HIDDEN MARKOV MODELS FOR DIGITAL MODULATION CLASSIFICATION IN UNKNOWN ISI CHANNELS

Anchalee Puengnim, Thierry Robert†, Nathalie Thomas* and Josep Vidal⋆*

* IRIT/ENSEEIHT/TéSA, 2 rue Charles Camichel, BP 7122, 31071 Toulouse cedex 7, France
† CNES, 18 avenue E. Belin, BPI 1219, 31401 Toulouse cedex 9, France
⋆ Technical University of Catalonia, Campus Nord-UPC, Jordi Girona 1-3, 08034 Barcelona, Spain
{anchalee.puengnim,nathalie.thomas}@tesa.prd.fr, thierry.robert@cnes.fr, josep.vidal@upc.edu

ABSTRACT

This paper addresses the problem of classifying digital linear modulations transmitted through an unknown finite memory channel. Hidden Markov Models (HMMs) are used to model the received communication signals. In a classification purpose, our main interest is to determine the posterior probabilities of these received signals conditionally to each class. This paper proposes to use the Baum-Welch algorithm to compute these probabilities which are then plugged into the optimal Bayes decision rule. The performance of the proposed classifier is assessed through several simulation results.

1. INTRODUCTION

Digital modulation classification consists of identifying the type of a modulated signal corrupted by noise and other impairments. It is required in many communication applications including cooperative and non-cooperative scenarios [1]. In a non-cooperative scenario, the classification of digitally modulated signals propagating through an intersymbol interference (ISI) environment has been studied by many researchers [2, 3, 4]. However, it still presents a great deal of issues. Indeed, without some kind of ISI mitigation, the performance of current classification techniques designed for additive white Gaussian noise (AWGN) channels degrades significantly. The Bayes-based classifiers [5, 6] suffer from high computational complexity in the presence of an unknown channel because averaging over the data symbols leads to an exponential computational cost. Moreover, Bayes-based classifiers assume the knowledge of noise variance when calculating the conditional class densities. A practical suboptimal Bayes classifier based on a “plug-in” rule was derived in [4]. The main idea of the proposed classifier was to estimate the channel coefficients as well as the other unknown model parameters by using Markov Chain Monte Carlo (MCMC) methods. The estimated parameters were then plugged into the posterior probabilities. The classical maximum *a posteriori* (MAP) classification rule was finally implemented with these estimated probabilities. Unfortunately, the complexity of the plug-in MCMC classifier could be prohibitive for some practical applications.

This paper studies a new digital modulation classifier based on HMMs. The classifier estimates HMM posterior probabilities as well as model parameters by using the forward/backward Baum-Welch (BW) algorithm [7]. The estimated posterior probabilities are then used for classification via the usual MAP rule. Here, we are interested in classifying linear modulation types transmitted through an unknown finite memory channel and corrupted by AWGN. As a result, our main goal is to determine the posterior probabilities that the received communication signal corresponds to modulation types belonging to a known dictionary. However, channel coefficient estimates can also be obtained as side results.

The proposed classifier will be compared with the Per-Survivor Processing (PSP) technique introduced in [2]. This technique estimates the data sequence and the unknown parameters of a communication signal, and classifies this communication signal by using the generalized likelihood ratio test (GLRT). It tackles the problem by using the PSP to estimate the channel coefficients and the data sequence in order to calculate the test statistic. Note that the classification thresholds of this method have to be determined empirically. Also, PSP requires good initialization. Another practical approach which might be used for comparison is based on constant modulus and alphabet-matched algorithms followed by cumulant-based classifiers [3]. However, this technique relies on the performance of blind equalizers which usually operate at high SNRs. Furthermore, the decision after the cumulant-based classifier requires one to measure the erratic behavior of the cumulant estimates, which could be dubious and complicated.

