Journal of ELECTRICAL ENGINEERING, VOL. 58, NO. 6, 2007, 334–341

AN EXPERT SYSTEM FOR DIGITAL SIGNAL TYPE CLASSIFICATION

Ataollah Ebrahimzadeh * — **Maryam Ebrahimzadeh** **

Because of rapid growing of radio communication technology of late years, importance of automatic classification of digital signal type is rising increasingly. Most of the proposed techniques can only classify low order of digital signals and/or a few kinds of digital signal. They usually require high levels of signal to noise ratio (SNR). This paper presents an expert technique that classifies a variety of digital signals. In this technique a multilayer perceptron neural network with self-adaptive step-size learning algorithm is proposed for determination the membership of the received signal. A combination of the higher order moments and the higher order cumulants up to eighth are proposed for extraction the prominent characteristics of the considered digital signals. In this technique, we have proposed a genetic algorithm for feature selection in order to reduce the number of features. Simulation results show that the proposed technique has high performance for identification the different digital signal types even at very low SNRs.

K e y w o r d s: statistical pattern recognition, signal type classification, SASS learning algorithm, a combination of the higher order moments and higher order cumulants, genetic algorithm

1 INTRODUCTION

Automatic signals type classification is an important subject for novel communication systems. Signal type classification is also believed to play an important part in future 4G software radios, [1]. The general idea behind the software radio architecture is to perform a considerable amount of the signal processing in software instead of it being defined in hardware. This would enable the radio to adapt to a changing environment and user requirements by simply updating the software or by using adaptable software systems. In such scenarios, a broadcaster could for example change to appropriate modulation schemes according to the capacity of the channel. A receiver incorporating automatic modulation recognition could then handle this in real times.

Automatic signal type classification techniques, usually, divided into two principle groups. One is the decision theoretic approach and the other is pattern recognition. Decision theoretic approaches use probabilistic and hypothesis testing arguments to formulate the recognition problem [2–3]. These methods suffer from their very high computational complexity, difficulty to implementation and lack of robustness to model mismatch [7]. Pattern recognition approaches, however, do not need such careful treatment. They are easy to implement. They can be further divided into two subsystems: the feature extraction subsystem and the classifier subsystem. The former extracts prominent characteristics from the raw data, which are called features, and the latter is a classifier [4–15].

In [4] the authors introduced a modulation classifier based on the zero-crossing characteristic of the intercepted signal. The considered signal types were: BPSK, QPSK, 8PSK, BFSK, 4FSK, 8FSK. The decision about the modulation type is based on the variance of the zero-crossing interval sequence, the frequency and phase difference histograms. In [5], it is proposed a technique that is based on the constellation shape. This technique used a Fuzzy-C means clustering method for classification of PSK4, PSK8 and QAM16. The accuracy rate of the identification exceeded 90 % for $SNR > 5$ dB. In [6], the authors proposed a technique for identification ASK2, ASK4, PSK2, PSK4, FSK2 and FSK4 signals. The classifier is based on a decision flow. These digital signal types have been identified with a success rate around 90 % at $SNR = 10$ dB. In [7], the authors proposed a technique based on elementary fourth-order cumulants. In [8], the authors proposed a classifier to discriminate among ASK, 4DPSK, 16QAM and FSK digital signals. The chosen features are: the kurtosis of the signal, the number of peaks in the phase probability density function (PDF) and the mean of the absolute value signal frequency. A fuzzy classifier was used in this technique. For $SNR > 5$ dB, the identifier worked properly. When SNR was less than 5 dB, the performance was worse. In [9], for the first time, Ghani and Lamontagne proposed using the multi-layer perceptron (MLP) neural network with back-propagation (BP) learning algorithm for automatic signal type identification. They showed that neural network classifier outperforms other classifiers such as K-Nearest Neighbor (KNN). In [10], power spectral density (PSD) measurements were used in conjunction with neural networks to identify the signal's type. This approach worked well for signals of interest whose power content distinctively varied with changes in frequency. It did not work as well with signal types like PSK. In [11], the authors introduced two classifiers: neural network classifier and fixed threshold classifier, for analog and digital mod-

* Faculty of Electrical and Computer Engineering, ** Faculty of Mathematics and Physics, University of Mazandaran; ataebrahim@yahoo.com (e_zadeh@nit.ac.ir), maryames437@yahoo.com

