Available online at www.sciencedirect.com

ScienceDirect

ELSEVIER

Journal of Biomedical Informatics 41 (2008) 874–881

Journal of Biomedical Informatics

www.elsevier.com/locate/yjbin

A new arrhythmia clustering technique based on Ant Colony Optimization

Mehmet Korürek *, Ali Nizam [1]

Electrical and Electronics Engineering Faculty, Department of Electronics and Communication Engineering, Istanbul Technical University, 34469 Istanbul, Turkey

Received 29 June 2007
Available online 23 February 2008

Abstract

In this paper, a new method for clustering analysis of QRS complexes is proposed. We present an efficient Arrhythmia Clustering and Detection algorithm based on medical experiment and Ant Colony Optimization technique for QRS complex. The algorithm has been developed based on not only the general signal detection knowledge, but also on the ECG signal's specific features. Furthermore, our study brings the power of Ant Colony Optimization technique to the ECG clustering area. ACO-based clustering technique has also been improved using nearest neighborhood interpolation. At the beginning of our algorithm, we implement signal filtering, baseline wandering and parameter extraction procedures. Next is the learning phase which consists of clustering the QRS complexes based on the Ant Colony Optimization technique. A Neural Network algorithm is developed in parallel to verify and measure the success of our novel algorithm. The last stage is the testing phase to control the efficiency and correctness of the algorithm. The method is tested with MIT-BIH database to classify six different arrhythmia types of vital importance. These are normal sinus rhythm, premature ventricular contraction (PVC), atrial premature contraction (APC), right bundle branch block, ventricular fusion and fusion. Our simulation results indicate that this new approach has correctness and speed improvements.

Keywords: ECG; Ant Colony Optimization (ACO); Clustering; Neural networks; Arrhythmia detection; *k*-Nearest neighborhood classifier

1. Introduction

Computer-based analysis, classification and interpretation of electrocardiography (ECG) arrhythmias have been the subject of considerable research effort in the recent years. Besides, one of the most difficult problems in these researches is the large variation in the morphologies of ECG waveforms, not only of different patients or patient groups but also within the same patient. The ECG waveforms may differ for the same patient to such an extent that they are dissimilar to each other and at the same time are alike for different types of beats [1]. In the last decades, cluster analysis has been combined with other techniques, and has been used to overcome these difficulties in many areas of ECG processing, such as classification of ECG arrhythmias [1–6], ECG feature selection [7], ECG character points detection [8], and classification of QRS morphology [9].

One of the other major problems in automatic ECG analysis is the difference between algorithmic approaches and cardiologist's point of view. Most of the researches try to cluster data into the same groups that are labeled by cardiologists and attempt to make cluster results consistent with clinical categories. However, because of the variations of ECG complexes' morphology, overlapping between clinical categories and cluster results often occurs. The ECG clustering methods, and the relationship between clustering results and clinical categories need further investigation [6].

The objective of this work is to achieve a better clustering analysis of ECG complexes using a new method that

* Corresponding author. Fax: +90 212 285 35 65.
E-mail addresses: korurek@itu.edu.tr (M. Korürek), anizam@iski.gov.tr (A. Nizam).
[1] Fax: +90 212 321 77 51.

1532-0464/$ - see front matter © 2008 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2008.01.014

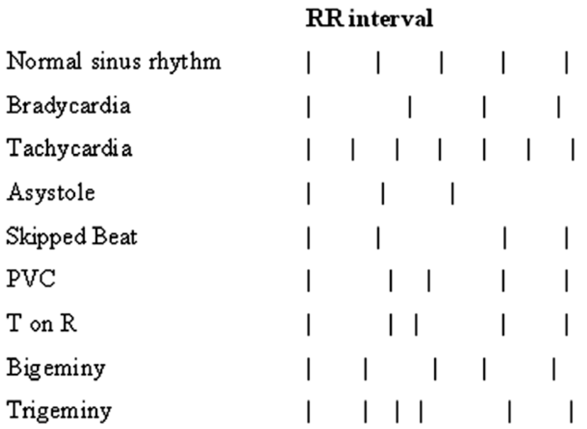


Fig. 1. The relation ship between RR interval and arrhythmia type [10].

aimed on closing the gap between general signal detection techniques and medical knowledge. Theoretically, a normal ECG is assumed to be a periodical signal in a long interval of time according to its period. As mentioned before, many modern researchers have attempted to analyze the ECG signals periodically using just one QRS period's parameters. However, these approaches are too general to solve classification problems about QRS morphologies. According to the medical point of view, the detection of arrhythmia depends on two or more ECG signal periods [23] as shown in Fig. 1 [10]. The previous period of an ECG signal has many indicators of current arrhythmia. For example, detecting arrhythmia types including PVC, T on R and Bigeminy depends on two or three periods of the ECG signal as shown in Fig. 1.

