ADAPTIVE CLUSTERING ALGORITHM FOR COOPERATIVE SPECTRUM SENSING IN MOBILE ENVIRONMENTS

Jesus Perez and Ignacio Santamaria

Advanced Signal Processing Group, University of Cantabria, Spain, https://gtas.unican.es/

ABSTRACT

In this work we propose a new adaptive algorithm for cooperative spectrum sensing in dynamic environments where the channels are time varying. We assume a centralized spectrum sensing procedure based on the soft fusion of the signal energy levels measured at the sensors. The detection problem is posed as a composite hypothesis testing problem. The unknown parameters are estimated by means of an adaptive clustering algorithm that operates over the most recent energy estimates reported by the sensors to the fusion center. The algorithm does not require all sensors to report their energy estimates, which makes it suited to be used with any sensor selection strategy (active sensing). Simulation results show the feasibility and efficiency of the method in realistic slow-fading environments.

Index Terms— Cooperative spectrum sensing, energy detection, clustering, likelihood ratio test, fading channels

1. INTRODUCTION

Spectrum sensing is a key operation in cognitive radio. Through spectrum sensing the cognitive radios (CRs) try to detect frequency bands that are not being used by the primary network. The performance of spectrum sensing is limited by shadowing and multi-path fading effects in the sensing channels between the primary users (PUs) and the CRs. By using cooperative spectrum sensing (CSS) the impact of those effects can be mitigated efficiently by the inherent multiuser/spatial diversity of the CR network [1], [2].

This work focuses on centralized CSS based on the soft fusion of the signal energy levels measured by the CRs [3], [4], [5], [6]. Each CR estimates the energy level at its location in the frequency band of interest and reports it to a fusion center (FC) through a control channel. Then, the FC makes a decision on the presence or absence of primary signals in the channel.

The likelihood radio test (LRT) is the optimal test when the FC knows the signal-to-noise ratio (SNR) at the CRs. In [3] and [4] the authors show that, when there is a single PU, the LRT leads to a linear fusion rule where the test statistic is a linear combination of the energy levels with appropriate weights.

Unfortunately, knowing the SNR at the CRs is very difficult, or even impossible, in practical scenarios. When these values are unknown the detection problem becomes a composite hypothesis test [7]. Generalized-LRT based algorithms have been proposed in [8] and [9] to solve this problem, whereas, in [10] and [11], the authors derive detection algorithms based on the Rao test and the locally most powerful test. All these works assume a single PU.

In this work, we propose a different approach to the CSS problem. First, we pose it as a composite hypothesis testing problem where the unknown parameters are the SNR at the CRs. Then, we derive a new adaptive clustering algorithm that allows the FC to dynamically estimate the SNR's from the most recent energy estimates reported to the FC by the CRs.

A similar approach to the CSS problem has been proposed in [5] and [6], where the authors use different Machine Learning techniques. However, these techniques are batch and, therefore, difficult to apply in mobile scenarios.

It is well known that the adequate selection of a subset of active CRs is a key issue in CSS because it permits to alleviate the control channels overhead without degrading the detection performance (see [1] and the references therein). This means that, at each time, the FC only fuses the energy levels from a subset of active CRs. Unlike the algorithms in [5] and [6], our method is able to deal with incomplete energy vectors (i.e. missing data), which makes it suited to be used with any CRs selection method.

2. SYSTEM MODEL

2.1. Primary network state

We consider a general model for the primary network activity where more than one PU can transmit simultaneously [6]. Let $s_i \in \{0, 1\}$ indicate the state of PU_i, where $s_i = 1$ means that it is transmitting (active) and $s_i = 0$ when it is inactive. The PU network state is given by the binary vector $\mathbf{s} = [s_1 \ s_2 \ \cdots s_{N_U}]^T$, where N_U is the number of PUs. Therefore, the channel is idle when $\mathbf{s} = \mathbf{0}$, whereas it is occupied when $\mathbf{s} \neq \mathbf{0}$.