ISSN 1335-3632

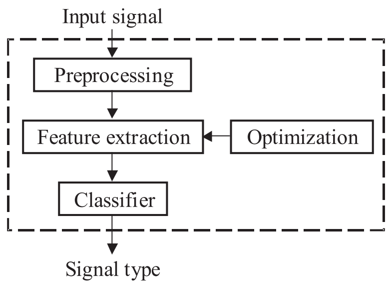

Fig. 1. General scheme of the proposed technique

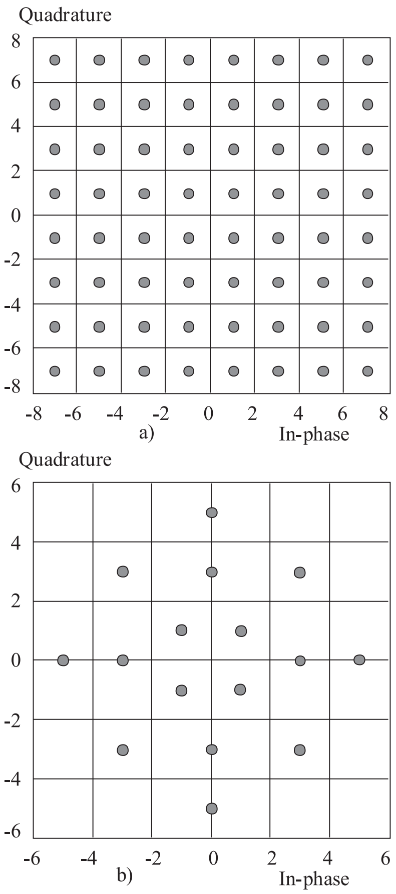

Fig. 2. Constellations of: a) QAM64, b) V29

ulation recognition. They showed that the neural network classifier has a higher performance than the threshold classifier. In [12], the authors used the mean and the next three moments of the instantaneous characteristics as the features of signal type classification. They used different classifiers and showed that the artificial neural network has better performance than K-Nearest Neighbor (KNN) classifier and the well known binary decision trees. They reported a success rate of 90 % with SNR ranges 15–25 dB. In [13], the authors proposed an identifier based on cyclic spectral features for identification of AM, USB, LSB, FM, ASK, FSK, BPSK, QPSK and SQPSK. It was claimed that cyclic spectrum posses more advantage than power spectrum in signal type recognition. The success rate of this identifier is reported around 90 % with SNR ranges 5–25 dB. In [14], the authors used a combination of the symmetry, fourth order cumulants and fourth order moments of the received signals as the features for identification of PSK2, PSK4, QAM8, ASK2 and ASK4. The classifier was a modified MLP neural network (with few output nodes). They reported a success rate about 92 % at SNR of 8 dB. In [15], the authors used the features that proposed in [6] and a MLP neural network as the classifier. This identifier showed a success rate about 93 % at $SNR = 8$ dB for identification of ASK2, ASK4, PSK2, PSK2, FSK2, FSK4, FSK4 and QAM16 digital signals.

From the published works, it can be found that: a) most of the proposed techniques can only recognize low orders of digital signals and/or a few kinds of digital signals, b) usually, the proposed techniques require high SNRs, c) the techniques, which use MLP neural networks as the classifier, have higher performances than others. In this paper, we present an expert technique that classifies a variety of digital signals with high accuracy even at very low SNRs.

Figure 1, shows the general scheme of the proposed expert technique. In this figure, the preprocessing module performs: the rejection of noise outside of the signal bandwidth, carrier frequency estimation (or to be known), recovery of the complex envelope, etc. This module is similar to signal type classification techniques and hence will not be explained here. Feature extraction module, extraction the prominent characteristics of the received signal, *ie* a combination of the higher order moments and higher order cumulants. Section 2, describes the feature extraction module. A multilayer perceptron neural network with self-adaptive step-size (SASS) learning algorithm is proposed for determination the membership of the received signal [16]. Section 3 presents this classifier. Optimization module is a genetic algorithm (GA) that is used for feature selection. Section 4 introduces the genetic algorithm. Some of the evaluation results of this system are shown in Section 5. Finally, Section 6 concludes the paper.

2 EFFICIENT FEATURES

Features define format of digital signals. As we know, different types of the digital signal have different properties; therefore finding the suitable features in order to identify them (especially in case of higher order and/or non-square) is a difficult task. In this paper we have considered the following digital signals: ASK2, ASK4, ASK8, PSK2, PSK4, PSK8, QAM8, QAM16, QAM32, QAM64, V29, and V32. Figure 2 shows the constellation of some of these signals. Based on our extensive researches, a combination of the higher order moments and higher order cumulants up to eighth provide a fine way for discrimination of the considered digital signal types. Following phrases, briefly describe these features.

