

Clustering of ECG Signals Based on Fuzzy Neural Network with Initial Weights Generated by Genetic Algorithm

Elaheh Sayari¹, Mahdi Yaghoobi²

1- Young Researchers and Elite Club, Marvdasht Branch, Islamic Azad University, Marvdasht, Iran

Email: elahe.sayari.66@gmail.com (Corresponding author)

2- Department of Electrical Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran

Email: yaghoobi@mshdiau.ac.ir

Received March 2013

Revised August 2013

Accepted Sept. 2013

ABSTRACT:

Early detection of heart diseases/abnormalities can prolong life and enhance the quality of living through appropriate treatment. Whereas clustering of electrocardiogram (ECG) signals will help to identify the heart diseases as soon as possible. In this regard, neural network and fuzzy logic have been used in many application areas while each of them has the advantages and disadvantages. Thus, the present paper utilizes the proposed fuzzy neural network (FNN) with initial weights generated by genetic algorithm (GFNN) for the sake of improvement testing speed, accuracy and for reducing the chance of the FNN getting stuck on a local minimum.

Four types of ECG beats (normal beat, congestive heart failure beat, ventricular tachyarrhythmia beat and atrial fibrillation beat) obtained from the Physio Bank databases were clustered by the proposed GFNN model. Model of evaluation results indicate that the proposed model can perform more accurately and it has less testing speed than the conventional statistical methods, a single ANN and FNN. The total clustering accuracy of the GFNN model is 98.23%.

KEYWORDS: Clustering, electrocardiogram, Signals, Neural Network, Fuzzy Logic, Beat.

1. INTRODUCTION

The electrocardiogram (ECG) signal is the recording of the bioelectrical activities of the cardiac system. It provides valuable information about the functional aspects of the heart and the cardiovascular system. Early detection of heart diseases/abnormalities can prolong life and enhance the quality of living through appropriate treatment [1]. For effective diagnostics, the study of ECG pattern and heart rate's variability signal may have to be carried out over several hours. Thus, Conventional methods of monitoring and diagnosing electrocardiographic changes rely on detecting the presence of particular signal features by a human observer. Due to large number of patients in intensive care units and the need for continuous observation of such conditions, several techniques for detecting automated electrocardiographic changes have been developed in the past 10 years to attempt to solve this problem [2].

Fuzzy set theory plays an important role in dealing with uncertainty when making decision in medical applications. Therefore, fuzzy sets have attracted the growing attention and interest in modern information technology, production technique, decision making, pattern recognition, diagnostics, data analysis, etc [3].

Neuro-fuzzy systems are fuzzy systems which use the artificial neural networks (ANNs) theory in order to determine their properties (fuzzy sets and fuzzy rules) by processing data samples [4]. Neuro-fuzzy systems harness the power of the two paradigms: fuzzy logic and ANNs, by utilizing the mathematical properties of ANNs in tuning rule-based fuzzy systems that approximate the way of human process information [5]. Gas can be applied for either optimization or clustering. The advantages of GAs over conventional parameteric-optimal techniques are that they are appropriate for ill-behaved problems, for global optima in highly nonlinear spaces and for adaptive algorithms [6].

Nasiri et al (2009) present the Genetic-ESVM, a novel clustering system based on a GA, which is designed to improve the generalization performance of the so-called Emphatic Support Vector Machine (ESVM) cluster. In this paper, the feature selection found by the Genetic-SVM greatly improves the quality of clustering with respect to other algorithms [7].

Chua and Woei (2011) present a non-singleton fuzzy logic classifier (NSFLC) and assess its ability to cope with uncertainties in ECG signals clustering. The NSFLC has fuzzy clustering boundary and noise suppression capability and also achieves good

clustering results using features that are easier to extract, but contains more uncertainties [8].

Doğan and Mehmet (2013) present a paper for Clustering of ECG signals based on the kernelized fuzzy c-means algorithm. In fact, this paper uses kernel methods to improve the clustering performance of the well known fuzzy c-means algorithm by mapping a given dataset into a higher dimensional space non-linearly. This method overcomes the drawbacks of the traditional algorithms such as, sensitivity to initialization and lack of prior knowledge for optimum parameters of the kernel functions [9].

De Carvalho Junior et al (2013) present a paper for clustering of ECG signals based on fuzzy cluster algorithm. This article presents the viability analysis and the development of heart disease identification embedded system. In fact, the goal of the developed system is the analysis of heart signals. The ECG signals are applied into the system that performs an initial filtering, and then uses a Gustafson–Kessel fuzzy clustering algorithm for the signal clustering and correlation [10].