This paper is organized as follows: Section 2 presents the signal model used for modulation classification. The received signal is then be modeled as a probabilistic function of an hidden state represented by a first order HMM. Section 3 recalls the main steps of the BW algorithm which determines the probability of the observation sequence given the model and estimates the unknown model parameters. Section 4 studies the new “plug-in” modulation classifier rule based on the BW algorithm. Simulation results and conclusions are reported in Section 5 and 6, respectively.

2. SIGNAL MODEL AND PROBLEM FORMULATION

Our assumptions regarding the operating communication system and the signal model are similar to [1]. After preprocessing, the baseband complex envelope of the received signal sampled at one sample per symbol at the output of a matched filter can be written

$$\begin{aligned} x(n) &= \textstyle\sum_{l=0}^{q} h_l d(n-l) + z(n), \quad n = 1,2,...,L, \\ &= hs^T(n) + z(n) \end{aligned} \tag{1}$$

where

- L is the number of symbols in the observation interval,
- $d(n) \in \{d_1, d_2, ..., d_M\}$ is an independent and identically distributed (i.i.d.) symbol sequence of M-values drawn from one of C constellations denoted $\{\lambda_1, \lambda_2, ..., \lambda_C\}$,

- $s(n) = [d(n), d(n-1), ..., d(n-q)]^T$,
- $\mathbf{h} = [h_0, h_1, ..., h_q]^T$ is a vector containing the $q+1$ taps of the linear finite impulse response (FIR) channel,
- q is the channel memory,
- $z(1), ..., z(L)$ is an i.i.d. complex Gaussian noise sequence which has zero-mean and variance σ^2 ($z(n)$ and $d(n)$ are also supposed to be independent).

The received signal $x(n)$ can be modeled as a probabilistic function of an hidden state at time n which is represented by a first order HMM with the following characteristics:
1) The state of the HMM at the nth time instant is $\mathbf{s}(n)$. Thus, $s(n)$ takes its values in $\{s_1, s_2, ..., s_N\}$ of size $N = M^{q+1}$, where s_j is the jth possible value of $s(n)$.
2) The state transition probability distribution is

$$a_{ij} = P[s(n+1) = s_j | s(n) = s_i], \tag{2}$$

which equals $1/M$ when all symbols are equally likely.
3) The initial state distribution vector $\pi = (\pi_1, ..., \pi_N)^T]$ is defined by $\pi_i = P[s(1) = s_i] = 1/N$.

The probability density function (pdf) of the observation $x(n)$ conditioned on state j, denoted as $p_j(x(n)) \triangleq p(x(n)|s_j)$ can be written

$$p_j(x(n)) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{|x(n) - m_j|^2}{2\sigma^2}\right), \tag{3}$$

for $j = 1, ..., N$, where $m_j = \sum_{l=0}^{q} h_l d_j(n-l)$. We denote as $m = [m_1, ..., m_N]^T$ the vector containing all means. Our objective is to recognize the type of linear modulated constellation transmitted though a finite memory channel corrupted by AWGN without any knowledge regarding m_j and the noise variance σ^2. Given a sequence of observation $x = [x(1), ..., x(L)]^T$, the proposed strategy calculates the posterior probability of the observation sequence given each possible model λ denoted as $P(x|m, \sigma^2, \lambda)$. The observation vector is then assigned to the most likely class, according to the MAP rule:

$$\widehat{\lambda}_{\text{MAP}} = \arg\max_{\lambda} P(x|m, \sigma^2, \lambda), \tag{4}$$

where $\widehat{\lambda}_{\text{MAP}}$ is the estimated constellation and $\lambda \in \{\lambda_1, ..., \lambda_C\}$. However, the calculation of the posterior probability $P(x|m, \sigma^2, \lambda)$ given m_j and σ^2 requires high complexity in the order of $2LN^L$ computations. Moreover, the two parameters m_j and σ^2 are not available in general. This paper proposes to compute the posterior probability $P(x|m, \sigma^2, \lambda)$ by using the BW algorithm involving a complexity in the order of $N^2 L$ operations.