2.2. Energy detection

Let W denote the channel bandwidth. The CRs perform energy detection [12] for a time duration of τ, so they take $M = W\tau$ signal samples during τ. The normalized received energy estimate at CR_j is

$$e_j = \frac{2}{\eta_j} \sum_{m=1}^{M} |x_j[m]|^2,$$

where $x_j[m]$ denotes the m-th signal sample and η_j is the noise variance at the CR_j.

2.3. Distribution of the energy estimates

The distribution of the energy estimates depends on the PU network state. Let $e_j|\mathbf{s}$ be the received energy estimate at CR_j conditioned to the primary network state $\mathbf{s}$. According to the central limit theorem, if M is large enough (e.g. $M \geq 20$ in practice), $e_j|\mathbf{s}$ is approximately normal distributed [6] with mean and variance given by

$$\mu_{\mathbf{s},j} = 2M(1 + \sum_{i=1}^{N_U} s_i \gamma_{i,j}), \quad \sigma^2_{\mathbf{s},j} = 4(\mu_{\mathbf{s},j} - M), \quad (1)$$

where $\gamma_{i,j}$ denotes the SNR at CR_j when PU_i is the only active user. Note that the variance is completely determined by the mean.

Let us assume that, at a given time, the CRs report their energy estimates to the FC. Then, the FC generates the energy vector $\mathbf{e} = [e_1 \ e_2 \ \cdots \ e_{N_S}]^T$, where N_S denotes the number of CRs. Since the local energy estimates are Gaussian distributed, the energy vector follows a multivariate Gaussian distribution

$$\mathbf{e}|\mathbf{s} \ \sim \ \mathcal{N}(\boldsymbol{\mu_s}, \boldsymbol{\Sigma_s}), \tag{2}$$

with mean $\boldsymbol{\mu_s} = [\mu_{s,1} \ \mu_{s,2} \ \cdots \ \mu_{s,N_S}]^T$. Assuming that the energy estimates from different CRs are independent, the covariance matrix is $\boldsymbol{\Sigma_s} = 4\text{diag}(\boldsymbol{\mu_s} - \mathbf{1}M)$. Note that, besides being a diagonal matrix, the covariance matrix is fully determined by the mean. This is a key fact that motivates the algorithm proposed in next sections.

When $\mathbf{s} = \mathbf{0}$ (that is under the null hypothesis), the Gaussian is spherical with centroid $\mu_0 = 2M\mathbf{1}$ and covariance matrix $\boldsymbol{\Sigma_0} = 4M\mathbf{I}$. Therefore, the distribution of $\mathbf{e}|\mathbf{0}$ only depends on M, which is assumed to be known by the FC.

To summarize, each PU network state $\mathbf{s}$ is associated with a multidimensional Gaussian distribution in the energy parameter space, which is fully determined by its mean $\boldsymbol{\mu_s}$.

3. LRT DETECTOR

The CSS detection problem can be posed as the following binary hypothesis test

$$\begin{aligned} H_0 \ &: \ \mathbf{s} = \mathbf{0} \\ H_1 \ &: \ \mathbf{s} \neq \mathbf{0} \end{aligned}$$

According to (2), the distribution of the energy vectors under each hypothesis are

$$\begin{aligned} p(\mathbf{e}|H_0) \ &= \ f(\mathbf{e}|\boldsymbol{\mu_0}), \\ p(\mathbf{e}|H_1) \ &= \ \sum_{\mathbf{s} \neq \mathbf{0}} \frac{\pi_{\mathbf{s}}}{1 - \pi_{\mathbf{0}}} f(\mathbf{e}|\boldsymbol{\mu_s}), \end{aligned} \tag{3}$$

where $f(\cdot|\boldsymbol{\mu_s})$ denotes the Gaussian probability density function with mean $\boldsymbol{\mu_s}$ and covariance matrix $\boldsymbol{\Sigma_s} = 4\text{diag}(\boldsymbol{\mu_s} - \mathbf{1}M)$. The distribution of $\mathbf{e}|H_1$ is a mixture of $2^{N_u} - 1$ Gaussian densities, being $\pi_{\mathbf{s}}$ the prior probability that the PU network is in state $\mathbf{s}$. On the other hand, $\mathbf{e}|H_0$ is a Gaussian random vector with known parameters that only depends on M.