Jewajinda and Prabhas (2013) present a paper for clustering of ECG signals based on a parallel genetic algorithm. In fact, this paper uses the cellular compact genetic algorithm (c-cGA) for adaptive hardware to implement it. The c-cGA not only provides a strong search capability while maintaining genetic diversity using multiple GAs but also has a cellular-like structure and is a straight-forward algorithm suitable for hardware implementation [11].

Thus, the present study utilizes a method for clustering of ECG signals based on the proposed fuzzy neural network with initial weights generated by genetic algorithm (GFNN). The proposed method improves testing speed, accuracy and also reduces the chance of the FNN getting stuck on a local minimum and analyzes uncertainties, accurately.

2. GENETIC ALGORITHM (GA)

GAs are for general purposes, in contrast to search algorithms, which are for solving complex problems. Based on the mechanics of natural selection and natural genetics, GAs work by repeatedly modifying a population of artificial structures through the application of genetic operators. Fitness information, instead of gradient information, is the only requirement for GAs. Gas can be applied for either optimization or clustering. The advantages of GAs over conventional parameter optimization techniques are appropriate for ill-behaved problems, for global optima in highly nonlinear spaces and for adaptive algorithms [12].

The required parameters are binary coded, combined together as a string, or structure called a chromosome, while each bit of the chromosome is treated as a gene. The GA starts with a population of n randomly

generated structures, where each structure encodes a solution to the task at hand. Thereafter, the GA further processes a fixed number of generations, or continues until it satisfies some predetermined criterion, by using three operators, selection, crossover, and sequential mutation. The structure with optimum, or the highest fitness value of the last population is selected. In the GA, reproduction is implemented by a selection operator, with the selection based on population improvement or “survival of the fittest” operator, duplicating structures with higher fitness values and deleting structures with lower fitness values. The probability of being duplicated for each gene is defined as:

$$PS_i = \frac{f_i}{\sum_{i=1}^s f_i} \quad (1)$$

Where f_i denotes the fitness function value of the i th chromosome and s is the population number.

This is only one of the many types of selections possible in a GA. If crossovers when combined with selection, yields good components of good structures and combines to give even better structures. The crossover forms $n=2$ pairs of parents if population number is n . Each pair produces two offspring structures to the mutation stage. The offspring are the outcomes of cutting and splicing of the parent structures at various randomly selected crossover points. The approach for selecting crossover points is one point crossover, two-point crossover and a uniform crossover. For instance, the two-point crossover randomly selects the switch point for two parents, switching the genes from the switch point of the two parents to then generate two new offspring.

In contrast, mutation creates new structures that are similar to the current ones, with a small, pre-specified probability randomly altering each component of each structure. For instance, if the third gene of the parent, 1011110, is specified as the mutation gene and the mutation probability is within the threshold value, e.g. 0.001, then the new structure should be 10011110. The reason for using mutation is to prevent missing some significant information during reproduction and crossover, thus preventing a local minimum.

3. MATERIALS AND METHODS

The above sections have indicated the significance of researching subject, as well as providing essential basic information. The proposed model consists of (1) data collection, (2) special pattern model (GFNN).

The objectives of using a GA are to reduce the chance of the FNN getting stuck on a local minimum and to accelerate the testing speed. Thus, the GA is implemented first in order to reach near optimum.

Thereafter, the FNN is employed to fine-tune the results.

3.1. Data Description

Physio Bank database [12] is a large and growing archive of well-characterized digital recordings of physiologic signals and related data for use by the biomedical research community. Physio Bank currently includes databases of multi-parameter cardiopulmonary, neural, and other biomedical signals from healthy subjects and patients with a variety of conditions with major public health implications, including sudden cardiac death, congestive heart failure, epilepsy, gait disorders, sleep apnea, and aging. The databases of normal beat, congestive heart failure beat, ventricular tachyarrhythmia beat, and atrial fibrillation beat were studied in this work. The waveforms of four different ECG beats (normal beat, congestive heart failure beat, ventricular tachyarrhythmia beat, and atrial fibrillation beat) considered in this study are shown in Fig. 1(a)–(d).