So in our approach, two QRS periods' parameters are considered: RRt_n, $RRt_{(n+1)}$. Also, QRS amplitude, $QRSht_n$, is used as the third parameter as shown in Fig. 2. We also used QSR area and QT interval features in order to distinguish right bundle branch block and fusion arrhythmias. RR parameter is used for controlling rhythm change. And QRS amplitude parameter indicates the correlation of rhythm and amplitude. We compared the results of one and two periods approach from many perspectives in Section 4.

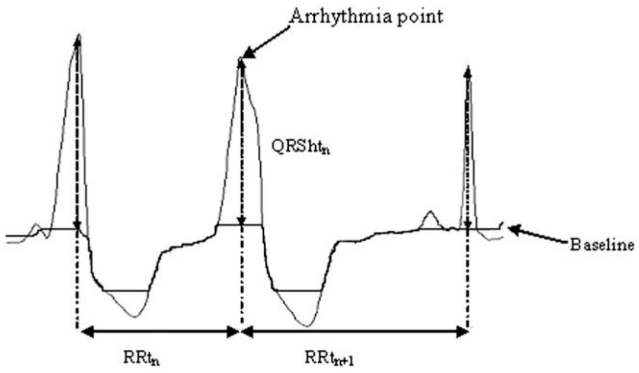


Fig. 2. Two QRS periods and the chosen parameters of an ECG signal [22].

For implementing our methodologies, firstly, QRS signal is separated from the noise using an FIR filter. And then QRS baseline correction algorithm consisting of a median filter approach is used. RR distance and RR height parameters are extracted temporally. After that, a novel clustering algorithm based on Ant Colony Optimization (ACO) is developed for clustering arrhythmia types. A Neural Network algorithm is developed in parallel to verify and measure our novel algorithm success. The last step is the *k*-nearest neighborhood classification algorithm which is used to test the algorithm's efficiency mathematically.

2. Model

2.1. General system model

The system contains learning and test phases. The learning phase consists of three stages and is constructed as shown in Fig. 3. The first stage of learning phase involves filtering the QRS signal to remove noise effect, baseline correction and parameter extraction. The second stage involves the storage of information over Relational Database Management System (RDBMS) [11]. The last stage involves the clustering of QRS parameters using the ACO technique to construct arrhythmia classes. A Neural Network algorithm is developed in parallel with our new approach to verify and measure the success of our novel algorithm.

Test or control phase is used to control algorithm's efficiency and correctness as shown in Fig. 4. The first stage of testing phase is cleaning the QRS signal, baseline detection and parameter extraction. Previously built cluster information serves as an input to the testing system.

2.2. Preprocessing

Preprocessing stage contains filtering, baseline detection, QRS complex detection and normalization. A low pass linear phase filter is applied for impulsive noise. Filter characteristic is shown in the following Eqs. (1)–(5):

$$y = F(b, a, x) \tag{1}$$

$$\begin{aligned} y(n) = b(1) * x(n) + \ldots + b(nb+1) * x(n-nb) \\ - \ldots - a(na+1) * y(n-na) \end{aligned} \tag{2}$$

$$B = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -1] \tag{3}$$

$$A = [1 \quad -1] \tag{4}$$

If we chose the z domain to show the filter function, then

$$Y(z) = \frac{1 - z^{-6}}{1 - z^{-1}} X(z) \tag{5}$$

After removing noise from the ECG signal, a median filter with 500 samples wide is designed for baseline estimation as shown in Fig. 2.

After baseline correction, an adaptive QRS detection algorithm is used. Adaptive threshold is applied to the rectified signal. ECG time samples and a window function are

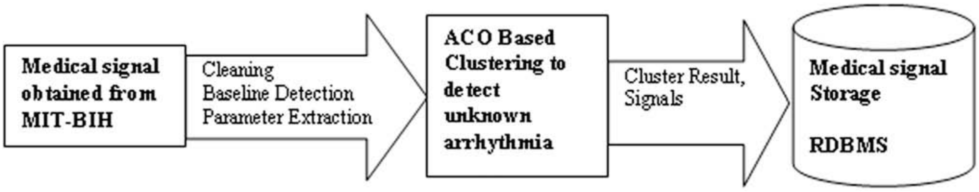


Fig. 3. Training system model.