Given the observed energy vector $\mathbf{e}$, the LRT is

$$\frac{p(\mathbf{e}|H_1)}{p(\mathbf{e}|H_0)} \underset{H_0}{\overset{H_1}{\gtrless}} \lambda, \tag{4}$$

where the likelihoods of $\mathbf{e}$ under both hypotheses are given by (3), and λ is a suitable detection threshold [7].

4. ADAPTIVE PARAMETER ESTIMATION

The application of the LRT requires the FC to know the current values of $\{\boldsymbol{\mu_s}\}$ and $\{\pi_{\mathbf{s}}\}$. The estimation of these parameters can be formulated as an adaptive clustering problem in the energy space, where the feature vectors are the energy vectors that arrive sequentially to the FC, and each cluster is associated with a primary network state. The means $\{\boldsymbol{\mu_s}\}$ are the clusters centroid, and the probabilities $\{\pi_{\mathbf{s}}\}$ are the cluster weights. The PU network state can be different for different energy vectors.

We assume that, at each time, only a subset of CRs report their energy estimates to the FC. This makes the clustering algorithm to be able to deal with missing data.

4.1. Adaptive clustering

In mobile environments the sensing channels are time varying and therefore the cluster centroids are moving in the energy space. In addition, consecutive energy vectors can belong to different clusters that correspond to different PU network states. Then, the FC applies a sliding window containing the N most recent energy vectors. Each time a new energy vector arrives to the FC, it is stored, and the oldest one is removed from the window. Using the new window of energy vectors, the clustering algorithm readjusts the estimates of the cluster centroids and weights. This way, the FC may compute the LRT (4), at any time, with the current estimates of $\{\boldsymbol{\mu_s}\}$ and $\{\pi_{\mathbf{s}}\}$. This procedure is depicted in Figure 1 where $\mathbf{e}(n)$ denotes the n-th energy vector in the sliding window.

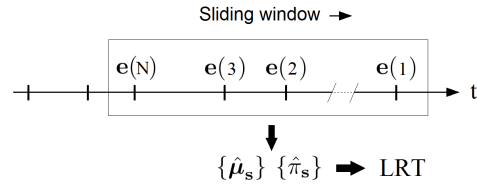**Fig. 1**. Sliding window over the energy vectors.

The window size N is a key parameter. The larger N, the more accurate clustering results. On the other hand, the channels (and therefore the centroids) must remain practically constant during the total acquisition time of the energy vectors.

4.2. Clustering algorithm

Our clustering problem has two of peculiarities that make it different from conventional clustering problems:

1. The FC fuses the energy estimates from a subset of the CRs, so the energy vectors can be incomplete.

2. The centroid of the null hypothesis cluster, $\mathbf{s} = \mathbf{0}$, is known in advance: $\mu_0 = 2M\mathbf{1}$.

These peculiarities require an ad-hoc clustering algorithm, which is presented in the following. It can be viewed as a modified version of the K-means algorithm to deal with the particularities of the problem at hand.