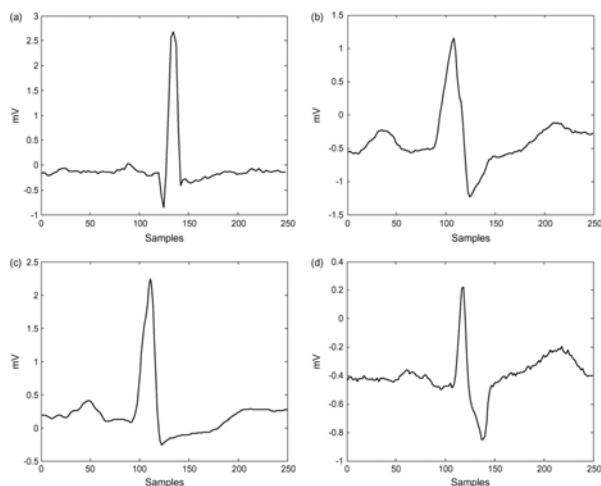


Fig. 1. Waveforms of the ECG beats (a) normal beat, (b) congestive heart failure beat, (c) ventricular tachyarrhythmia beat, and (d) atrial fibrillation beat.

The PhysioBank database contains 50 records. The subjects were 25 men aged 32–89 years, and 25 women aged 23–89 years. Each of the 50 records is slightly over 30 minutes long. In most records, the upper signal is a modified limb lead II, obtained by placing the electrodes on the chest. The lower signal is usually a modified lead V1 (occasionally V2 or V5, and in one instance V4); as for the upper signal, the electrodes are also placed on the chest. This configuration is routinely used for the Physio Bank database. The original analog recordings were made by using nine Del Mar Avionics model 445 two-channel recorders, designated A–I. The analog outputs of the playback unit were filtered to limit analog-to-digital converter (ADC) saturation and for antialiasing, using a band pass from 0.1 to 100 Hz

relative to real time, well beyond the lowest and highest frequencies recoverable from the recordings. The bandpass-filtered signals were digitized at 360 Hz per signal relative to real time by using hardware which were constructed at the MIT Biomedical Engineering Center and at the BIH Biomedical Engineering Laboratory. The sampling frequency was chosen to facilitate implementations of 60 Hz digital notch filters in arrhythmia detectors. The ADCs were unipolars, with 11-bit resolution over ± 5 mV range. Given the sampling frequency and the resolution of the ADC, shown that the difference encoding implies a maximum recordable slew rate of ± 225 mV/s. In practice, this limit was exceeded by the input signals very infrequently, only during severe noise on a small number of records [13].

The ECG signals were divided into two separate data sets- the training data set and the testing data set. In this study, the 360 vectors (90 vectors from each class) were used for training and the 360 vectors (90 vectors from each class) were used for testing. The highest accuracy was obtained by dividing the data into two equal parts for training and testing. The training data set was used to train the GFNN model, whereas the testing data set was used to verify the accuracy and the effectiveness of the trained GFNN model for clustering of the four classes of ECG signals.

3.2. GFNN Architecture

In order to evaluate performance of GFNN model, first the data are obtained through PhysioBank. After this procedure, the collected data can be applied to train the proposed GFNN. The GFNN structure is presented in this study [6]. The GFNN is able to handle the fuzzy inputs, weights and outputs and it basically consists of two components: (1) GA and (2) FNN. Each part is thoroughly discussed in the following subsections:

3.2.1. Initial Weight's Generation through GA

Various attempts have been made to combine GA and ANN's learning for the optimization of the weights and/or topologies of neural networks. Some researchers [14], [15] achieved good results while others [16], [17] found that learning did not help a lot. Sun (2011) used GA's to find possible regions containing the global optimum, and then he used learning as a final fine-tuning operator [18]. The present study applies GA to generate the initial weights for the FNN, which are close to the global optimum. Not only this decreases the testing time, but also it may help to reduce the chance of the FNN getting stuck on a local minimum. Then the FNN-learning algorithm proposed in the next subsection is utilized as a final fine-tuning operator. The procedures for GA are as follows:

Step 1: Randomly generate n structures of population and set up the number of generation and fitness function.

Step 2: Assess the fitness function value of each chromosome.

Step 3: Process the chromosome operation, selection, crossover, and mutation.

Step 4: Evaluate the fitness function of each new chromosome.

Step 5: Eliminate the chromosomes with lower fitness function values and add the new chromosomes with higher fitness function values.

Step 6: If the stop criterion is satisfied, stop; otherwise, go back to Step 3.