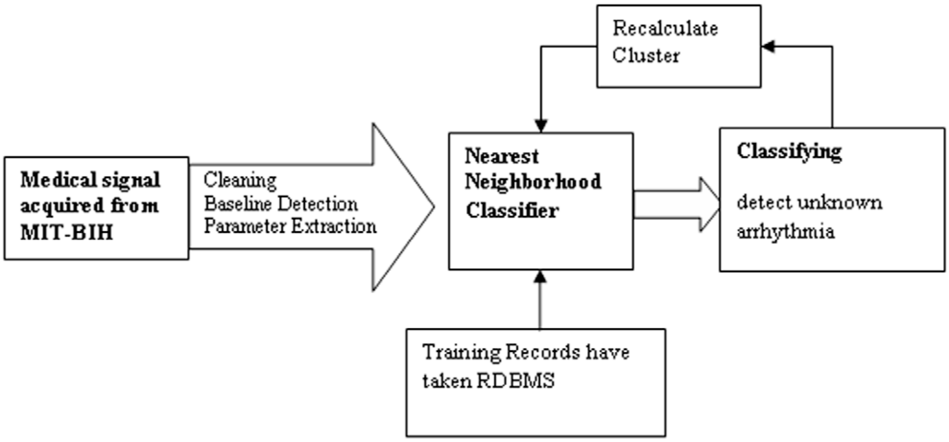

Recalculate Cluster
Medical signal acquired from MIT-BIH
Cleaning
Baseline Detection
Parameter Extraction
Nearest Neighborhood Classifier
Classifying
detect unknown arrhythmia
Training Records have taken RDBMS

Fig. 4. Testing system model.

convolved. This window is 100 samples length and the height of the window is 40% of the mean of the last four R wave's height. The place of the sample which has maximum height value at the inside of the window shows the candidates of the R point place. After detecting the candidates of the R point, a control vector, which covers 20 samples before the R wave peak and 20 samples after that, is used to verify the real R point place according to the QRS shape. After real R point is verified, it is used as the reference point and the next 100 samples following this point are discarded (closing eye) from researching a new R point.

2.3. Normalization of ECG signals

ECG signal parameters are normalized according to the last eight normal ECG records. Normalization is done using the following equations:

$$QRShm_{(t)} = \frac{\sum_{i=t-7}^{t} QRSh(i)}{8}; \quad QRSh(t) \in \text{‘}NormalECG\text{’} \qquad (6)$$

$$RRm_{(t)} = \frac{\sum_{i=t-7}^{t} RR(i)}{8}; \quad RR(t) \in \text{‘}NormalECG\text{’} \qquad (7)$$

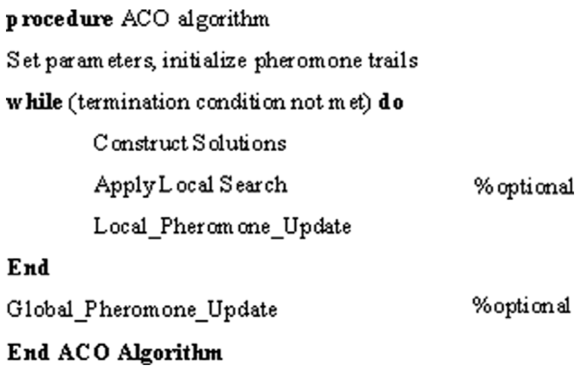
2.4. Clustering using Ant Colony Optimization technique

2.4.1. ACO algorithm

Inspired by the collective behavior of an ant colony, Dorigo designed the ant system (AS) [12], and later continued to develop this system [13–16]. The ACO technique has emerged recently as a novel meta-heuristic in the class of naturally derived problem solving strategies (other categories include neural networks, simulated annealing, and evolutionary algorithms). The AS optimization algorithm is basically a multi-agent system in which low level interactions between single agents (i.e., artificial ants) result in the complex behavior of the colony as a whole. AS optimization algorithms are inspired by the behavior of real ants, which deposit a chemical substance (called pheromone) on the ground. Ants lay pheromone (in varying quantities) on the ground as they move, thus creating a trail. The general parts and structure of ACO algorithm are shown in Fig. 5.