Probability distribution moments are the generalization of concept of the expected value. We know that the general expression for the i^{th} moment of a random variable is given by [17]:

$$\mu_i = \int_{-\infty}^{\infty} (s-m)^i f(s)\mathrm{d}s \tag{1}$$

where m is the mean of the random variable. The definition for the i^{th} moment for a finite length discrete signal is given by:

$$\mu_i = \sum_{k=1}^{N} (s_k - m)^i f(s_k) \tag{2}$$

where N is the data length. In this study signals are assumed to be zero mean. Thus:

$$\mu_i = \sum_{k=1}^{N} s_k^i f(s_k)\,. \tag{3}$$

Now, the auto-moment of the random variable may be defined as follows:

$$M_{pq} = E\big[s^{p-q}(s^*)^q\big] \tag{4}$$

where p is the moment order and s^* stands for complex conjugation of s. Assume a zero-mean discrete based-band signal sequence of the form $s_k = a_k + jb_k$. Using the definition of the auto-moments, the expressions for different orders may be easily derived. For example:

$$M_{41} = E\big[(a+jb)^3(a-jb)\big] = E\big[a^4 - b^4 + 2ab(a^2+b^2)j\big]\,. \tag{5}$$

$$M_{63} = E\big[s^3(s^*)^3\big] = E\big[|s|^6\big] = E\big[(a^2+b^2)^3\big] = E\big[a^6 + 3a^4b^2 + 3a^2b^4 + b^6\big]. \tag{6}$$

For, definition of cumulants, consider a scalar zero mean random variable with characteristic function:

$$\hat{f}(t) = E\big\{e^{jts}\big\}\,. \tag{7}$$

Expanding the logarithm of the characteristic function as a Taylor series, one obtains:

$$\log \hat{f}(t) = k_1(jt) + \frac{k_2(jt)^2}{2!} + \dots + \frac{k_r(jt)^2}{r!} + \dots \tag{8}$$

The constants k_r in (8) are called the cumulants (of the distribution) of . The symbolism for p^{th} order of cumulant is similar to that of the p^{th} order moment. More specially:

$$C_{pq} = \mathrm{Cum}[\ \underbrace{s,\dots,s}_{(p-q)\text{ terms}}\ \underbrace{s^*,\dots,s^*}_{(q)\text{ terms}}\,. \tag{9}$$

For example:

$$C_{81} = \mathrm{Cum}\big(s,s,s,s,s,s,s,s^*\big) \tag{10}$$

Moments may be expressed in terms of cumulants as:

$$M[s_1,\dots,s_n] = \sum_v \mathrm{Cum}\big[\{s_j\}_{j\in v_1}\big]\dots\mathrm{Cum}\big[\{s_j\}_{j\in v_q}\big] \tag{11}$$

where the summation index is over all partitions $v = (v_1,\dots,v_q)$ for the set of indices $(1,2,\dots,n)$, and q is the number of elements in a given partition. The n^{th} order cumulant is a function of the moments of orders up to (and including) n:

$$\mathrm{Cum}[s_1,\dots,s_n] = \sum_v (-1)^{q-1}(q-1)!E\Big[\prod_{j\in v_1} s_j\Big]\dots E\Big[\prod_{j\in v_1} s_j\Big] \tag{12}$$

where the summation is being performed on all partitions $v = (v_1,\dots,v_q)$ for the set of indices $(1,2,\dots,n)$. Assume $n=3$. In such a case, the available set of indices is $(1,2,3)$, and four different types of partitioning may be obtained for that set: $\{(1,2,3)\}$ leading to $q=1$, $\{1,(2,3)\}$ leading to $q=2$, $\{2,(1,3)\}$ leading to $q=2$, $\{3,(1,2)\}$ leading to $q=2$, $\{(1),(2),(3)\}$ leading to $q=3$. Therefore:

$$\begin{aligned}\mathrm{Cum}[s_1,s_2,s_3] &= (-1)^{1-1}(1-1)!E[s_1s_2s_3] \\ &+ (-1)^{2-1}(2-1)!E[s_1]E[s_2s_3] \\ &+ (-1)^{2-1}(2-1)!E[s_2]E[s_1s_3] \\ &+ (-1)^{2-1}(2-1)!E[s_3]E[s_1s_2] \\ &+ (-1)^{3-1}(3-1)!E[s_1]E[s_2]E[s_3] \\ &= E[s_1s_2s_3] - E[s_1]E[s_2s_3] - E[s_2]E[s_1s_3] - E[s_3]E[s_1s_2] \\ &+ 2E[s_1]E[s_2]E[s_3]\,.\end{aligned} \tag{13}$$