In the present study, the fitness function is defined as:

$$F = \frac{N}{\sum_{i=1}^N (T_i - Y_i)^2} \quad (2)$$

Where N denotes the number of population, T_i represents the i th desired output, and Y_i is the i th actual output. The applied coding method is the on the basis of 1/2, since it is the most commonly used binary coding. The size of population in the present study is set to be 50.

3.2.2. Fuzzy Neural Network (FNN)

Fuzzy number operations: Before describing the FNN architecture, the fuzzy numbers and the fuzzy number operations are defined by the extension principle. In the proposed algorithm, real numbers are denoted by lower case letters (e.g. a,b,...) and fuzzy numbers are denoted by upper case letters under a bar (e.g. \bar{A}, \bar{B}, \dots), respectively. Since input vectors, connection weights and output vectors of multi-layer feedforward neural networks are fuzzified in the proposed FNN, the addition, multiplication and nonlinear mapping of the fuzzy numbers are necessary for defining the proposed FNN. Thus, they are defined as follows:

$$\begin{aligned} \bar{Z}(z) &= \bar{X}(x) + \bar{Y}(y) \\ &= \max \{ \bar{X}(x) \wedge \bar{Y}(y) | z = x + y \}, \end{aligned} \quad (3)$$

$$\begin{aligned} \bar{Z}(z) &= \bar{X}(x) \cdot \bar{Y}(y) \\ &= \max \{ \bar{X}(x) \wedge \bar{Y}(y) | z = x \cdot y \}, \end{aligned} \quad (4)$$

$$\bar{Z}(z) = \max \{ \bar{Net}(x) | z = f(x) \} \quad (5)$$

Where $\bar{x}, \bar{y}, \bar{z}$ are the fuzzy numbers, $\bar{\cdot}(\cdot)$ denotes the membership function of each fuzzy number; \wedge is the minimum operator; and $f(x) = (1 + \exp(-x))^{-1}$ is the activation function

of hidden units and output units of the proposed FNN. The α -cut of the fuzzy number \bar{x} is defined as:

$$\bar{X}[\alpha] = \{x | \bar{X}(x) \geq \alpha, x \in \mathbb{R} \quad \text{for } 0 < \alpha \leq 1 \quad (6)$$

Where $\bar{X}[\alpha]$ represents $\bar{X}[\alpha] = [\bar{X}[\alpha]^L, \bar{X}[\alpha]^U]$ and $\bar{X}[\alpha]^L$ and $\bar{X}[\alpha]^U$ are the lower boundary and the upper boundary of the α -cut set $\bar{X}[\alpha]$, respectively.

FNN-learning algorithm: The proposed FNN learning algorithm is similar to an EBP-type learning algorithm. However, some assumptions are needed to be clarified, as follows:

- fuzzify a three-layer feed forward neural network with n_I input units, n_H hidden units, and n_O output units (i.e., input vector, target vectors connection weights and thresholds are fuzzified by using the presented method by Möller et al [19];
- the input vectors are non-negative fuzzy numbers;
- These fuzzy numbers are asymmetric Gaussian shaped fuzzy numbers.

The input- output relation of the proposed FNN (Fig. 2) is shown by the extension principle [6]. Efendigil *et al.* (2009) note the details of ever layer with execution frames [20].

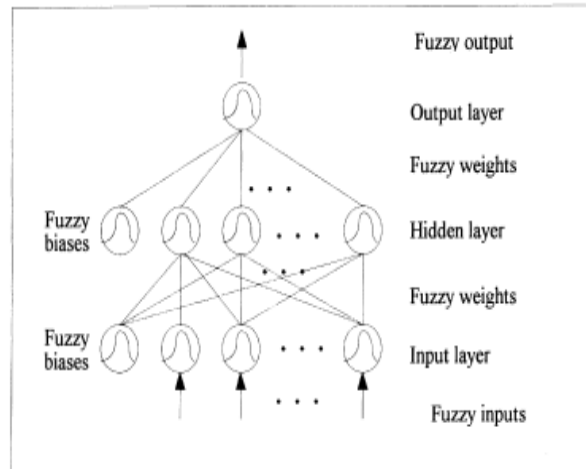


Fig. 2. The FNN architecture

Learning algorithm:

Step 1: Initialize the fuzzy weights and the fuzzy biases through the GA.

Step 2: Repeat Step 3 for $\alpha = \{\alpha_1, \dots, \alpha_n\}$ where α is the α -cut set.

Step 3: Repeat the following procedures for $p=1, 2, \dots, m$, where p is the number of training samples.