2.4.2. The proposed algorithm for clustering

ACO technique can be applied to clustering successfully. The ACO algorithm has been utilized with different favors to solve the clustering problem [17]. In this approach's first stage, ants visit other cities randomly and they lay the pheromone according to inverse proportionality of Gaussian



```
procedure ACO algorithm
Set parameters, initialize pheromone trails
while (termination condition not met) do
        Construct Solutions
        Apply Local Search                    %optional
        Local_Pheromone_Update
End
Global_Pheromone_Update                       %optional
End ACO Algorithm
```

Fig. 5. The ACO algorithm's general structure [16].

distance. After several iterations (cycles), trail intensity (pheromone) between close nodes of trails will be increased; on the other hand, the trail intensity (pheromone) far between the nodes of trails will be decreased. In the second stage, ants will favor to visit the closer nodes and then reinforce the trail with their own pheromone. Every ant only needs to visit (1/10) cities not all of the cities, then the ants decreasingly visit the cities every time. Finally, a number of clans (clusters) will be built. The tournament selection technique is used for a proportionate selection mechanism, and the selection of a new node is based on randomly selection of some lines among the available lines; then, selection of the shortest line among the previous randomly selected lines is continued as shown in Fig. 6. The choice of the previous path |XS| as a next path is prohibited.

Our study is based on these techniques, and some improvements have been implemented. Iteration number is smartly increased and the first and second stages are integrated. Besides, algorithm simplicity is also increased.

The proposed ACO algorithm for clustering can briefly be illustrated as follows. The ACO clustering algorithm usually consists of the following steps:

Step 1: Initialize: Input n data sets and assign m ants randomly to m nodes, and initially m is equal to $n/10$.
Step 2: Find the candidate nodes the next time for the ants to visit. These nodes are chosen randomly.
Step 3: Each ant visits the other nodes according to the nearest neighborhood interpolation depending on the Gaussian distribution. Select the nearest neighborhood node.
Step 4: Update the pheromone quantity of visited trail.
Step 5: Repeat Step 2 through Step 5 until the iteration number is reached.
Step 6: Perform clustering using the value of pheromone quantity.

2.5. Classification of ECG features using clustering Neural Network

Neural Network is one of the most used methods of ECG beat recognition and classification [1–3,5,6,18]. Multi-layer perception (MLP) [19–21] based on the neural networks has been chosen to be able to recognize and classify the ECG signals [3]. In our study, a backpropagation-based Neural Network algorithm is developed in parallel to verify and measure the success of our algorithm. The structure of such a network is presented in Fig. 7. Different samples sets having time domain features are selected for each arrhythmia type.

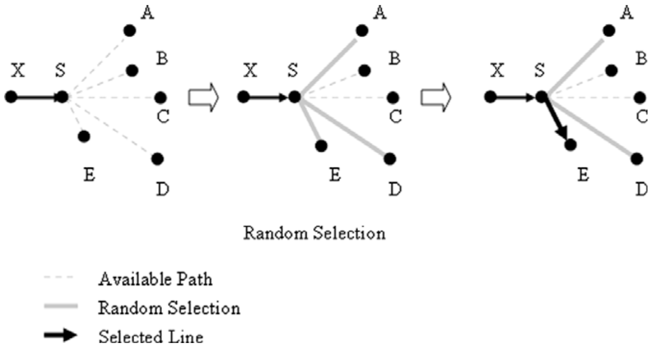


Fig. 6. The tournament selection mechanism of ACO algorithm [17].

Neural network structure consists of (number of input parameters) × 10 × 25 × 6) nodes. Feed forward algorithm is used for training.

Tan-sigmoid transfer function tansig is used in the hidden layers and the linear transfer function purelin is used in the output layer. The characteristics of tansig and purelin are shown in Fig. 8a and b.

2.6. k-Nearest neighborhood classifier

Nearest neighborhood classifier is used for testing purposes. After a new beat is input to the system, the cluster list is scanned to determine the clusters to which it belongs. If there are two or more such clusters, the beat is placed into the cluster which has the maximum number of nearest members. k is the total number of beat's nearest neighbors and chosen as 5 in this work.

Step 1: Set the member list of the clusters.
Step 2: Take the next input vector and find its k nearest members according to the minimum distance measure.

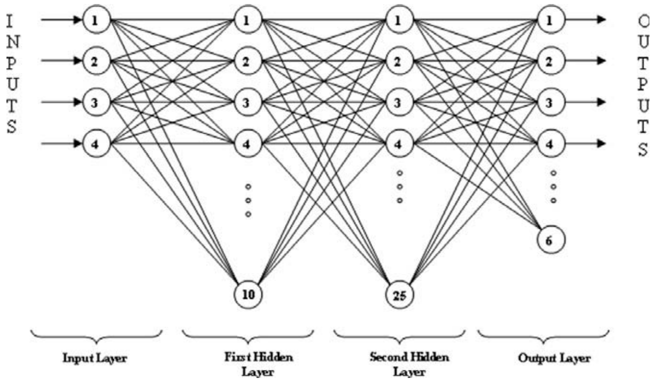


Fig. 7. Optimum Neural Network architecture used for classification of ECG signals.