In the same manner, cumulant expressions up to eighth order can be computed. For example:

$$C_{42} = M_{42} - |M_{20}|^2 - 2M_{21}^2\,, \tag{14}$$

$$C_{63} = M_{63} - 9M_{41}M_{21} - 6M_{21}^3\,, \tag{15}$$

$$C_{80} = M_{80} - 35M_{40}^2 - 630M_{21}^4 + 420M_{20}^2M_{40}^2\,, \tag{16}$$

We have considered the second, fourth, sixth and eighth order of the moments and cumulant as the features. These features are: M_{20}, M_{21}, M_{40}, M_{41}, M_{42}, M_{60}, M_{61}, M_{62}, M_{63}, M_{80}, M_{81}, M_{82}, M_{83}, M_{84}, C_{20}, C_{21}, C_{40}, C_{41}, C_{42}, C_{60}, C_{61}, C_{62}, C_{63}, C_{80}, C_{81}, C_{82}, C_{83} and C_{84}. Therefore the total number of the features is 28. We have computed all of these features for the considered digital signals. For simplifying the indication, the signals ASK2, ASK4, ASK8, PSK2, PSK4, PSK8, QAM8, QAM16, QAM32, QAM64, V29, and V32 are substituted with P_1, P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9, P_{10}, P_{11}, and P_{12} respectively. Table 1 shows some of these features for the considered digital signal types. These values are computed under the constraints of unit variance and noise free.

Table 1. Some of the features for a number of digital signal types.

	P_4	P_5	P_6	P_2	P_8	P_9	P_{11}
M_{41}	1	0	0	1.64	0	0	0
M_{61}	1	−1	0	2.92	−1.32	−.38	1.06
M_{63}	1	1	1	2.92	1.96	1.9	2.25
M_{82}	1	−1	0	5.25	−2.48	−.74	2.19
M_{84}	1	1	1	5.25	3.13	2.89	3.78
C_{42}	−2	−1	−1	−1.36	−0.68	−.69	−.58
C_{61}	16	4	0	8.32	2.08	0	−1.5
C_{63}	16	4	4	8.32	2.08	2.11	1.49
C_{80}	−244	−34	1	−30.1	−13.99	−1.9	−5.6
C_{82}	−244	−46	0	−30.1	−29.82	−8.4	22.3
C_{84}	−244	−18	−17	−30.1	17.38	16.6	27.5

3 NEURAL NETWORK CLASSIFIER

We have used a MLP neural network as the classifier. A MLP neural network consists of an input layer of source nodes, one or more hidden layers of computation nodes (neurons) and an output layer [18]. The number of nodes in the input and the output layers depend on the number of input and output variables, respectively. Figure 3 shows a typical MLP architecture consists of input layer, one hidden layer and output layer, respectively. Inputs are propagated through the network layer by layer and MLP gives a non-linear mapping of the inputs at the output layers. The input vector $x = (x_1, x_2, \dots, x_N)^\top$ is transformed to an intermediate vector of hidden variables $\boldsymbol{u}$, using the activation function φ_1. The output u_j of the j^{th} node in the hidden layer is obtained as follows:

$$u_j = \varphi\Big(\sum_{i=1}^{N} w_{i,j}^1 x_i + b_j^1\Big) \tag{17}$$

where b_j^1 and $w_{i,j}^1$ represent the bias and the weight of the connection between the j^{th} node(in the hidden layer) and the i^{th} (input) node, respectively. The superscript 1 represents the connection (first) between the input and the hidden layers. The output vector $y = (y_1, y_2, \dots, y_Q)^\top$ is obtained from the vector of intermediate variables $\boldsymbol{u}$ through a similar transformation using activation function φ_2 at the output layer. For example, the output of the neuron k can be expressed as follows:

$$u_k = \varphi\Big(\sum_{l=1}^{M} w_{l,k}^2 x_i + b_k^2\Big) \tag{18}$$

where the superscript 2 represents the connection (second) between the neurons of the hidden and the output layers. There are several forms of activation functions φ_1 and φ_2, such as logistic function, hyperbolic tangent and piece- wise linear function.