- Forward calculation: Calculate the a-cut set of the fuzzy output vector \bar{O}_p corresponding to the fuzzy input vector \bar{X}_p .
- Back-propagation: Adjust the fuzzy weights and the fuzzy biases using the cost function $E_p(\alpha)$. Cost function defined as:

$$E = \frac{1}{2} \sum_p (T_p - O_p)^2 \quad (7)$$

Where T_p and O_p are the target and the actual outputs, respectively.

Step 4: If the stop condition is not satisfied, go to Step 2.

4. DISCUSSION AND RESULTS

Selection of the FNN inputs is the most important component of designing the Fuzzy neural network based on pattern clustering since even the best cluster will perform poorly if the inputs are not selected well. Input selection has two meanings: (1) which components of a pattern, or (2) which set of inputs represent the best given pattern. ECG signals are applied as the input of FNN system and four types of heart beat as output are got.

Therefore, this section will apply the GA to set up firstly, the initial fuzzy weights for the FNN, and then the FNN will fine-tune the weights, clearly accelerating the testing speed.

The setup of the GA for GFNN is as follows:

- The number of genes: 50.
- Crossover rate: 0.2 and 0.5.
- Mutation rate: 0.8 and 0.5.
- Generation replacement type: whole.
- Generation number: 1,000.
- Crossover type: single-point, double-point, and uniform are tested.
- The number of hidden nodes: 3, 4, and 6.

4.1. Setup for initial weights

Based on the above setup, the GA can determine the initial fuzzy weights for the second example, with nine training pairs. The computation results are presented in Tables 1-3. The execution time in the tables is the real time spent by implementing GFNN program and using an IBM compatible PC-586/133 in the Windows-95 environment.

Tables 2 and 3 indicate that the two-point crossover can provide the lowest MSE values after training 1,000 generations. However, the uniform crossover always has a higher MSE value compared to one-point and two-point crossovers. The results are similar to previous research studies. Therefore, this study chooses the two-point crossover to formulate the initial fuzzy

weights for the GFNN. By using the GA computation it is possible to generate identical genes after certain generations. In order to overcome this problem, the present study employs replacement of the whole generation. Thus, the genes can also avoid getting old quickly. As shown in Table 2, the number of different genes decreases from 50 to 30 after 1,000 generations, so this implies that there are only 30 solutions. However, due to computation time, only three groups of weights with the largest fitness functional values are used for fine-tuning through error back propagation type learning algorithm. Since the main objective of using the GA is to determine the "rough" solution for the GFNN, no more different combinations of crossover rates and mutation rates are verified. However, the results shown in Table 1 indicate when the crossover rate and the mutation rate are 0.2 and 0.8, respectively; the fitness function value is at a maximum one.

Regarding the number of generations to run, the study pre-tests different numbers of generations. With 500 generations, the fitness function value is still very large. However, if the training is continues the MSE value of 5,000 generations, only it declines to 0.002, compared with the MSE value of 1,000 generations. However, the computational time is increased by 5 times, indicating that it is not efficient enough for practical applications. Thus, the final conclusion is to choose 1,000 generations as the stop criteria.

4.2. Fine-Tuning for the GFNN Weights

This section uses four groups of weights (best, second, third, and worst) obtained from the GA as the initial weights of the FNN. The results in Table 4 indicate that the GA neither improves nor decreases the MSE value for the network structure $2\pm 3\pm 1$. However, in case of the network structure $2\pm 4\pm 1$, all the testing cases have lower MSE values which are compared to the networks without a GA, except in the worst case. The network structure $2\pm 6\pm 1$ also has similar results, since using the best results from GA. We can always provide the best results through fine-tuning of FNN, if the network topology can be well determined. The determination of network topology can be done by trial and error. In Table 4, the symbols "*" and "#" imply the best and the worst results, respectively, for the corresponding network type.

In addition, Table 4 shows that the percentage decrease in the training epochs for FNN is 95 if it is combined with GA. Since like network structure 2-4-1, it reduces 100,000 training epochs. Basically, every 1,000 training epochs require around 2 minutes. However, the time which is spent on 1,000 generations for the GA is only 6 minutes. This implies that the GFNN can save 14 minutes as compared to FNN. Table 5 shows the difference between GFNN and GA.

In summary, if the network topology can be well-determined and the GA can also be well setup, then the GA can definitely improve the network performance in

both speed and accuracy. However, the way to determine the network structure is trial and error.

