhi MO, Ghani N Maniha Abd. Novel](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0005) [adaptive bacteria foraging algorithms for global optimization.](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0005) [Appl Comput Intell Soft Comput 2014, 7 pages](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0005).
2. [Passino KM. Biomimicry of bacterial foraging for distributed](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0010) [optimization and control. IEEE Control Syst Mag 2002;](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0010) [22(3):52–67](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0010).
3. [Liu Y, Passino KM. Biomimicry of social foraging bacteria for](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0015) [distributed optimization: models, principles, and emergent behav-](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0015) [iors. J Opt Theory Appl 2002;115(3):603–28](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0015).
4. [Kim DH, Abraham A, Cho JH. A hybrid genetic algorithm and](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0020) [bacterial foraging approach for global optimization. Inform Sci](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0020) [2007;177(18):3918–37](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0020).
5. [Mishra S. A hybrid least square-fuzzy bacterial foraging strategy](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0025) [for harmonic estimation. IEEE Trans Evol Comput 2005;](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0025) [9(1):61–73](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0025).
6. Tripathy M, Mishra S, Lai LL, Zhang QP. Transmission loss reduction based on FACTS and bacteria foraging algorithm. In: Proc PPSN; 2006. p. 222–31.
7. [Mishra S, Bhende CN. Bacterial foraging technique-based opti-](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0035) [mized active power filter for load compensation. IEEE Trans](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0035) [Power Delivery 2007;22(1):457–65](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0035).
8. Kim DH, Cho CH. Bacterial foraging based neural network fuzzy learning. In: Proc IICAI; 2005. p. 2030–6.
9. [Avriel M. Nonlinear programming: analysis and methods. 1st ed.](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0045) [New York: Dover; 2003. p. 288–99 [ch. 10, sec. 10.1]](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0045).
10. [Snyman JA. Practical mathematical optimization: an introduction](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0050) [to basic optimization theory and classical and new gradient-based](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0050) [algorithms. New York: Springer-Verlag; 2005](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0050).
11. Darken C, Moody J. Toward faster stochastic gradient search. In: Proc advances neural inform process syst (NIPS), vol. 4. Palo Alto (CA): Morgan Kaufmann; 1992. p. 1009–16.
12. [Armijo L. Minimization of functions having Lipschitz continuous](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0060) [first partial derivatives. Pac J Math 1966;16(1):1–3](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0060).
13. Plagianakos VP, Magoulas GD, Vrahatis MN. Learning rate adaptation in stochastic gradient descent. In: Proc advances convex anal global optimization, noncovex optimization appl, vol.

54. Dordrecht (Netherlands): Kluwer; 2001. p. 433–44 [ch. 27].

1. [Hsin HC, Li CC, Sun M, Sclabassi RJ. An adaptive training](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0070) [algorithm for back propagation neural networks. IEEE Trans](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0070) [Syst Man Cyber 1995;25(3):512–4](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0070).
2. Orr GB, Leen TK. Using curvature information for fast stochastic search. In: Proc advances neural inform process syst, vol. 9. Cambridge (MA): MIT Press; 1997.
3. Giudici M, Queirolo M, Valle M. Evaluation of gradient descent learning algorithms with adaptive and local learning rate for recognizing hand-written numerals. In: Proc ESANN, Bruges, Belgium; 2002. p. 289–94.
4. Le Cun Y, Simard PY, Pearlmutter BA. Automatic learning rate maximization by online estimation of the Hessians eigenvectors. In: Proc Eur symp artificial neural netw (ESANN); 2002. p. 156–63.
5. [Marquardt D. An algorithm for least-squares estimation of](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0090) [nonlinear parameters. SIAM J Appl Math 1963;11:431–41](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0090).
6. Kennedy J, Eberhart R. Particle swarm optimization. In: Proc IEEE int conf neural netw; 1995. p. 1942–8.
7. [Holland JH. Adaptation in natural and artificial systems. Ann](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0100) [Harbor (MI): Univ. Michigan Press; 1975](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0100).
8. [Back T. Evolutionary algorithms in theory and practice: evolution](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0105) [strategies evol programming. In: Proc genetic algorithms. London](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0105) [(U.K.): Oxford Univ. Press; 1996](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0105).
9. [Ibn Ibrahimy M, Ahsan R, Khalifa OO. Design and optimization](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0110) [of Levenberg–Marquardt based neural network classifier for](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0110) [EMG signals to identify hand motions. Meas Sci Rev 2013;](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0110) [13(3):142–51](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0110).
10. [Sapna S, Tamilarasi A, Kumar MP. Backpropagation learning](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0115) [algorithm based on Levenberg Marquardt algorithm. CS IT –](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0115) [CSCP 2012;2012:393–8](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0115).
11. Tang WJ, Wu QH, Saunders JR. A novel model for bacteria foraging in varying environments. In: Proc ICCSA, LNCS, vol. 3980; 2006. p. 556–65.
12. Li MS, Tang WJ, Tang WH, Wu QH, Saunders JR. Bacteria foraging algorithm with varying population for optimal power flow. In: Proc Evol Workshops, LNCS, vol. 4448; 2007. p. 32–41.
13. [Tripathy M, Mishra S. Bacteria foraging-based to optimize both](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0130) [real power loss and voltage stability limit. IEEE Trans Power Syst](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0130) [2007;22(1):240–8](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0130).
14. Biswas A, Dasgupta S, Das S, Abraham A. Synergy of PSO and bacterial foraging optimization: a comparative study on numerical benchmarks. In: Proc 2nd int symp hybrid artificial intell syst (HAIS) advances soft computing ser, vol. 44. Germany: Springer-

