Performance Study of Cyclostationary based Digital Modulation Classification Schemes

Udit Satija, M. S. Manikandan and Barathram Ramkumar

School of Electrical Sciences

Indian Institute of Technology

Bhubaneswar, India

{us11, msm, barathram}@iitbbs.ac.in

Abstract—**Automatic Modulation Classification (AMC) is a essential component in Cognitive Radio (CR) for recognizing the modulation scheme. Many modulated signals manifest the property of cyclostationarity as a feature so it can be exploited for classification. In this paper, we study the performance of digital modulation classification technique based on the cyclostationary features and different classifiers such as Neural Network, Support Vector Machine, k-Nearest Neighbor, Naive Bayes, Linear Discriminant Analysis and Neuro-Fuzzy classifier. In this study we considered modulations i.e. BPSK, QPSK, FSK and MSK for classification. All classification methods studied using performance matrix including classification accuracy and computational complexity (time). The robustness of these methods are studied with SNR ranging from 0 to 20dB. Based upon the result we found that combining cyclostationary features with Naive Bayes and Linear Discriminant Analysis classifiers leads to provide better classification accuracy with less computational complexity.**

I. INTRODUCTION

Automatic Modulation Classification (AMC) aims to detect the unknown modulation format employed in the received signal with a high probability of success and in a short period of time [1]. AMC has taken a significant place in software defined radio and cognitive radio. AMC plays important role in dynamic spectrum management and interference identification. It has both military and civilian application. AMC also helps interoperability between various wireless standard as it is an intermediate step.

AMC algorithms can be grouped into two different categories: Maximum likelihood (ML) and Feature Based (FB) methods [2]. ML based technique exploits the probability density function (PDF) of the observed waveform and offers optimal solution but suffers from computational complexity. On the other side, FB methods employ some features for classification at reduced complexity and nearly optimal. If these features are designed in a proper way then its performance is comparable to that of ML methods. Different types of features have been used in AMC e.g., wavelet transform, higher order cumulants and moments, cyclo-stationarity, instantaneous amplitude, phase, and frequency in [2]. Disadvantage of ML based method is that one should have either channel information or one must be able to estimate the channel based on their statistics. Feature based AMC is a two step process. First, signal features are extracted which is then fed to a classifier for classification.

In cyclostationary based AMC, as the name suggests cyclostationary features of the received signal are employed as in [3], [4] and [5]. For the cyclostationary signal, both

mean and autocorrelation are periodic, so spectral correlation function (SCF) can be calculated by taking fourier transform of cyclic autocorrelation. SCF exhibits peaks whose location and magnitude are unique for each modulation scheme. The peaks of SCF are captured by cyclic domain profile (CDP). Using this CDP, modulation schemes can be classified.

Aa mentioned earlier, classifier is a integral part of feature based AMC. Neural Network (NN), Support Vector Machine (SVM), k-Nearest Neighbor (k-NN), Naive Bayes (NB), Linear Discriminant Analysis (LDA) and Neuro-Fuzzy (NF) are the some of the widely used classifiers. The design of classifiers is a very important in terms of computational complexity and the performance of classification in the low SNR regime. The main objective of the paper is to find a suitable classifier that works well with cyclostationary features extracted from multiple modulation schemes. This paper is organized as follows: Section I describes the introduction about the AMC. In Section II, we have provided the cyclostationarity features in detail. Section III discusses different classifiers. Section IV presents all the simulation results and we finally concluded the paper in section V.

II. CYCLOSTATIONARITY ANALYSIS

If the mean and the autocorrelation of the random process s(t) is periodic i.e. μ (t+T)= μ (t) and R_s(t+T,w+T)=R_s (t,w) for all t and w, then the process is said to be cyclostationary process.