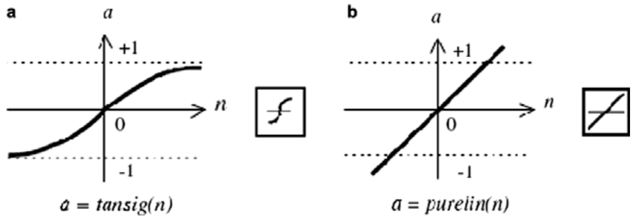


Fig. 8. Function characteristics of NN transfer functions. (a) tansig used in hidden layers; (b) purelin used in output layer.

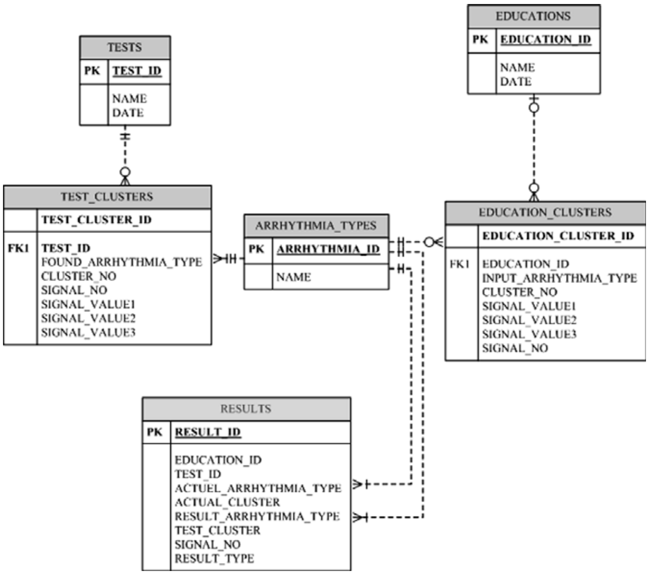


Fig. 9. The relational data model of system.

Step 3: Find the cluster which the maximum number of nearest members belong.
Step 4: Assign this cluster as the input vector's cluster.
Step 5: Repeat 2 to 4 for all input vectors.

2.7. Relational data model of system

A generic relational data model of system is developed to store the necessary information as shown in Fig. 9. Trained set is written to the database and the RDBMS stores processed medical data and the results of the tests and look-up definitions. Using RDBMS, many different types of learning sets and testing sets can be obtained parametrically with the help of SQL commands easily. Clustering results are written to the RDBMS also.

The education and tests information are stored in the EDUCATION and TEST tables. EDUCATION and TEST tables have ID columns for referencing the special instances and using in the future works. ARRHYTHMIA_TYPES table is used to store arrhythmia types as a lookup table. EDUCATION_CLUSTERS table is used to store learning phase results. Each signal point's value is stored in separate columns as SIGNAL_NO, SIGNAL_VALUE1, SIGNAL_VALUE2 and SIGNAL_VALUE3. TEST_CLUSTERS table is used to store test results, which shows the found cluster information. The results are stored in a separate table named RESULTS. RESULTS table data are populated programmatically, and there is no physical relationship between RESULTS, EDUCATION_CLUSTERS, and TEST_CLUSTERS tables because of increasing flexibility. RESULT_TYPE column in RESULTS table is used to store success or failure of test.

When an education result's success is investigated in the system, it can be done using a simple SQL command as follows:

```
SELECT success. total_number / all.total_number total_success_factor
  (SELECT COUNT(*) total_number
  FROM results
  WHERE actuel_arrhythmia_type=result_arrhythmia_type
      AND education_id=:p_education_id) success,
  (SELECT COUNT(*) total_number
   FROM results
   WHERE education_id=:p_education_id) all


(:p_education_id is the variable of education instance)
```

(8)

3. Methods and results

3.1. Software program

A Windows-based real-time software program was written using MATLAB®. Fig. 10 shows the form of software. The software project supports learning and testing phases from one interface. The program is initiated by clicking on "Start Learning" upon which the ECG data are obtained from the MIT-BIH database.

Clustering results can be viewed visually on the screen of the software. Different arrhythmia types are shown with different symbols and different clusters of arrhythmia types are shown with different grades.