3. THE BW ALGORITHM

The BW algorithm is based on a forward-backward procedure which estimates iteratively the unknown model parameters maximizing the posterior probability of the unknown parameters. After convergence, the BW algorithm provides MAP estimates of m and σ^2 such that:

$$(\widehat{m}, \widehat{\sigma}^2) = \arg\max_{m, \sigma^2} P(m, \sigma^2 | x, \lambda). \tag{5}$$

The algorithm needs a forward operation to compute $P(m, \sigma^2|x, \lambda)$ whereas a forward/backward algorithm is necessary to estimate the unknown parameters m_j and σ^2. This section briefly recalls the principles of the standard BW algorithm. An LMS-type update BW algorithm is also discussed.

3.1. The Standard BW Algorithm

The standard BW algorithm [7] estimates $P(x|m, \sigma, \lambda)$ by using the following three step procedure iteratively:
1) Compute the normalized forward variable $\alpha_i(n)$

- Initialization:

$$\alpha_i(1) = \pi_i p_i(x(1)), \quad 1 \le i \le N \tag{6}$$

- Induction:

$$\alpha_j(n+1) = c(n) p_j(x(n+1)) \sum_{i=1}^{N} \alpha_i(n) a_{ij}, \tag{7}$$

for $n = 1, 2, ..., L-1$, $j = 1, ..., N$, and where $c(n) = \left(\sum_{i=1}^{N} \alpha_i(n)\right)^{-1}$,

2) Compute the normalized backward variable $\beta_i(n)$

- Initialization:

$$\beta_i(L) = c(L), \quad 1 \le i \le N \tag{8}$$

- Induction:

$$\beta_i(n) = c(n) \sum_{j=1}^{N} a_i p_j(x(n+1)) \beta_j(n+1), \tag{9}$$

for $n = L-1, ..., 1$ and $i = 1, ..., N$.

3) Estimate the model parameters as follows

$$\widehat{m}_i = \frac{\sum_{n=1}^{L} \gamma_i(n) x(n)}{\sum_{n=1}^{L} \gamma_i(n)}, \tag{10}$$

$$\widehat{\sigma}^2 = \frac{1}{L} \sum_{n=1}^{L} \sum_{i=1}^{N} \gamma_i(n) |m_i - x(n)|^2, \tag{11}$$

where $\gamma_i(n) = \alpha_i(n) \beta_i(n)$.

In a batch mode implementation, steps 1 to 3 are carried out iteratively with updated values of $p_j(x(n))$ until convergence. Thus, the desired posterior probability given the model is computed as follows

$$\widehat{P}(x|m, \sigma, \lambda) = \frac{\sum_{i=1}^{N} \alpha_i(L)}{\sum_{i=1}^{L} c(i)}. \tag{12}$$

Different modifications have been applied to the standard BW algorithm to improve estimation/classification performance or reduce computation complexity. These modifications are presented in Section 3.3.

3.2. Regularization

For a linear channel, we have the relationship $\mathbf{m} = \mathbf{S}\mathbf{h}^T$, where $\mathbf{S}$ is the state matrix defined as $\mathbf{S} = [s_1, s_2, ..., s_N]$. This information has been used to regularize the estimated mean values. For this, at each iteration, the estimated means are projected into the space spanned by the columns of $\mathbf{S}$ [8]:

$$\mathbf{m} \leftarrow \mathbf{S}\mathbf{S}^{\sharp}\mathbf{m} \tag{13}$$

where $S^{\sharp}$ is the pseudo-inverse of $\mathbf{S}$ and $\hat{\mathbf{h}} = S^{\sharp}\mathbf{M}$.

3.3. The LMS-type Update Algorithm

The standard Baum-Welch algorithm suffers from the "curse of dimensionality" because the computation complexity and memory requirement are proportional to the square of the number of the states. Furthermore the convergence rate is rather slow. Thus, it is worth seeking improvements in terms of memory and computation speed. In this paper, we have implemented the LMS-update type algorithm initially presented in [8]:

$$m_i(n) = m_i(n-1) + \mu_m \gamma_i(n) e_i(n), \tag{14}$$

$$\sigma^2(n) = (1-\mu_s)\sigma^2(n-1) + \mu_s \left(\sum_{i=1}^{N} \gamma_i(n) |e_i(n)|^2 \right), \tag{15}$$

where $e_i(n) = x(n) - m_i(n-1)$ for $i = 1, ..., N$.