Table 1. Four smallest and the worst MSE values after 1,000 generations through one-point crossover

hidden nodes	Crossover rate	Mutation rate	Best	Second	Third	Fourth	Worst	The number of genes	Execution Time (min)
3	0.2	0.8	0.07345	0.07345	0.07366	0.07367	0.1754	29	4
	0.5	0.5	0.05765	0.05747	0.05769	0.05770	0.1731	25	5
4	0.2	0.8	0.03983	0.03999	0.04005	0.04123	0.4465	21	6
	0.5	0.5	0.06342	0.06343	0.06343	0.06399	0.2449	23	6
6	0.2	0.8	0.05321	0.05315	0.05335	0.05312	0.1108	27	7
	0.5	0.5	0.01563	0.01545	0.01572	0.01589	0.1912	27	8

Table 2. Four smallest and the worst MSE values through two-point crossover when crossover rate is 0.2 and mutation rate is 0.8

Hidden nodes	Best	Second	Third	Fourth	Worst	The number of genes	Execution Time (min)
3	0.065452	0.069123	0.076239	0.078987	0.337814	34	5
4	0.021259	0.021834	0.021843	0.021945	0.101256	30	5
6	0.034765	0.034865	0.034901	0.035308	0.067943	30	6

Table 3. Four smallest and the worst MSE values through uniform crossover when crossover rate is 0.2 and mutation rate is 0.8

Hidden nodes	Best	Second	Third	Fourth	Worst	The number of genes	Execution Time (min)
3	0.882029	0.885010	0.886968	0.886977	0.886995	19	6
4	0.886933	0.886935	0.886996	0.886997	0.886999	9	6
6	0.882548	0.886998	0.886998	0.887018	0.891414	11	9

Table 4. The computational results of GFNN

Network type	Source of initial weights	Training MSE	Training epochs	Testing MSE
2-3-1	Best	*0.021347	150	*0.027439
2-3-1	Second	0.022744	100	0.029998
2-3-1	Third	#0.025437	120	#0.030007
2-3-1	Worst	0.023895	150	0.029511
2-3-1	Without GA	0.017431	30,000	0.022451
2-4-1	Best	0.006144	4,200	*0.017654
2-4-1	Second	*0.006140	4,500	0.017662
2-4-1	Third	0.006172	5,000	0.017665
2-4-1	Worst	#0.027919	6,000	#0.02786
2-4-1	Without GA	0.012876	106,000	0.018359
2-6-1	Best	#0.012812	1,500	0.018656
2-6-1	Second	0.012654	1,300	#0.018805
2-6-1	Third	*0.012456	1,200	*0.018543
2-6-1	Worst	0.0012499	1,200	0.018756
2-6-1	Without GA	0.018599	70,000	0.020743

Table 5. The differences between GA and GFNN results

Network type	Source of initial weights	Initial weights MSE from GA	MSE after training using FNN	Percentage decrease
2-3-1	Best	0.068400	0.021998	70.3

2-3-1	Second	0.069700	0.022742	69.8
2-3-1	Third	0.070259	0.025471	64.3
2-3-1	Worst	0.025659	0.023108	91.5
2-4-1	Best	0.027200	0.007412	72.4
2-4-1	Second	0.027867	0.007416	78.8
2-4-1	Third	0.027876	0.07435	78.8
2-4-1	Worst	0.010028	0.027919	72.8
2-6-1	Best	0.038097	0.010639	74.9
2-6-1	Second	0.038215	0.010550	75.0
2-6-1	Third	0.038319	0.010470	77.8
2-6-1	Worst	0.063744	0.010499	82.5
Average				75.7

The confusion matrix showing the clustering results of the GFNN used for clustering of the ECG signals is given in Table 6.

Table 6. Confusion matrix

Desired result	Output result			
	Atrial fibrillation beat	Ventricular tachyarrhythmia beat	Congestive heart failure beat	Normal beat
Normal beat	88	1	0	0
Congestive heart failure beat	2	1	2	87
Ventricular tachyarrhythmia beat	0	1	86	3
Atrial fibrillation beat	0	87	2	0

The test performance of the clusters can be determined by the computation of specificity, sensitivity and total clustering accuracy. The specificity, sensitivity and total clustering accuracy are defined as:

Specificity: number of true negative decisions/ number of actually negative cases.

Sensitivity: number of true positive decisions/ number of actually positive cases.

Total clustering accuracy: number of correct decisions/total number of cases.