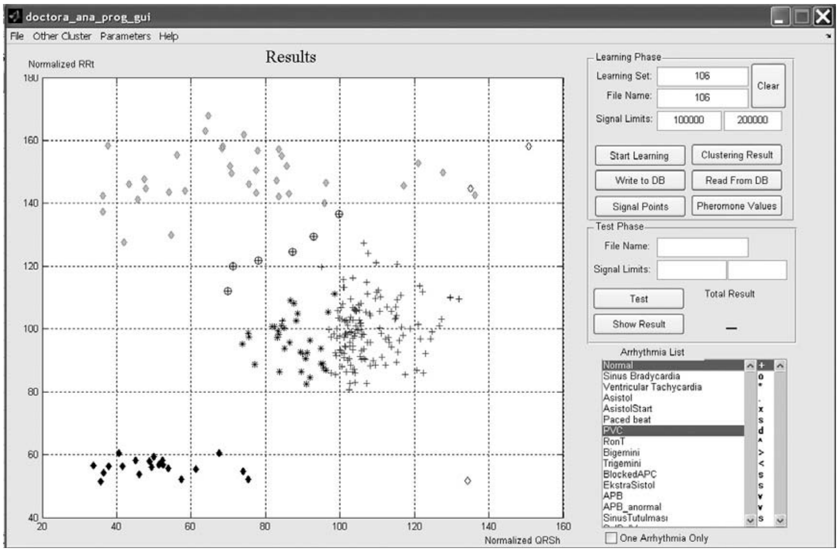
A program has been developed on modular architecture. Each algorithm is developed in a separate MATLAB® function. The modular classifier approach allows the parallel development of all component classifiers. Moreover, we choose to divide the training data samples into different data subsets, and therefore further reduction of the training time is obtained for individual classifiers.

Oracle XE®, which is Oracle® corporation free database product, is used as RDBMS.

3.2. Result

The ECG signals are taken from MIT-BIH [22] ECG database, which are used to classify six different arrhythmias for training. Sampling frequency is $f = 360$ Hz. These are normal sinus rhythm, premature ventricular contraction (PVC), atrial premature contraction (APC), right bundle branch block, ventricular fusion and fusion. These records have been taken for 32 different patients.

Totally 8771 ECG periods are used to test our algorithms' correctness. The distribution of records is shown in Table 1. Different training and testing sets have been built from these record sets using random sampling. When choosing these sets, the numbers of training and testing sample sizes have been considered to reflect the actual number of arrhythmias inside the medical area. The number of *i*. Arrhythmia samples in the training set is calculated using



Learning Set: 106

File Name: 106

Signal Limits: 100000 200000

Fig. 10. A sample clustering result is shown on the interface of software.

Table 1
The number of sample records according to arrhythmia type

Arrhythmia type	Number of samples
Normal	6622
Fusion (of paced and normal beat)	127
PVC	1395
Right bundle branch block beat	127
APC, atrial premature beat	323
Fusion of ventricular and normal beat	177
Total	8771

$$\text{Arrhythmia Number in Set}(i) = \text{round}\left(r * \frac{\text{The Total Number Of Arrhythmia Type}(i)}{\text{Total Number Of Samples}} * \mu\right) \quad (9)$$

where r is the selected number of samples in the training set. The rest of the samples are used for testing purposes. μ is the deviation constant used for creating different record sets. Its value changes $1 \pm 0, 25$ (default is one).

In our works, sensitivity is considered as the most critical success factor. The description of sensitivity is the fraction of real events that are correctly detected in

$$\text{Sensitivity} = 100^{*}(\text{TP}/(\text{TP} + \text{FN})) \quad (10)$$

True positive (TP) means that the number of the true events of arrhythmia that has been successfully detected, and false negative (FN) is the count of missed beats. This is because missing a life-threatening ECG beat is considered to be more serious than missing a few false alarms which can later be screened out manually [23]. The total sensitivity of an arrhythmia class is calculated as follows:

Table 2
The sensitivity analysis results from patient test data classified by NN using one period's parameters

Training set	Normal		PCV		APC		Right bundle branch block		Ventricular fusion		Fusion		Sensitivity(%)
	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	
1	5762	553	1129	117	113	41	62	2	55	30	35	28	90.03
2	5651	746	1079	179	126	35	76	1	64	21	31	32	87.40
...	...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...
25	5762	527	1053	205	124	30	63	1	67	18	39	24	89.80

Table 3
The sensitivity analysis results from patient test data classified by ACO using one period's parameters

Training set	Normal		PCV		APC		Right bundle branch block		Ventricular fusion		Fusion		Sensitivity(%)
	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	
1	5875	440	1108	138	98	56	62	2	56	29	30	33	91.21
2	5759	638	1115	143	102	59	75	2	56	29	38	25	88.90
...	...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...
25	5847	442	1118	140	102	52	63	1	56	29	35	28	91.25