The conventional autocorrelation is given by

$$R_s(\tau) = E[s(t + \frac{\tau}{2})s(t - \frac{\tau}{2})] \tag{1}$$

For cyclostationary signal autocorrelation is periodic in T, then we can write as,

$$R_s(t + T, \tau) = R_s(t, \tau) \tag{2}$$

Therefore one can expand it using Fourier series

$$R_s(t + \frac{\tau}{2}, t - \frac{\tau}{2}) = \sum_{\alpha_0} R_s^{\alpha_0} e^{-j2\pi\alpha_0 t} \tag{3}$$

where, α_0 is k/T, for all k= 1,2,3....

$$R_s^{\alpha_0}(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_{-T/2}^{T/2} R_s(t, \tau)e^{i2\pi\alpha_0 t}dt. \tag{4}$$

The weiner theorem can be extended to cyclostationary process. Then, Spectral Correlation Function (SCF) is the Fourier

transform of this cyclic autocorrelation function which is,

$$S_s^{\alpha_0} = F.T.[\hat{R}_s^{\alpha_0}] \tag{5}$$

where, *F.T.* is the fourier transform.
There are few number of samples available in practice, hence SCF is estimated using these samples. Let us define the cyclic periodogram for the time invariant Fourier transform $S_T(t,f)$

$$S_{sT}^{\alpha_0}(t,f) = \frac{1}{T}S_T(t,f+\frac{\alpha_0}{2})S_T^*(t,f-\frac{\alpha_0}{2}) \tag{6}$$

where,

$$S_T(t,f) = \int_{t-T/2}^{t+T/2} s(v)e^{-j2\pi fv}dv \tag{7}$$

So, the estimate of the SCF can be found by frequency smoothing of $S_{sT}^{\alpha_0}(t,f)$

$$S_{sT}^{\alpha_0}(t,f)_{\Delta f} = \frac{1}{\Delta f}\int_{f-\Delta f/2}^{f+\Delta f/2} S_{sT}^{\alpha_0}(t,u)du \tag{8}$$

Then, SCF can be obtained by increasing the observation length T and reducing the Δf.

$$S_s^{\alpha_0}(f) = \lim_{\Delta f \to 0}\lim_{T \to \inf} S_{sT}^{\alpha_0}(t,f)_{\alpha_0} \tag{9}$$

So, Spectral Coherence(SC) function is actually the covariance of the two spectral components present at f=$\pm\alpha_0/2$. So, S_u(f)=S_s(f+$\alpha_0/2$) and S_v(f)=S_s(f-$\alpha_0/2$). So,the SC function is

$$C_s^{\alpha_0}(f) = \frac{S_s^{\alpha_0}(f)}{[S_u(f)S_v(f)]^{1/2}} \tag{10}$$

The magnitude of spectral coherence function lies between 0 and 1. To reduce the computational complexity Cyclic domain profile (CDP) can be used which is defined as

$$I(\alpha_0) = max_f|C_s^{\alpha_0}(f)| \tag{11}$$

As mentioned earlier, CDP is unique for each modulation scheme. In Figure 2, we can see that for different modulation schemes i.e. BPSK, QPSK, FSK and MSK, CDP is unique. Figure 2 shows CDP features for different noise scenarios. Left most column and middle column of Figure 2 denotes CDP profile in without noise case and 10 SNR case respectively. Right most column of Figure 2 shows the CDP profile at 5 dB as it is evident from the figure that the spectral features in the low SNR case is getting obscure.

III. CLASSIFIERS

Classifiers are the most important part of the feature based AMC. AMC is a two step process as shown by flow chart in Figure 1. In the first step, classifiers are trained with cyclostationary features extracted from different modulation schemes. Then, in the next step classification is performed for the new test signal by extracting its cyclostationary features. So for the classification, different types of classifiers used in this paper are explained below:
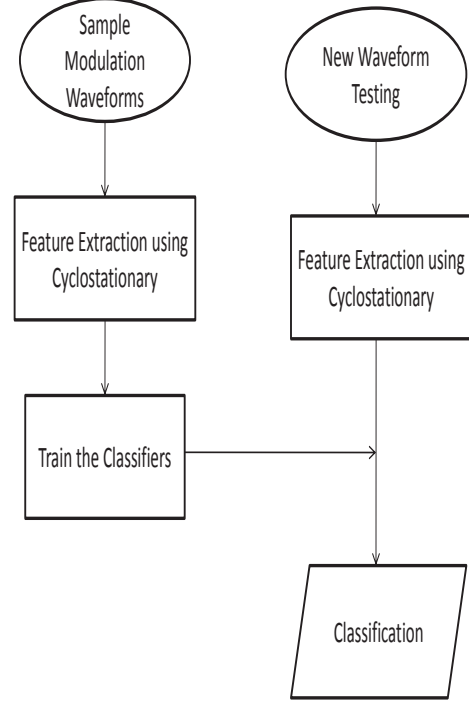