The initialization and time-induction calculation for the forward variable can be computed as in the standard BW algorithm. The calculation of backward variable can be obtained by using the fixed-lag or sawtooth-lag schemes [9]. In this work, we have implemented on the fixed-lag case where $\Delta > q+1$ and apply for each n the normalized backward recursion from $n+\Delta$ to n. However, for the normalized backward recursion from $n+\Delta$ to n, the calculation of the normalized forward variable from n to $n+\Delta$ is required for the fixed-lag Δ. This means that the calculation can be started as soon as the observation symbols are greater than $2+\Delta$.

4. CLASSIFICATION RULE

This section studies the following classification rule

$$\text{Assign } x \text{ to } \lambda_i \text{ if } \widehat{P}(x|\lambda_i) \geq \widehat{P}(x|\lambda_j), \forall j = 1, ..., C, \tag{16}$$

where $\widehat{P}(x|\lambda_i) \triangleq \widehat{P}(x|m, \sigma, \lambda_i)$ is obtained from (12). Note that the whole sequence (of length L) is required to estimate $\widehat{P}(x|\lambda_i$ even if online LMS-update type algorithm has been used for the computation of $m_i(n)$ and $\sigma^2(n)$. Note also that the observation length L required to properly identify the modulation constellations should be greater than the maximum number of states $N_{\max} = M_{\lambda_c}^{q+1}$ (i.e. $L > N_{\max}$) so that every possible state can be reached by the algorithm.

The performance of the proposed classifier will be evaluated by using the average probability of correct classification P_{cc} defined by

$$P_{cc} = \frac{1}{C} \sum_{i=1}^{C} P\left[\text{assigning } x \text{ to } \lambda_i | x \in \lambda_i\right],$$

where C is the number of possible modulations.

5. SIMULATION RESULTS

Many simulations have been carried out to evaluate the performance of the proposed classifier. All constellations have been normalized (unit energy). The signal-to-noise ratio (SNR) in decibels is defined as

$$\text{SNR} = 10 \log_{10} \left(\frac{|h|^2}{\sigma^2} \right).$$

Since the iterative BW algorithm may converge to a local maximum of the likelihood function, one important issue is parameter initialization. The impulse response of the unknown channel can be estimated by using higher-order statistics (HOS) of the received signal. According to [10], the impulse response of a qth-order moving average (MA) system can be calculated from the estimated fourth-order cumulants of its output as

$$\widehat{h}(k) = \frac{\widehat{c}_{4,x}(q,0,k)}{\widehat{c}_{4,x}(q,0,0)}, \quad k = 0, ..., q, \tag{17}$$

where $\widehat{c}_{4,x}(t_1, t_2, t_3)$ is an estimate of

$$c_{4,x}(t_1,t_2,t_3) = \text{cum}(x^*(t), x(t+t_1), x(t+t_2), x^*(t+t_3))$$

with

$$\text{cum}(w,x,y,z) = E(wxyz) - E(wx)E(yz) - E(wy)E(xz) - E(wz)E(xy).$$

This procedure generally yields good estimations at reasonably high operating SNRs.

5.1. Parameter estimation

This section studies the convergence and tracking characteristics of the LMS-type update algorithms. A 4QAM signal is transmitted through a linear channel whose complex impulse response is $h = [1, 0.75 + 0.25j]^T$. The output of the filtered sequence is then contaminated by an additive complex white Gaussian noise with variance $\sigma^2 = 0.01$. The initial values and the step-sizes of the LMS-type update algorithm have been adjusted as follows:

$$\mu_m = 0.6, \mu_s = 0.1, \sigma^2_{\text{init}} = 1, \Delta = 5.$$

Figures 1 and 2 display typical estimates for the real and imaginary parts of h_1 and the variance σ^2 for a single run. Figure 3 shows the average MSE versus SNR for the estimated real and imaginary parts of h_1. Of course, better performance can be achieved for high SNRs, as expected.