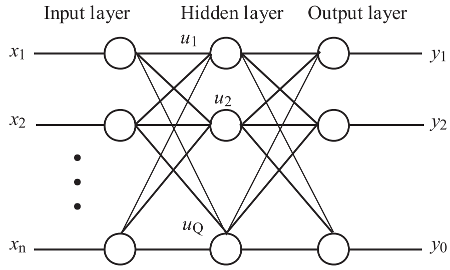

Fig. 3. Typical structure of MLP (with one hidden layer)

In this paper a single hidden layer MLP neural network was chosen as the classifier. The issue of learning algorithm is very important for MLP. Backpropagation (BP) learning algorithm is still one of the most popular algorithms. In BP a simple gradient descent algorithm updates the weight values:

$$w_{ij}(t+1) = w_{ij}(t) - \frac{\delta E}{\delta w_{ij}}(t)\,, \tag{19}$$

$$\frac{\delta E}{\delta w_{ij}} = \frac{\delta E}{\delta y_j}\frac{\delta y_j}{\delta u_i}\frac{\delta u_i}{\delta w_{ij}} \tag{20}$$

where w_{ij} represents the weight value from neuron j to neuron i, ε is the learning rate parameter, E represent the error function, y_j is the output and u_i is the weighted sum of the input of neuron i. However under certain conditions, the BP network classifier can produce non-robust results and easily converge to local minimum. Moreover it is time consuming in training phase. New algorithms have been proposed so far in order to network training. However, some algorithms require much computing power to achieve good training, especially when dealing with a large training set.

In this paper, SASS learning algorithm is used [16]. SASS is an adaptive step-size method. It is based on the bisection method for minimization in one dimension, in which the minimum of a valley is found by taking a step in the descent direction of half the previous step. The method has been adapted to allow the step-size to both increase and decrease. It uses the same update rule resilient but updates Δ_{ij} differently. It uses two previous signs and has an increment factor of 2.

$$\Delta_{ij}(t) = \begin{cases} 2.0 * \Delta_{ij}\,, \quad \text{if } \frac{\delta E}{\delta w_{ij}}(t-1)\frac{\delta E}{\delta w_{ij}}(t) \geq 0\,, \\ \qquad\qquad \text{and } \frac{\delta E}{\delta w_{ij}}(t-2)\frac{\delta E}{\delta w_{ij}}(t) \geq 0\,, \\ 0.5 * \Delta_{ii}(t-1)\,, \quad \text{otherwise.} \end{cases} \tag{21}$$

This update value adaptation process is then followed by the actual weight update process, which is governed by the following equation:

$$w_{ij}(t+1) = w_{ij}(t) + \Delta w_{ij}(t)\,. \tag{22}$$

Table 2. Correct matrix of SPTECH at $SNR = 3$ dB

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
P_1	96											
P_2		95										
P_3			96									
P_4				97								
P_5					95							
P_6						94						
P_7							93					
P_8								91				
P_9									92			
P_{10}										91		
P_{11}											90	
P_{12}												91

Table 3. Correct matrix of SPTECH at $SNR = 9$ dB

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
P_1	99											
P_2		100										
P_3			100									
P_4				100								
P_5					99							
P_6						100						
P_7							100					
P_8								98				
P_9									99			
P_{10}										98		
P_{11}											98	
P_{12}												98

4 GENETIC ALGORITHM AND OPTIMIZATION OF THE NUMBER OF FEATURES

From Section 3 it is found that the total number of the features was a lot. Although some of these features may carry good classification information when treated separately, there is a little gain if they are combined together (due to the sharing the same information content). For reduce the number of features, we can use a feature selection method. With doing so, we can achieve the least suitable possible number of features without compromising the performance of the signal type classifier. In this paper, we shall focus on a stochastic search method using GA.

GA has ability to efficiently search large spaces about which little is known, but also is easy to implement [19]. GA is a stochastic optimization algorithm, which adopts Darwin's theory of survival of the fittest. The use of GA needs consideration of these basic issues: chromosome (genome) representation, selection function, genetic operators like mutation and crossover for reproduction, creation of initial population, termination criteria, and the evaluation function.

How to encode a solution of the problem into a chromosome is a key issue for genetic algorithms. In this paper, real-coded genomes were used. The genome contains the row numbers of the selected features from the total set. In this genome encoding method, one has to specify the length of the genome string. In a GA, the selection of individuals to produce successive generations plays a vital role. There are several ways to select a new intermediate population. In this paper we apply the elitism. Elitism is the name of the method that first copies the best chromosome (or few best chromosomes) to the new population.

Genetic operators are the basic search mechanism of the GA for creating new solutions based on the existing population. The operators are two basic types: mutation and crossover. Crossover produces two new individuals (offspring) from two existing individuals (parents). Crossover occurs with a crossover probability of P_C. A point is chosen for two strings where their genetic information's are exchanged. In this paper, we use one-point crossover, and $P_C = 0.8$. Mutation is intended to prevent falling of all solutions in the population into a local optimum of the solved problem. Mutation operation randomly changes the offspring resulted from crossover. In this paper, mutation occurs with mutation probability of 0.05.