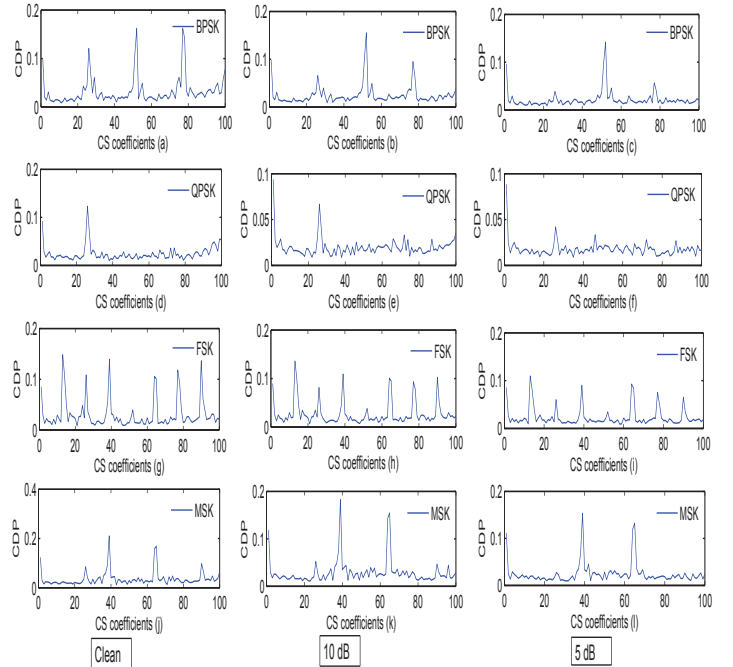Fig. 1. Flow Chart for Classification

Fig. 2. Cyclostationary Features

A. Artificial Neural Networks

Artificial neural network (ANN) is the mostly used classifier in AMC. ANNs are motivated by central nervous system and these are widely used in machine learning and pattern recognition. Due to its capability of adaptation and learn to work with complicated signals, it is very useful when applied to classification problems. ANNs have flexible structures that make it easy to implement [2], [6]. ANNs can be classified into supervised or unsupervised networks. In the supervised ANNs, the network is trained using one part of the data set and then tested using other part. On the other side, the unsupervised ANNs cluster the input data and train themselves. The first technique gives more accurate results but it requires a large number of data compared to the unsupervised techniques. Most of the ANNs that have been tested in AMC field use the supervised learning techniques including Back propagation, Multi-Layer Perceptrons (MLP) and Radial Basis Function (RBF). The Self Organizing Map (SOM) neural network has been used in designing AMC is an unsupervised ANNs. MLP requires less memory so it is more attractive. Single MLP ANNs are mostly used but some have used cascaded MLP ANNs [2]. RBFNNs are superior in training rates as compare to MLP ANNs but having equivalent classification performance.