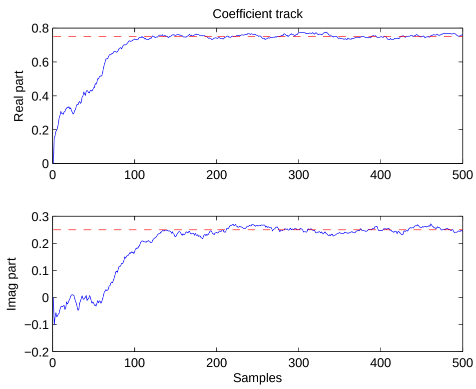

Fig. 1. Estimated real and imaginary part of h_1.

5.2. Classification performance

This section studies the performance of the plug-in MAP classifier defined in (16). All simulations have been obtained from 1000 trials belonging to each class λ_i (i.e. a total of 4000 signals for the four-class problem, and 2000 signals for the two-class problem). For our experiments, the mean vector m was initialized randomly or by (17) whereas the initial noise variance was set to $\sigma^2_{\text{init}} = 1$. The step-size for the LMS algorithm was set to $\mu_s = 0.1$ and $\Delta = 5$ for the fixed-lag scheme.

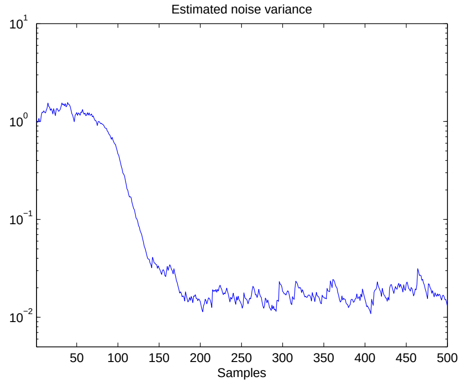

Fig. 2. Estimated noise variance.

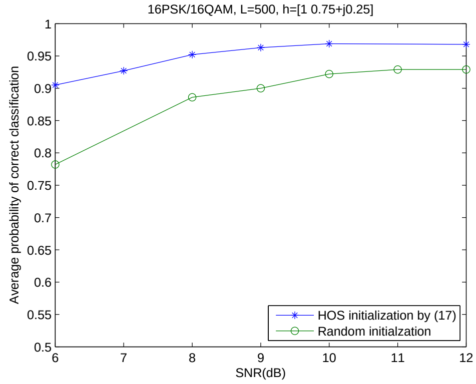

Fig. 4. Average probability of correct classification versus SNR.

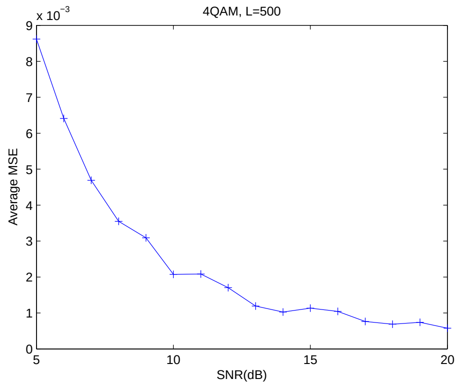

Fig. 3. Average MSE versus SNR.

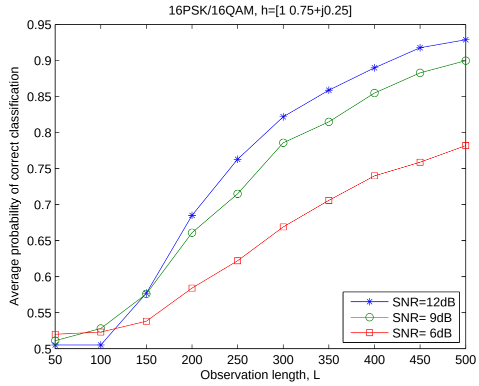

Fig. 5. Average probability of correct classification versus observation length.