GA will rate its own performance around that of the evaluation (fitness) function. The fitness function used in the present work returns the number of correct identification of the test data. The better identification results give rise to higher fitness index. To start the solution process, the GA has to be provided with an initial population. In this paper the random generation of initial solutions for the population is used [20]. The solution process continues from one generation to another selecting and reproducing parents until a termination criterion is satisfied. Convergence of a GA can be defined in several ways. In our application, the maximum number of generation is used as the terminating criterion.

5 SIMULATION RESULTS

This section presents some simulation results of the proposed identifier. All signals are digitally simulated in MATLAB editor. We assumed that carrier frequencies were estimated correctly. Thus, we only considered complex base-band signals. Gaussian noise was added according to SNRs, -3, 0, 3, 6, 9 and 18 dB to the simulated signals. Each signal type has 2400 realizations. These are then divided into training, validation and testing data sets. The activation functions of tan-sigmoid and logistic were used in the hidden and the output layers, respectively. The MSE is taken equal to 10^{-6}. The MLP classifier is allowed to run up to 5000 training epochs. Based

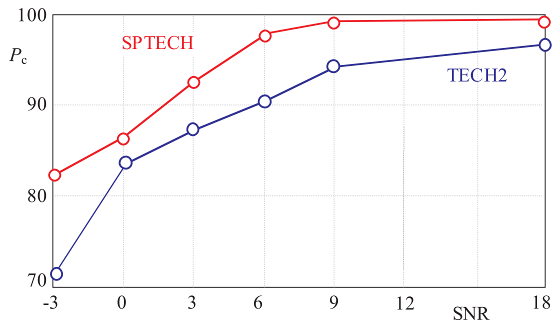

Fig. 4. Comparison between the performances of SPTECH and TECH2 at different SNRs

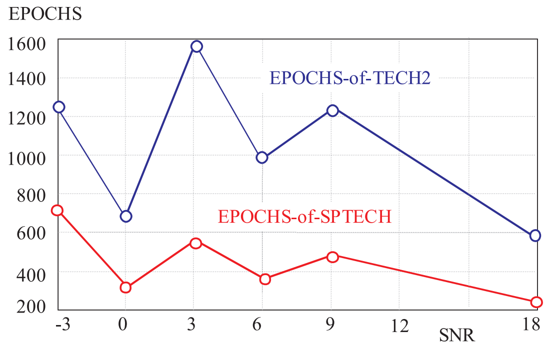

Fig. 5. Comparison between the number of epochs of SPTECH and TECH2 at different SNRs.

on our extensive experiments it seems that the number of 22 neurons is adequate for reasonable classification.

Table 4. The performances of SPTECH at different SNRs

SNR	Training	Testing
−3	83.34	82.24
0	87.86	86.50
3	93.76	93.42
6	97.78	97.66
9	99.38	99.26
18	99.62	99.44

5.1. Performance of the straight proposed technique

In this subsection, we evaluate the performance of straight proposed technique (SPTECH), *ie* full features (all of 28 features) are used. Tables 2, 3 show the correct matrix of SPTECH at two selected values of SNR. Table 4 shows the performances of the SPTECH at various SNRs. It can be seen that the performances of the classifier is generally very good even at low SNRs. This is due the two facts: the chosen classifier and proposed features.

As mentioned the speed of the learning algorithm is an important issue for a MLP neural network. The efficiency of the learning algorithm affects the amount of experimentation that can be done. To indicate the effectiveness of chosen learning algorithm (SASS), we have compared it with the SUPERSAB learning algorithm that is an adaptive learning rate algorithm [21]. In SUPERSAB learning algorithm each weight w_{ij}, connecting node j with node i, has its own learning rate that can vary in response to error surface. The system that uses MLP with SUPERSAB learning algorithm as the classifier is named as TECH2. Based our experiments, any number in the vicinity of 38 neurons seems to be adequate for reasonable classification. Other simulations setup is the same. Figure 4 shows the comparison between the performances of TECH2 and SPTECH at different SNRs. Figure 5 compares these systems in case of the numbers of epochs that are needed in each SNR. It can be seen that the performance of the SPTECH is better than the performance of TECH2. Also the number of epochs in case of SPTECH is less than in case of TECH2.

5.2 Performances of the proposed technique with applying the genetic algorithm

In the GA, a population size of ten individuals was used starting with randomly generated genomes. One has to specify the number of features (the length of the genome string) that varies from 1 to 28. We have experimented the identifier using several features selected using GA.