B. Support Vector Machines

Although the ANNs are widely used with AMC, but they have some limitations in training that they can end up with over fitting and/or local minimum. Whereas, SVMs not only overcome these limitations but also provide more generality at lower SNR [2]. The main principle of SVM is to classify data by finding the best hyperplane that separates the data points of one class from other class. The best hyperplane is that which chooses maximum margin between the two classes. The data points closer to the separating hyperplane are called support vectors. Let M is the margin between two hyperplane. So,

$$max.M \qquad s.t. \quad y.x_i^T.w \geq M$$

where, y is the class label and $x_i^T.w$ is the observation in the decision surface.
Let $\|w\| = \frac{1}{M}$, then the optimization problem becomes,

$$min.\frac{1}{2}\|w\| \quad or \quad \frac{1}{2}w^Tw$$
$$s.t. \quad y.x_i^T.w \geq 1$$

This is called primal problem to maximize the margin between the separating hyperplane. Using Lagrangian it can be converted into dual problem which is

$$L(w,\lambda) = arg.\max_{\lambda}\frac{1}{2}w^{*T}w^* + \sum_i \lambda_i \qquad \forall \quad \lambda_i \geq 0$$

where, w^* is the weighted sum of observations defined as,

$$w^* = \sum_{i=1}^{N}\lambda_i y_i x_i$$

A kernel function is used to map the non linear features from input domain to feature domain. Some of the well known kernel functions are RBF, MLP and polynomial function. Binary SVM (BSVM) is used to classify two class problems.

In more than two classes problem the first class is classified among all others classes. Afterwards, it uses another SVM in order to classify the second class against the rest of the classes and so on. Multi-class SVM (MSVM), which uses higher dimension feature spaces, is used to classify multi-class problems.

C. k-Nearest Neighbor

k- Nearest Neighbor (k-NN) is non parametric method used for classification and regression. In k-NN classification output is a class membership. By a majority vote of its neighbors an object is classified. If the object is most common with its nearest neighbors then that object is assigned to that class [7]. The k-NN search techniques have been used in different applications such as bioinformatics, image processing and data compression, document retrieval, computer vision, multimedia database, and marketing data analysis. k-NN can be used for classification, local weighted regression, missing data computation and interpolation, density estimation, distance-based learning functions, such as k-means clustering. In k-means clustering random centers are chosen initially then based on the Euclidean distance the centers are updated. k-NN is a instance based learning where the function is approximated locally and it is sensitive to the local structure of the data. The k-NN algorithm is the simplest algorithm among all other machine learning algorithms. The training phase comprised of only storing the feature vectors and class labels of samples. Different distance matrices are used. Euclidean distance is the commonly used distance metric. Another metric such as overlap metric (or Hamming distance) is used for discrete variables. If the distance metric is learned with specialized algorithms such as Large Margin Nearest Neighbor or Neighborhood components analysis then classification accuracy of k-NN algorithm can be improved significantly.

D. Linear Discriminant Analysis

Linear discriminant analysis (LDA) methods are commonly used in statistics, pattern recognition and machine learning to find a linear combination of features which characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier or for dimensionality reduction. LDA is used to express one dependent variable as linear combination of other variables or features. LDA has continuous independent variables and a categorical dependent variable (i.e. the class label). LDA explicitly attempts to create the difference between the classes of data. These classifiers are attractive because they have closed-form solutions that can be easily computed, are inherently multiclass, and have proven to work well in practice [6], [8]. Consider a set of observations x (i.e. features attributes, measurements or variables) from training set. The classification problem is to find a good predictor for the new sample based on the given observation set. LDA assumes that both conditional probability density functions p(x/y=0) and p(x/y=1) are normally distributed with mean and covariance (μ_0, $\sigma_{y=0}$) and (μ_1, $\sigma_{y=1}$) respectively. Where y is the class label of the particular class. So, Bayes optimal solution under this assumption is to predict from second class by comparing