5.2.1. 2-class problem

Consider a set of two modulation formats $\lambda = \{16\text{PSK}, 16\text{QAM}\}$. This particular example is interesting because the two modulation formats 16PSK and 16QAM have the same number of states and are difficult to distinguish in the presence of ISI and noise. Figure 4 shows the average probability of correct classification versus SNR for this problem. Note that two different initializations of the channel coefficients have been considered, namely HOS initialization using (17) and random initialization. Of course, the performance improves when the HOS initialization is used. Figure 5 displays the average probability of correct classification versus the number of observations for different SNRs. This allows one to adjust the number of observations required to achieve a given classification performance. For instance, at SNR $=$ 9dB, the observation length should satisfy $L \geq 500$ to ensure $P_{\text{cc}} \geq 0.9$. When operating at lower SNRs, larger values of L are necessary to ensure $P_{\text{cc}} \geq 0.9$.

For comparison, we consider a two-tap FIR channel with impulse response $\mathbf{h} = [0.707, 0.707]^T$ studied in [2]. The frequency response characteristics of this channel is compared to that of $\mathbf{h} = [1, 0.75 + 0.25j]^T$ in Fig. 6. This figure shows that this new channel exhibits a severe ISI due to its strong attenuation in the modulation passband. Figure 7 compares the performance of the MAP classifier (16) and the PSP/GLRT classifier as a function of the observation length. The proposed classifier provides better performance for $300 < L < 900$. Note that the two classifiers achieve the same performance for SNR $=$ 9dB and $L > 900$.

5.2.2. 4-class problem

This section considers a set of four modulations which have been studied in [4],[3], i.e., $\lambda = \{\text{BPSK}, 4\text{QAM}, 8\text{PSK}, 16\text{QAM}\}$. It is required to adjust the value of the LMS step-size parameter μ_m for each constellation. The values of μ_m used in this paper have been obtained by minimizing the average MSE of the estimated parameters. The following results have been obtained: $\mu_m = 0.3$ for BPSK, $\mu_m = 0.6$ for 4QAM, $\mu_m = 10$ for 8PSK, and $\mu_m = 20$ for 16QAM. The average probabilities of correct classification obtained with the classifier (16) for random and HOS initializations are displayed in Fig. 8. This results shows the necessity of having a good channel initialization. The probabilities of correct classification of each candidate modulation type are plotted in Fig. 9. This figure indicates that 4QAM and 16QAM are more difficult to classify than BPSK and 8PSK for the same SNR.

6. CONCLUSIONS

This paper addressed the problem of classifying digital modulations in the presence of a finite memory unknown channel. The received communication signal was classified according to a plug-in MAP rule. This rule required to estimate the posterior distri-

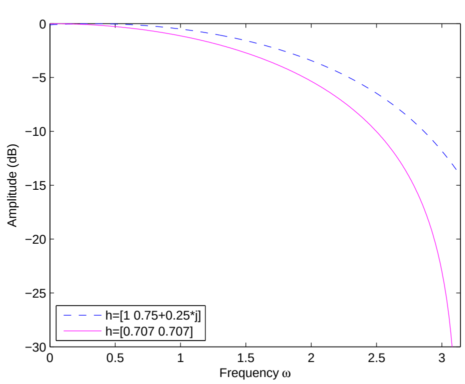

Fig. 6. Amplitude spectra for two channels with ISI.

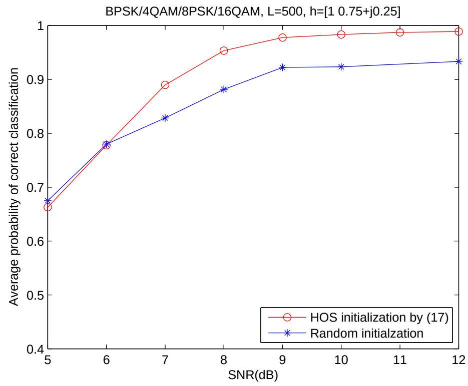

Fig. 8. Average probability of correct classification versus SNR.