Table 5. Performance of PTECH with five features selected using GA.

SNR	TP with GA	TP without GA
−3	80.15	82.24
0	83.25	86.50
3	90.25	93.42
6	94.68	97.66
9	97.51	99.26
18	98.74	99.44

Table 6. Performance of PTECH with seven features selected using GA

SNR	TP with GA	TP without GA
−3	81.46	82.24
0	86.13	86.50
3	93.25	93.42
6	97.34	97.66
9	98.92	99.25
18	99.28	99.44

Table 5 shows the performance of the proposed technique (PTECH) using five features selected by GA. Table 6 shows the performance of PTECH using seven features selected by GA. Table 7 show the performance of PTECH using twenty features selected by GA. Also in these tables the performance of the straight identifier (SPTECH) is showed. It can be seen that in Table 6 the PTECH records a performance degradation less than 1 %

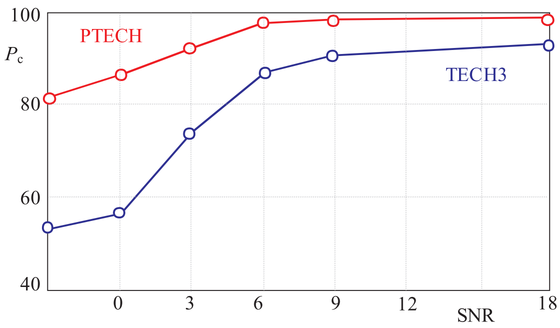

Fig. 6. Performances comparison between PTECH and TECH3.

Table 7. Performance of PTECH with twenty features selected using GA

SNR	TP with applying GA	TP without GA
−3	82.06	82.24
0	86.40	86.50
3	93.34	93.42
6	97.55	97.66
9	99.18	99.25
18	99.38	99.44

Table 8. Correct matrix of PTECH with only seven features selected using GA at $SNR = 3$ dB

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
P_1	96											
P_2		94										
P_3			96									
P_4				98								
P_5					95							
P_6						92						
P_7							93					
P_8								91				
P_9									92			
P_{10}										91		
P_{11}											90	
P_{12}												91

only at $SNR = -3$ dB. For other levels of SNR, the difference is negligible. Thus it can be said that the proposed technique makes high performance at most SNR values with only seven features that have been selected using GA. Tables 8, 9 show the correct matrices of PTECH at $SNR = 3$ dB and $SNR = 9$ with seven features that have been selected by GA. It is found that proposed method is able to identify the different types of digital signal with high accuracy only seven selected features using GA.

As mentioned the features play a vital role for identification of digital signals. In order to indicate the effectiveness of the chosen features, we have used the features that have been introduced in [7]. We name this classifier as THECH3. Other simulations setup is the same. Figure 6 show a performance comparison between TECH3 and our proposed technique (PTECH) at different SNRs. Results imply that our chosen features have effective properties in signal representation.

Table 9. Correct matrix of PTECH with only seven features selected using GA at $SNR = 9$ dB

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
P_1	99											
P_2		100										
P_3			100									
P_4				100								
P_5					99							
P_6						99						
P_7							100					
P_8								100				
P_9									97			
P_{10}										98		
P_{11}											98	
P_{12}												97

6 CONCLUSIONS

Automatic signal type identification is an important issue for both the civilian and military domain. Most of the proposed techniques can only recognize a few kinds of digital signal and/or lower orders of digital signals. They usually need high SNRs for classification of the considered digital signals. These problems are mainly due to the two facts: the features and the classifier that are used in the identifiers. In this paper we have proposed a number of the features, i.e. a combination of the higher order moments and the higher order cumulants (up to eighth), which have high ability in representing of the digital signal types. As the classifier, we have proposed a MLP neural network with SASS learning algorithm as the classifier. This classifier has high classification ability. By using the mentioned features and classifier, we have presented a highly efficient classifier. This classifier discriminates different digital signal types with high accuracy even at very low SNRs. But there are a lot of features are used for this classification. In order to reduction the complexity of the proposed identifier we have used an optimizer, *ie* genetic algorithm. Using this idea reduces the number of features without trading off the generalization ability and accuracy. The optimized identifier also has high performance for identification of the considered different kinds of digital signal at all SNRs. This high efficiency is achieved with only seven features, which have been selected using genetic algorithm. One the advantage of the genetic algorithm features selection is its simple implementation. For future works, we can use the GA in order to construct of the classifier as well as the features selection. Also we can consider another set of digital signal types and evaluate this technique for classification of them. We can select the proper features that introduced by others and combine them with the features that are proposed in this paper in order to have suitable features set for classification the different types of digital signal.