TABLE I. CONFUSION MATRIX FOR DIFFERENT CLASSIFIER USING CYCLOSTATIONARY FEATURES

Classifier	Modulation	SNR=0dB				SNR=5dB				SNR=10dB				SNR=15dB				SNR=20dB			
Neural Network	BPSK	0	100	0	0	26	74	0	0	59	41	0	0	95	5	0	0	100	0	0	0
	QPSK	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0
	FSK	0	72	28	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0
	MSK	1	95	0	4	0	2	1	97	0	0	0	100	0	0	0	100	0	0	0	100
Naive Bayes	BPSK	87	13	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0
	QPSK	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0
	FSK	0	76	24	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0
	MSK	0	100	0	0	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100
Linear Discriminant Analysis	BPSK	29	71	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0
	QPSK	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0
	FSK	0	98	2	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0
	MSK	0	95	0	5	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100
k-Nearest Neighbor	BPSK	0	100	0	0	50	50	0	0	100	0	0	0	100	0	0	0	100	0	0	0
	QPSK	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0
	FSK	0	100	0	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0
	MSK	0	100	0	0	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100
Support Vector Machine (Default Kernel)	BPSK	0	98	0	2	0	47	0	53	9	4	0	87	86	0	0	14	100	0	0	0
	QPSK	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0
	FSK	0	100	0	0	0	43	57	0	0	0	100	0	0	0	100	0	0	0	100	0
	MSK	0	93	0	7	0	16	0	84	0	3	0	97	0	0	0	100	0	0	0	100
Neuro-Fuzzy	BPSK	72	5	2	21	90	1	3	6	99	0	1	0	99	1	0	0	100	0	0	0
	QPSK	3	11	11	75	2	14	13	71	12	70	8	10	13	82	0	5	1	97	0	2
	FSK	0	0	98	2	0	0	99	1	0	0	100	0	0	0	100	0	0	0	100	0
	MSK	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100	0	0	0	100

log likelihood ratio with some threshold γ.

$$(x - \mu_0)^T \sigma_{y=0}^{-1}(x - \mu_0) + ln|\sigma_{y=0}| - (x - \mu_1)^T \sigma_{y=1}^{-1}(x - \mu_1)$$
$$-ln|\sigma_{y=0}| < \gamma$$

E. Naive Bayes Classifier

A Naive Bayes classifier is a simple probabilistic classifier based on strong (naive) independence assumptions by using Bayes' theorem [9], [10]. The Naive Bayes classifier is commonly used when features are independent of one another within each class, but it appears to work well in practice even when that independence assumption is not valid. First in training phase it estimates the parameters of a probability distribution using training samples, assuming features are conditionally independent given the class and second in prediction phase, the method computes the posterior probability of the new test sample belonging to each class. Then according to largest posterior probability features X given class Y, P(X|Y) it classifies the test sample. Assumption of class independence helps the Naive Bayes classifier to better estimate the parameters required for accurate classification with less training data than many other classifiers. It also provides support for normal(Gaussian), kernel, multinomial, and multivariate multinomial distributions for different features. All model parameters (i.e., class priors and feature probability distributions) can be approximated with relative frequencies (or by maximum likelihood probability) from the training set. A class' prior may be calculated by assuming equiprobable classes or by calculating an estimate for the class probability (i.e., (prior for a given class) = (number of samples in the class) / (total number of samples)) from the training set. Parameters for a feature's distribution may be estimated by assuming a distribution or generate nonparametric models for the features from the training set. Another common technique to obtain a new set of Bernoulli-distributed features by discretizing the

TABLE II. COMPUTATIONAL COMPLEXITY IN CYCLOSTATIONARY FEATURES

Classifier	Computational Time(in sec.)
Naive Bayes	1.70918
Support Vector Machine	5.666738
Linear Discriminant Analysis	1.698175
K-Nearest Neighbor	1.649138
Neural Network	68.359(training)+1.787(testing)
Neuro Fuzzy	1.75431

feature values that uses maximum a posteriori or MAP decision rule which chooses the maximum probable hypothesis which as follows:

$$classify(f_1, f_2, ...f_n) = argmax_c \prod_{i=1}^{N} p(C = c)$$
$$p(F_i = f_i | C = c)$$

F. Neuro-Fuzzy Classifier

This classifier is a hybridization of Neural network and Fuzzy. This hybridization provides intelligence to the neural network in the learning and connection structure [11]. Fuzzy logic provides tuning of training parameters in neural network. Fuzzy logic helps in unsupervised learning using its membership functions through clustering algorithms. It uses multilayer feed-forward network for fuzzification, fuzzy inference and defuzzification.