Verlag, Innovations in hybrid intelligent systems, ASC; 2007. p. 255–263.

1. [Ulagammai L, Vankatesh P, Kannan PS, Padhy NP. Application](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0140) [of bacteria foraging technique trained and artificial and wavelet](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0140) [neural networks in load forecasting. Neurocomputing 2007;](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0140) [70(16–18):2659–67](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0140).
2. [Munoz MA, Lopez JA, Caicedo E. Bacteria foraging optimization](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0145) [for dynamical resource allocation in a multizone temperature](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0145) [experimentation platform. Anal Des Intel Syst SC Tech ASC](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0145) [2007;41:427–35](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0145).
3. [www.physionet.org/2000](http://www.physionet.org/2000).
4. [Dasgupta Sambarta, Das Swagatam, Abraham Ajith. Adaptive](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0155) [computational chemotaxis in bacterial foraging optimization: an](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0155) [analysis. IEEE Trans Evol Comput 2009;13(4)](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0155).
5. [Datta T, Misra IS. Improved adaptive bacteria foraging algorithm](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0160) [in optimization of antenna array for faster convergence. Progr](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0160) [Electromag Res C 2008;1:143–57](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0160).
6. [Bakwad KM et al. Multimodal function optimization using](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0165) [synchronous bacterial foraging optimization technique. IETE J](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0165) [Res 2010;56(2):80–7](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0165).
7. [Kowsalya M. Optimal size and siting of multiple distributed](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0170) [generators in distribution system using bacterial foraging opti-](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0170) [mization. Swarm Evol Comput 2014;1:58–65](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0170).
8. [Xu Xin, Chen Hui-ling. Adaptive computational chemotaxis](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0175) [based on field in bacterial foraging optimization. Soft Comput](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0175) [2014;18(4):797–807](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0175).
9. [Kora Padmavathi, Kalva Sri Ramakrishna. Hybrid bacterial](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0180) [foraging and particle swarm optimization for detecting Bundle](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0180) [Branch Block. SpringerPlus 2015;4(1):1–19](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0180).
10. [Yu Wei et al. Application of multi-objective genetic algorithm to](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0185) [optimize energy efficiency and thermal comfort in building design.](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0185) [Energy Build 2015;88:135–43](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0185).
11. [Yakup Kutlu, Damla Kuntalp, Mehmet Kuntalp, et al. Arrhyth-](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0190) [mia classification using higher order statistics Signal Processing.](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0190) [Commun Appl Conf 2008](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0190).
12. [Ceylan Rahime, Ozbay Yksel, et al. Wavelet neural network for](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0195) [classification of bundle branch blocks. Proc World Congr Eng](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0195) [2011;4](http://refhub.elsevier.com/S1110-8665(16)30014-7/h0195).