A true negative decision occurs when both the cluster and the physician suggest the absence of a positive detection. A true positive decision occurs when the positive detection of the cluster coincided with a positive detection of the physician. The values of the statistical parameters (sensitivity, specificity and total clustering accuracy) were given in Table 7. The total clustering accuracy of the GFNN model was 98.23%.

Table 7. The Values of statistical parameters

ECG beats	Statistical parameters (%)

	Sensitivity	Specificity	Total clustering accuracy
Normal beat	96.57	99.86	98.23
Congestive heart failure beat	94.00	98.28	
Ventricular tachyarrhythmia beat	95.43	98.12	
Atrial fibrillation beat	97.67	99.34	

The values of the statistical parameters (sensitivity and total clustering accuracy) in different methods and the proposed method are shown in Table 8.

Table 8. Comparison between different methods

Clusters	Statistical parameters (%)	
	sensitivity	accuracy
Block-Based neural networks [21]	74.9	94.5
Non-singleton fuzzy logic (NSFLC) [7]	89.1	97.55
KFCM-ACOR [9]	85	92.768
KFCM [9]	73.67	87.196
FCM [9]	78.45	86.74
k-NN (k = 5) [9]	88.15	94.39
k-NN (k = 3) [9]	88.76	94.98
Fuzzy cluster algorithm [10]	75	92
Cellular compact genetic algorithm (c-cGA) [11]	73.6	96.0
Proposed method (GFNN)	91.36	98.23

By comparing of different methods with the proposed method, we find out that the proposed cluster in this paper acts better than other clusters. There are also many advantages in this method. So, it utilizes the proposed fuzzy neural network with initial weights generated by genetic algorithm (GFNN) to improve testing speed, accuracy and to reduce the chance of the FNN getting stuck on a local minimum and to analyze uncertainties, accurately.

Using GFNN for clustering of ECG signals is very useful because Neuro-fuzzy system includes of fuzzy logic and ANNs, by utilizing the mathematical properties of ANNs in tuning rule-based fuzzy systems that approximate the way human process information (artificial neural networks (ANNs) theory determines properties (fuzzy sets and fuzzy rules) and also Fuzzy set theory plays an important role in dealing with uncertainty when making decision in medical applications) and also we use GA to reach near optimum. As a result, the accuracy in this method. Also, it is 98.23% that is higher and better than the accuracy of other methods.

5. CONCLUSION

This paper presented a new application of GFNN model employing GA for clustering of the ECG signals. The presented GFNN model combined the fuzzy neural network capabilities and the genetic algorithm approach. The ECG signals that are obtained from different patients signals have been used as GFNN inputs while GFNN has clustered ECG signals to identify of the heart diseases as soon as possible. Model evaluation results indicate that proposed GFNN model can perform more accurately, less testing speed than the conventional statistical methods, a single ANN and FNN. Also, It has a lower chance of the FNN getting stuck on a local minimum.

The clustering results and statistical measures were used for evaluating the GFNN. The total clustering accuracy of the GFNN model was 98.23%. The obtained results demonstrated that the proposed GFNN model can be used in clustering the ECG signals by taking into consideration for the misclustering rates.