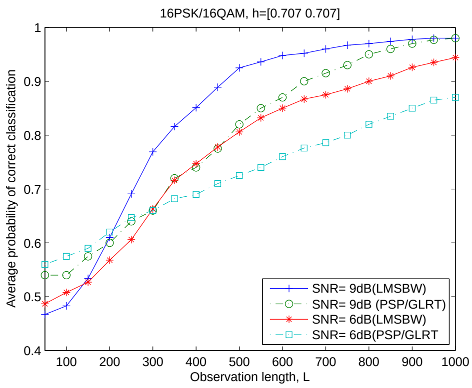

Fig. 7. Average probability of correct classification versus observation length.

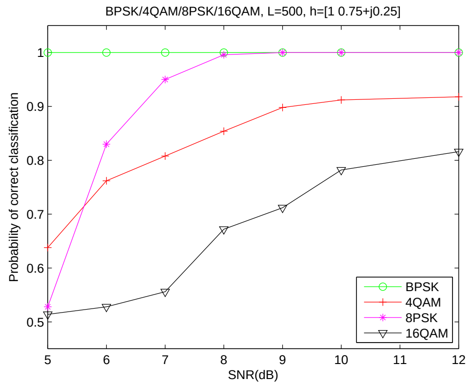

Fig. 9. Probability of correct classification of each candidate modulation format versus SNR.

bution of the received communication signal conditionally to each modulation belonging to a known dictionary. This estimation was conducted by using the Baum Welch algorithm for Hidden Markov Models which has shown interesting properties for speech recognition. The performance of the proposed classifier was assessed by means of several simulation results. It is important to note that the proposed classifier in unsensitive to phase offsets. The impact of frequency offset on the classification performance should be investigated.

7. REFERENCES

[1] A. Swami and B. Sadler, "Hierarchical digital modulation classification using cumulants," *IEEE Trans. Commun.*, vol. 48, no. 3, pp. 416–429, 2000.

[2] N. Lay and A. Polydoros, "Modulation classification of signals in unknown ISI environments," *Proc. IEEE MILCOM*, vol. 1, pp. 170–174, Nov. 1995.

[3] S. Barbarossa, A. Swami, B. Sadler, and G. Spadafora, "Classification of digital constellations under unknown multipath propagation conditions," *Proc. SPIE*, vol. 79, pp. 278–305, April 2000.

[4] S. Lesage, J. Tourneret, and P. M. Djuric, "Classification of digital modulation by MCMC sampling," *Proc. IEEE ICASSP*, pp. 2553–2555, 2001.

[5] C. Huang and A. Polydoros, "Likelihood method for MPSK modulation classification," *IEEE Trans. Commun.*, vol. 43, pp. 1493–1504, 1995.

[6] C. Long, K. Chugg, and A. Polydoros, "Further results in likelihood classification of QAM signals," *Proc. IEEE MILCOM*, vol. 1, pp. 57–61, Oct. 1994.

[7] L. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," *Proc. of the IEEE*, vol. 77, pp. 257 – 286, Feb. 1989.

[8] J. A. R. Fonollosa and J. Vidal, "Application of Hidden Markov Models to blind channel characterisation and data detection," in *Proc. IEEE ICASSP-94*, vol. 4, pp. 185–188, April 1994.

[9] V. Krishnamurthy and J. Moore, "On-line estimation of Hidden Markov Model parameters based on the Kullback-Leibler information measure," *IEEE Trans. Signal Processing*, vol. 41, pp. 2557 – 2573, Aug. 1993.

[10] J. M. Mendel, "Tutorial on higher-order statistics (spectra) in signal processing and system theory: Theoretical results and some applications," *Proc. of the IEEE*, vol. 79, pp. 278–305, March 1991.