We consider the following objective function

$$J(\mathbf{r}, \boldsymbol{\mu}) = \sum_{n=1}^{N} \sum_{\mathbf{s}} r_{\mathbf{s}}(n) b_{\mathbf{s}}(n), \tag{5}$$

where $r_{\mathbf{s}}(n)$ denotes the indicator function about the PU network state at time n, so $r_{\mathbf{s}}(n) = 1$ if the PU network is assumed to be in state $\mathbf{s}$, and $r_{\mathbf{s}}(n) = 0$ otherwise. The term $b_{\mathbf{s}}(n)$ in (5) is a dissimilarity measure between the energy vector $\mathbf{e}(n)$ and the centroid of cluster $\mathbf{s}$. It is given by

$$b_{\mathbf{s}}(n) = \sum_{j=1}^{N_s} v_j(n)(e_j(n) - \mu_{\mathbf{s},j})^2,$$

where $v_j(n)$ is an indicator variable that equals 1(0) if the energy estimate $e_j(n)$ is available(unavailable) at the FC. Therefore, $b_{\mathbf{s}}(n)$ is the sum of the squared differences (in the energy space) between the available energy estimates and the corresponding components of $\boldsymbol{\mu_s}$. Note that if all energy estimates were available $(v_j(n) = 1, \forall j)$, $b_{\mathbf{s}}(n)$ would be the squared Euclidean distance between $\mathbf{e}(n)$ and $\boldsymbol{\mu_s}$, as in the conventional K-means algorithm.

The goal of the clustering algorithm is to find the values for the assignments $\mathbf{r} = \{r_{\mathbf{s}}(n)\}$ and the centroids $\boldsymbol{\mu} = \{\mu_{\mathbf{s},j}\}$ so as to minimize the objective function,

$$\hat{\mathbf{r}}, \hat{\boldsymbol{\mu}} = \underset{\mathbf{r},\boldsymbol{\mu}}{\operatorname{argmin}} \ J(\mathbf{r}, \boldsymbol{\mu}). \qquad (6)$$

From $\hat{\mathbf{r}}$, the computation of the clusters weights (the probabilities of the PU network states) is straightforward

$$\hat{\pi}_{\mathbf{s}} = \frac{1}{N} \sum_{n=1}^{N} \hat{r}_{\mathbf{s}}(n).$$

As in the conventional K-means algorithm, we consider an iterative procedure to minimize (6). Each iteration involves two successive steps:

1) Minimization of J with respect to $\mathbf{r}$ assuming $\boldsymbol{\mu}$ is fixed. This is done by assigning the observations to the cluster with the closest centroid.

2) Minimization of J with respect to the centroids $\boldsymbol{\mu}$ with $\mathbf{r}$ held fixed, which is done by updating the centroids for the current assignments $\mathbf{r}$. J is a quadratic function of the centroids' coordinates. It can be minimized by setting its partial derivatives to zero

$$\frac{\partial J}{\partial \mu_{\mathbf{s},j}} = 0 \quad \Rightarrow \quad \mu_{\mathbf{s},j} = \frac{\sum_{n=1}^{N} v_j(n) r_{\mathbf{s}}(n) e_j(n)}{\sum_{n=1}^{N} v_j(n) r_{\mathbf{s}}(n)}.$$

Since $\boldsymbol{\mu_0}$ is fixed and known, only the centroids for $\mathbf{s} \neq \mathbf{0}$ are updated.

Algorithm 1 : Clustering with missing data

1: initialize $\hat{\boldsymbol{\mu}}$ with the result of the last clustering
2: **repeat**
3: —— **Minimization with respect to r** ——
4: **for** $n = 1$ to N **do**
5: $\quad b_{\mathbf{s}}(n) = \sum_{j=1}^{N_s} v_j(n)(e_j(n) - \hat{\mu}_{\mathbf{s},j})^2$
6: $\quad$ initialize $\hat{r}_{\mathbf{s}}(n) = 0, \forall \mathbf{s}$
7: $\quad \hat{r}_{\mathbf{s}}(n) = 1$, where $\mathbf{s} = \underset{\mathbf{t}}{\operatorname{argmin}} \{b_{\mathbf{t}}(n)\}$
8: **end for**
9: —— **Minimization with respect to μ** ——
10: **for** $j = 1$ to N_S **do**
11: $\quad \hat{\mu}_{\mathbf{s},j} = \frac{\sum_{n=1}^{N} v_j(n) \hat{r}_{\mathbf{s}}(n) e_j(n)}{\sum_{n=1}^{N} v_j(n) \hat{r}_{\mathbf{s}}(n)}, \qquad \forall \mathbf{s} \neq \mathbf{0}$
12: **end for**
13: ————————————————————————
14: **until** $\hat{\mathbf{r}}$ converges

Iterations stop when there is no further change in the assignments $\hat{\mathbf{r}}$. Since each iteration reduces the value of J, the convergence of the algorithm is ensured. However, as in the conventional K-means, the algorithm may converge to a local minimum of J.