References

[1] NOLAN, K. E.—DOYLE, L.—MACKENZIE, P.—MAHONY, D. O.: Modulation Scheme Classification for 4G Software Radio Wireless Network, Proc. IASTED, 2002.

[2] WEI, W.—MENDEL, J. M.: Maximum-Likelihood Classification for Digital Amplitude-Phase Modulations, IEEE Trans. Commun. **48** (2000), . 189–193.

[3] PANAGOTIOU, P.—POLYDOROS, A.: Likelihood Ratio Tests for Modulation Classifications, Proc. MILCOM, 2000, pp. 670–674.

[4] HSUE, S. Z.—SOLIMAN, S. S.: Automatic Modulation Classification Using Zero-Crossing, IEE Proc. Radar, Sonar and Navigation **137** (1990), 459–464.

[5] MOBASSERI, B. G.: Digital Modulation Classification Using Constellation Shape, Signal Processing **80** (2000), 251-277.

[6] AZZOUZ, E. E.—NANDI, A. K.: Automatic Recognition of Digital Modulations, Signal Processing **47** (1995), 55–69.

[7] SWAMI, A.—SADLER, B. M.: Hierarchical Digital Modulation Classification Using Cumulants, IEEE Trans. Comm. **48** (2000), 416-429.

[8] LOPATKA, J.—MACREJ, P.: Automatic Modulation Classification Using Statistical Moments and a Fuzzy Classifier, Proc. ICSP, 2000, pp. 121–127.

[9] CHANI, N.—LAMONTAGNE, R.: Neural Networks Applied to the Classification of Spectral Features for Automatic Modulation Recognition, Proc. MILCOM, 1993, pp. 1494–1498.

[10] HAGEDORN, G.—JAMES, B.—MILLER, C.: Neural Network Recognition of Signal Modulation Types, Proc. ANNIEC, 1997, pp. 170–175.

[11] NANDI, A. K.—AZZOUZ, E. E.: Algorithms for Automatic Modulation Recognition of Communication Signals, IEEE Trans. Comm. **46** (1998), 431-436.

[12] LOUIS, C.—SEHIER, P.: Automatic Modulation Recognition with Neural Network, Proc. MILCOM, 1993, pp. 111-115.

[13] EBRAHIMZADEH, A.—SEYEDIN, S. A.: Signal Identification Using an Efficient Method, accepted for publication in IJEEE.

[14] ZHAO, Y.—REN, G.—WANG, X.—WU, Z.—GU, X.: Automatic Digital Modulation Recognition Using Artificial Neural-Networks, Proc. ICNNSP, 2003, pp. 257–260.

[15] HANNAN, J.—BISHOP, M.: A Class of Fast Artificial NN Learning Algorithms, Tech. Report, JMH-JMB 0/96, Dep. of Cybernetics, University of Reading, 1996.

[16] NIKIAS, C. L.—PETROPULU, A. P.: Higher-Order Spectra Analysis: A Nonlinear Signal Processing Framework, PTR Prentice-Hall, Englewood Cliffs, NJ, 1993.

[17] HAYKIN, S.: Neural Networks: A Comprehensive Foundation, MacMillan, New York, 1994.

[18] GOLDBERG, G. E.: Genetic Algorithms in Search, Optimization and Machine Learning, Addison-Wesley, NY, USA, 1989.

[19] MICHALEWICZ, Z.: Genetic Algorithms+Data Structures= Evolution Programs, 3rd Edition, Springer, NY, USA, 1999.

[20] TOLLENAERE, T.: Supersab: Fast Adaptive bp with Good Scaling Properties, Neural Networks **3** (990), 561–573.

Received 7 March 2006

Ataollah Ebrahimzadeh was born in Babol. He received his PhD degree in electrical engineering. Now he is a professor in the Faculty of Electrical and Computer Engineering at University of Mazandaran. His research interests are: General area of Signal Processing, Wireless Communications, Biomedical Engineering, Statistical Pattern Recognition, Artificial Intelligence, Digital Communications, Sensor Fusion, Adaptive Filtering, and Adaptive Control. He is the reviewer of a number of international journals and conferences. Also he was the chairman of the sessions in many conferences.

Maryam Ebrahimzadeh was born in Babolsar. She is with the faculty of mathematics and physics at University of Mazandaran. Her areas of researchs are: Mathematics in Electrical Engineering, Pattern Recognition, IT, Management in IT.