Table 4
The sensitivity analysis results from patient test data classified by NN using two periods' parameters

Training set	Normal		PCV		APC		Right bundle branch block		Ventricular fusion		Fusion		Sensitivity(%)
	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	
1	5970	345	1144	102	132	22	63	1	64	21	47	16	93.60
2	5879	518	1185	73	150	11	74	3	65	20	48	15	92.00
...	...	...	...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...	...	...	...
25	5954	335	1143	115	142	12	64	0	67	18	52	11	93.81

Table 5
The sensitivity analysis results from patient test data classified by ACO using two periods' parameters

Training set	Normal		PCV		APC		Right bundle branch block		Ventricular fusion		Fusion		Sensitivity(%)
	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	TP	FN	
1	6073	242	1171	75	132	22	62	2	61	24	33	30	95.02
2	6032	365	1163	95	137	24	74	3	65	20	41	22	93.42
...	...	...	...	...	...	...	...	...	...	...	...	...	
...	...	...	...	...	...	...	...	...	...	...	...	...	
25	6039	250	1177	81	138	16	63	1	64	21	42	21	95.07

Table 6
The comparison of system wide sensitivity results

Algorithm type		Normal	PCV	APC	Right bundle branch block	Ventricular fusion	Fusion	Total (%)
NN classifier	Two periods	93.70	92.29	90.41	98.05	76.86	77.78	93.05
	One period	90.39	86.68	77.40	98.05	72.94	55.56	88.90
ACO based	Two periods	95.49	93.33	86.78	97.07	74.51	84.06	94.40
	One period	92.00	88.81	64.39	97.56	65.88	59.01	90.50

Table 7
The comparison of computational time

	The number of iteration for each data set	Time for one period's parameters used (s)	Time for two period's parameters used (s)
Neural Network classifier	10000	550	600
ACO	1000	15	18

$$\text{Total Sensitivity of an Arrhythmia}(i) = \frac{\sum_{k=1}^{m} TP(k,i)}{\sum_{k=1}^{m} TP(k,i) + \sum_{k=1}^{m} FN(k,i)} * 100 \qquad (11)$$

where TP(k,i) is the number of TP belonging to the ith arrhythmia in kth test set and m is the number of test sets. The total sensitivity of the system belonging to all arrhythmia classes is calculated as follows:

$$\text{Total Sensitivity of the System} = \frac{\sum_{k=1}^{m} TTP(k)}{\sum_{k=1}^{m} TTP(k) + \sum_{k=1}^{m} TFN(k)} * 100 \qquad (12)$$

where TTP(k) is the total number of TP in the k. test set.

The simulation results are listed in Tables 2–5.

These are normal sinus rhythm, premature ventricular contraction (PVC), atrial premature contraction (APC), right bundle branch block, ventricular fusion and fusion (see Table 1).

The total result is shown in Table 6 (after mean of 25 trials).

As shown in Table 6 using the ACO algorithm and two period parameters together improves correctness of QRS arrhythmias at a higher level.

A comparison of the computational time result is shown in Table 7 (after mean of 25 trials).

The MSE training error is 10^{-3} for Neural Network system. As shown in Table 7, ACO algorithm outperforms NN on the same hardware. All of the results have been stored in a computer which has a 3 GHz P4 processor, and 1 GB RAM.

In this work, every beat is normalized with preceeding normal beats of the signal. For this reason, any arrhythmic beat should have at least two preceeding normal beats. It is difficult to find out enough training samples for some arrhythmia types in MITBIH data base that ensure this condition. In order to overcome this condition, we collected as much normal beats as possible before the arrhythmic beats even they are not so nearby.

Every tested arrhythmia should be learned before by the classifiers by means of its partners. It is important to note that; ACO is over YSA, with compared to their detection time, the stability of the results and the successes of the classification.

4. Future discussion

In this work, we have shown that the ECG signals' analyze results can be improved using the medical signals' special properties. Moreover, our results show that our new approach outperforms other methodologies, especially for arrhythmias which occur depending on two or more ECG signal periods. In our study, time dependent morphological parameters are used as features. In future, using different parameter sets such as wavelet parameters within our approach, correctness of classification can be improved. Furthermore, the proposed techniques have been developed for ECG signals. But it can be applied to the different types of data such as EEG and EMG analysis. Besides, the usage of ACO in the biomedical area is relatively new and needs further investigations.