IV. SIMULATION RESULTS

Performance of four different modulation schemes i.e. BPSK, QPSK, FSK and MSK based on cyclostationary features have been tested using six different classifiers. 100 different modulation signals were generated for the particular

modulation scheme for the testing (as can be seen in confusion matrix). So, we can see that at the low SNR all classifiers' performance decreases as shown in Table I by confusion matrix. Diagonal entries in the confusion matrix denotes classification accuracy and off diagonal entries represents misclassification. At 5dB Naive Bayes and LDA classifiers are performing well whereas other classifiers' performance is diminishing at high noise because at low SNR spectral features are lost. Table II shows the comparison of different classifiers in terms of computational time. SVM and Neural Network classifiers are giving worst performance both with classification as well in computational complexity whereas other classifiers like Naive Bayes, k-NN, Neuro Fuzzy and LDA are giving good performance in case of computational complexity. Though the k-NN and Neuro Fuzzy classifier are not giving good classification at lower SNR as compare to Naive Bayes and LDA but one can also use at high SNR case because their computational complexity is less.

V. CONCLUSION

In this paper, we studied the performance of digital modulation classification for BPSK, QPSK, FSK and MSK based on the cyclostationary features and different classifiers such as neural network, support vector machine, k-nearest neighbor, naive bayes, linear discriminant analysis and neuro-fuzzy classifiers. All classification methods were studied using performance matrix including classification accuracy and computational complexity in the range of SNR ranging from 0 to 20dB. It is evident from the result that combining cyclostationary features with Naive Bayes and LDA leads to provide better classification accuracy with less computational complexity.

REFERENCES

[1] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: Classical approaches and new trends," *IET Commun.*, Vol. 1, No. 2, pp. 137-156, Apr. 2007.

[2] A. Hazza, M. Shoaib, S. A. Alshebeili, and A. Fahad, "An overview of feature-based methods for digital modulation classification," in *proc. 2013 1st Int. Conf. Commun. Signal Process. Applicat. IEEE ICCSPA*, pp. 1-6.

[3] B. Ramkumar, "Automatic modulation classification for cognitive radios using cyclic feature detection," *IEEE Circuits and Systems Mag.*, Vol. 9 No. 2, pp. 27-45, 2009.

[4] S. Sobolewski, W. L. Adams, and R. Sankar, "Automatic Modulation Recognition techniques based on cyclostationary and multifractal features for distinguishing LFM, PWM and PPM waveforms used in radar systems as example of artificial intelligence implementation in test," *IEEE AUTOTESTCON*, 2012.

[5] G. Liu, Z. Zhang, and Y. Yang, "Recognition of Cyclostationary Signals Smoothed," *In Nat. Conf. on Inform. Technol. and Comput. Sci. Atlantis Press*, 2012.

[6] M. D. Wong, and A. K. Nandi, "Automatic digital modulation recognition using artificial neural network and genetic algorithm," *Signal Process.*, Vol. 84 No. 2, pp. 351-365, 2004.

[7] M. W. Aslam, Z. Zhu, and A. K. Nandi, "Automatic modulation classification using combination of genetic programming and KNN," *IEEE Trans. Wireless Commun.*, Vol. 11, No. 8 pp. 2742-2750, Aug. 2012.

[8] A. M. Martinez,and A. C. Kak, "Pca versus lda," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 23 No. 2, pp. 228-233, 2001.

[9] K. P. Murphy, "Naive Bayes classifiers," *University of British Columbia*,2006.

[10] G. H. John, and P. Langley, "Estimating continuous distributions in Bayesian classifiers," *Proc. of the Eleventh conference on Uncertainty in artificial intelligence*, pp. 338-345, August,1995.

[11] S. Abe, "Pattern classification: neuro-fuzzy methods and their comparison," *Springer Publishing Company, Incorporated*, 2012.