REFERENCES

- [1] S. C., Saxena, V. Kumar and V. Hamden, "Feature extraction from ECG signals using wavelet transforms for disease diagnostics," *International Journal of Systems Science*, Vol. 33, No. 13, pp. 1073–1085, 2002.
- [2] Y. C., Yeh, W. J., Wang and C. Wun Chiou, "Feature selection algorithm for ECG signals Using Range-Overlaps Method," *Expert Systems with Applications*, Vol. 37, Issue 4, pp. 3499–3512, 2010.
- [3] Y. C., Yeh, W. J., Wang and C., Wun Chiou, "Heartbeat Case Determination Using Fuzzy Logic Method on ECG Signals," *International Journal of Fuzzy Systems*, Vol. 11, No. 4, pp. 401-406, 2009.
- [4] T. M., Nazmy, H., El-Messiry and B., Al-Bokhit, "Adaptive Neuro-Fuzzy Inference System for Classification of ECG Signals," *Journal of Theoretical and Applied Information Technology*, Vol. 56, No. 14, pp. 568–54, 2010.
- [5] Z. C., Yu, G. Y., Wang, S. Y., He, Y. L., Tan and N., Xu, "The identification of ventricular escape beat based on wavelet transform and BP Neural Network," *IEEE Transactions on Neural Networks*, Vol. 15, No. 5, pp. 111-125, 2011.
- [6] R. A., Aliev, B. G., Guirimov, B., Fazlollahi and R. R., Aliev, "Evolutionary algorithm based learning of fuzzy neural networks. Part 2: Recurrent fuzzy neural networks," *Fuzzy Sets and Systems*, Vol. 160, No. 17, pp. 2553–2566, 2009.
- [7] J., Nasiri, M., Naghibzadeh, H., Yazdi, B., Naghibzadeh, "ECG arrhythmia classification with support vector machines and genetic algorithm," In: *Third UKSim European Symposium on Computer Modeling and Simulation*, 2009. EMS '09, pp. 187–192. IEEE Computer Society, Los Alamitos, doi:10.1109/EMS.2009.39.
- [8] T. W., Chua, and W. W., Tan, "Non-singleton genetic fuzzy logic system for arrhythmias classification," *Engineering Applications of Artificial Intelligence*, pp. 251-259, 2011.
- [9] B., Doğan, and K., Mehmet, "A new ECG beat clustering method based on kernelized Fuzzy c-means and hybrid ant colony optimization for continuous domains," *Applied soft computing*, pp. 3442-3451, 2013.
- [10] H. H., De Carvalho Junior, R. L., Moreno, T. C., Pimenta, P. C., Crepaldi, and E., Cintra, "A heart disease recognition embedded system with fuzzy cluster algorithm," *Computer Methods and Programs in biomedicine*, pp. 447-454, 2013.
- [11] Y., Jewajinda, and Ch., Prabhas, "A parallel genetic algorithm for adaptive hardware and its application to ECG signal classification," *Neural Computing and Applications*, pp. 1-18, 2013.
- [12] D. E., Goldberg, "Genetic Algorithms in Search. Optimization and Machine Learning," *Addison-Wesley*, Reading, MA, 1989.
- [13] A. L., Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. Ch. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C. K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals Circulation," Vol. 23, No. 101, pp. e215–e220, 2000, Available from: <http://circ.ahajournals.org/cgi/content/full/101/23/e215>.
- [14] P. C., Chen, C. C., Wu, W. L., Chiang and K. Yeh, "A Novel Stability Condition and its Application to GA-Based Fuzzy Control for Nonlinear Systems with Uncertainty," *Journal of Marine Science and Technology*, Vol. 17, No. 4, pp. 293-299, 2009.
- [15] E. A., Hossam, A., Abdennour and A. S., Abdulaziz, "Design and experimental investigation of a decentralized GA-optimized neuro-fuzzy power system stabilizer," *Electrical Power and Energy Systems*, Vol. 31, No. 32, pp. 751–759, 2010.
- [16] E. N., Mitrakis, C. A., Topaloglou, T. K., Alexandridis, J. B., Theocharis and C. Zalidis, "Decision Fusion of GA Self-Organizing Neuro-Fuzzy Multilayered Classifiers for Land Cover Classification Using Textural and Spectral Features," *IEEE Transactions on Geosciences and Remote Sensing*, Vol. 46, No. 7, pp. 121-127, 2008.

- [17] H., Hamsa Haseena, and J. K., Pau, **“Fuzzy Clustered Probabilistic and Multi Layered Feed Forward Neural Networks for Electrocardiogram,”** *Journal of Medical Systems*, Vol. 35, No. 2, pp. 179-188, 2009.
- [18] S., Xin, Y. Zheng, Y. Pang, C. Ye and L. Zhang, **“The Application of Neural Network Model Based on Genetic Algorithm for Comprehensive Evaluation, High Performance Networking, Computing, and Communication Systems,”** *Communications in Computer and Information Science*, Vol. 163, No. 1, pp. 229-236, 2011.
- [19] B., Möller, M. Beer, W. Graf and J.-U, **“Fuzzy finite element method and ITS application,”** *Trends in Computational Structural Mechanics*, Cimne, Barcelona, Spain, 2001.
- [20] T., Efendigil, S. Önüt and C. Kahraman, **“A decision support system for demand forecasting with artificial neural networks and neuro-fuzzy models: A comparative analysis,”** *Expert Systems with Applications*, April, Vol. 36, No. 3, Part 2, pp. 6697–6707, 2009.
- [21] W., Jiang, S. G., Kong, **“Block-Based neural networks for personalized ECG signal classification,”** *IEEE Trans Neural Netw*, Vol. 18, No. 6, PP. 1750–1761, 2007.