The algorithm is run to update $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\pi}}$ whenever a new energy estimate enters the sliding window. Each run is initialized with the last estimate of $\boldsymbol{\mu}$ which was obtained with the previous window of energy vectors. Therefore, the initial values of $\hat{\boldsymbol{\mu}}$ will be close to the new centroids. As a result, the algorithm requires very few iterations to converge.

5. SIMULATION RESULTS

In this section we obtain the ROC (Receiver Operating Characteristic) curve to illustrate the performance of the proposed algorithm in different scenarios. Each ROC curve averages the

LRT results over $4 \cdot 10^5$ energy vectors. Before applying the LRT, we estimate the current clusters centroids and weights by means of the clustering algorithm considering a sliding window with the N most recent energy vectors.

Unless otherwise indicated, we have considered the following assumptions and parameter values in the simulations,

- The channel bandwidth is $W = 5$ MHz.
- The noise spectral density is $\eta = -174$ dBm at all CRs.
- The channels between different pairs PU-CR are independent and identically Rayleigh distributed. For each sensing channel we generate time-correlated realizations using the Jakes' model [13], for a given maximum Doppler shift $f_D = 25$ Hz. Therefore, the coherence time of the sensing channels $T_C \approx \frac{0.4}{f_D} = 16$ ms [14].
- The number of signal samples for energy estimation (2.2) is $M = 100$, therefore, the sensing time is $\tau = \frac{M}{W} = 20 \mu s$
- The PU network state can change between two consecutive energy vectors. The activity of the PUs is modeled as independent and identically distributed homogeneous Markov chains with two states: inactive $(s_i(n) = 0)$ and active $(s_i(n) = 1)$. In the simulations we assume that the transition probabilities are $p_{0,0} = P(s_i(n) = 0|s_i(n-1) = 0) = 0.75$ and $p_{1,1} = P(s_i(n) = 1|s_i(n-1) = 1) = 0.5$. Accordingly, for a PU network with $N_U = 3$ users, the probability that the channel is idle is $\pi_{\mathbf{0}} = 0.3$.
- All PUs transmit identical power. Then, the average SNR is the same for all CRs (averaging over the sensing channel realizations and the PU's activity). The assumption ia mainly made to facilitate the interpretability of the results. Unless otherwise indicated, we assume that the average SNR is -5 dB.
- The CRs (or a subset of them) sense the channel periodically with constant sensing period $T_F = 1$ ms. This value requires the window size to be $N \leq \frac{T_C}{T_F} = 16$.
- The control channels between the CRs and the FC are error-free.

An ideal detector that assumes perfect knowledge of the unknown parameters is referred to as a clairvoyant detector [7]. Its ROC will be shown in some figures as an upper bound. The performance gap between the clairvoyant detector and our detector is the performance loss due to the parameter estimation errors.

At a given time, the total number of energy estimates in the sliding window will be $N_o = \sum_{n=1}^{N} \sum_{j=1}^{N_S} v_j(n)$, which can take values between $N_o = N$ and $N_o = N_S N$. In the simulations, when $N_o < N_S N$, the CRs that report their energy estimates are selected randomly. Obviously, if a CRs selection strategy were used, the performance of the proposed algorithm would be better.

Figure 2 shows the ROC curves for different SNR values when $N_U = 3$ and $N_S = 3$. In this scenario all energy estimates are available at the FC, so $N_o = N_S N$. The ROC curves of the clairvoyant detector are also shown. It can be observed that the performance loss due to the estimation errors is relatively low.