We used RDBMS as a medical signal storage system, and we show the benefits of RDBMS. We propose a generic data model for medical signals and use SQL to create many different types of data sets. In the future, the role of RDBMS in medical storage area needs further investigation, and the RDBMS can store and handle different types of medical signals natively. So the role of RDBMS will be increased in the medical area.

References

[1] Osowski S, Linh TH. ECG beat recognition using fuzzy hybrid neural network. IEEE Trans Biomed Eng 2001;48(11):1265–71.

[2] Ozbay Y, Karlik B. A recognition of ECG arrhythmias using artificial neural network. In: Proceedings of the 23rd Annual Conference Istanbul: IEEE/EMBS; 2001:1680–3.

[3] Ozbay Y, Ceylan R, Karlik B. A fuzzy clustering neural network architecture for classification of ECG arrhythmias. Comput Biol Med: Elsevier 2006;36:376–88.

[4] Lagerholm M, Peterson C, Braccini G, Edenbrandt L, Sornmo L. Clustering ECG complexes using hermite functions and self-organizing maps. IEEE 2000;47(7):838–48.

[5] Gao D, Madden M, Chambers D, Lyons G. Bayesian ANN classifier for ECG arrhythmia diagnostic system: a comparison study. In: Proceedings of International Joint Conference on Neural Networks, Montreal, Canada IEEE, July 31–August 4, 2005:2383–8.

[6] Wenyu Y, Gang L, Ling L, Qilian Y. ECG analysis based on Pca and Som. In: IEEE International Conference on Neural Networks & Signal Processing: Nanjing. China; 2003:14–7.

[7] Zhou SH, Rautaharju PM, Calhoun HP. Selection of a reduced set of parameters for classification of ventricular conduction defects by cluster analysis. Proc Comput Cardiol IEEE 1993:879–82.

[8] Shey JT, Lin K-P, Chang WH. Measurement 12-lead ECG character points using line segment clustering technique. In: Proceedings of 20th Ann. Int. Conference IEEE. vol. 20. Hongkong, China: Engineering in Medicine and Biology Society; 1998. p. 223–6.

[9] Dickhaus H, Gittinger J, Maier C. Classification of QRS morphology in holter monitoring. In: Proceedings of First Joint BMES/EMBS Conference IEEE; Atlanta, USA; 1999. p. 270.

[10] Tompkins WJ, Webster JG, editors. Design of microcomputer-based medical instrumentation. Englewood Clifs, NJ 07632: Prentice Hall Inc; 1981, ISBN: 0-13-201244-8. 398–9.

[11] Codd EF. A relational model of data for large shared data banks. Comm ACM 1970;13(6):377–87.

[12] Dorigo M, Maniezzo V, Colorni A. The ant system: an autocatalytic optimizing process. Technical Report 91–016, Milano, Italy; 1991. p. 1–21.

[13] Dorigo M. Learning by probabilistic boolean networks. IEEE Trans Neural Netw 1994;2:887–91.

[14] Dorigo M, Maniezzo V, Colorni A. Ant system: optimization by a colony of cooperating agents. IEEE Trans Syst Man Cybernet 1996;B 26(1):29–41.

[15] Dorigo M, Caro GD, Gambardella LM. Ant algorithms for discrete optimization. Artif Life: MIT Press 1999;5:137–72.

[16] Dorigo M, Stützle T. Ant colony optimization. MIT Press; 2004, ISBN: 0-262-04219-3.

[17] Tsai C-F, Tsai C-W, Wu H-C, Yang T. ACODF. A novel data clustering approach for data mining in large databases. Journal of Systems and Software: Elsevier 2004;73:133–45.

[18] Foo SY, Stuart G, Harvey B, Meyer-Baese A. Neural network-based EKG pattern recognition. Eng Appl Artif Intell: Pergamon/Elsevier 2002;15:253–60.

[19] Haykin S. Neural networks: a comprehensive foundation. New York: Macmillan; 1994, ISBN: 0023527617.

[20] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error propagation. In: Rumelhart DE, McClelland JL, editors. Parallel distributed processing, vol. 1. Cambridge, MA: MIT Press; 1986. [chapter 8], p. 318–62.

[21] Hagan MT, Demuth HB, Beale MH. Neural Network Design. PWS Publishing; 1996, ISBN: 0-9717321-0-8.

[22] http://www.physionet.org/physiobank/database/vfdb/.

[23] Palreddy S, Hu YH, Mani V, Tompkins WJ. A multiple-classifier architecture for ECG beat classification. Neural Networks for Signal Processing VII. Proceedings of IEEE Workshop. Amelia Island, FL, USA: IEEE; 1997. p. 172-81. ISBN: 0-7803-4256-9.