Figure 3 shows the ROC curves for a network with $N_U = 3$ and different number of CRs. Again, we assume that all energy estimates are available at the FC ($N_o = N_S N$). Obviously, the higher the number of sensors the better the performance. As it is also expected, the gain by increasing N_S is higher when the number of CRs is low.

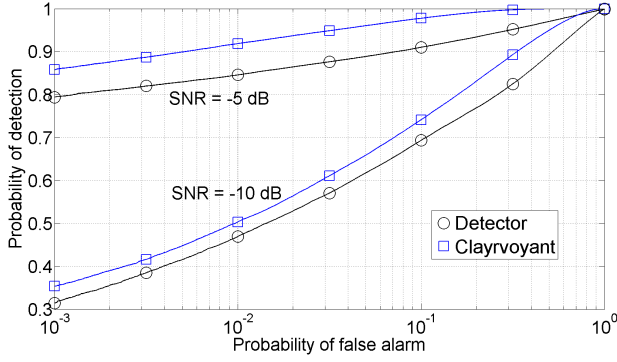Figure 4 compares the ROC curves for different number of energy estimates in the sliding windows (N_o). In this example

Fig. 2. ROC curves for $N_U = 3$, $N_S = 3$ and different values of average SNR.
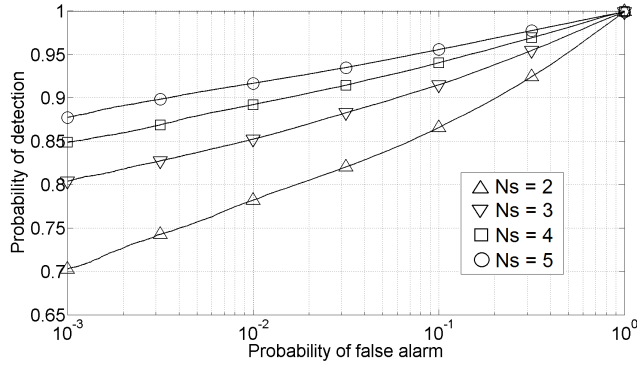

Fig. 3. ROC curves for $N_U = 3$ and different number of sensors.

the number of PUs and CRs are $N_U = 3$ and $N_S = 4$, respectively. In the case $N_o = 4N$ all CRs report their energy estimates to the FC, and hence the energy vectors do not have missing entries, whereas when $N_o = N$ only a single CR reports its energy level each time. As it is expected, the higher the N_0, the better the performance.
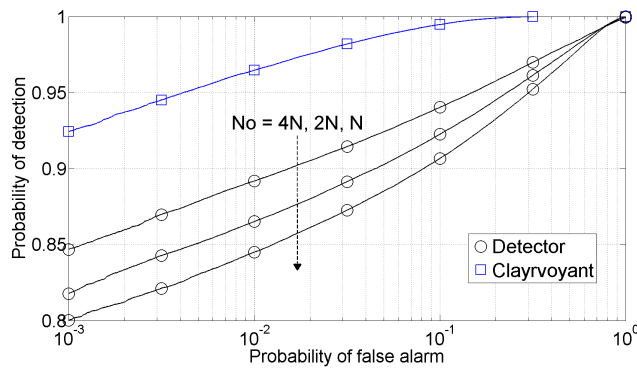

Fig. 4. ROC curves for $N_U = 3$, $N_S = 4$ and different number of energy estimates at the FC.

Figure 5 shows the ROC curves for different values of the sensing period T_F. The window size N is chosen to guarantee that the energy vectors in the sliding windows have been obtained for a constant channel: $N \leq T_C/T_F$. The lower the N the less accurate are the clustering results. The number of PUs and CRs are $N_U = 3$ and $N_S = 4$, respectively, but only the energy estimates from two CRs are available for clustering each time ($N_o = 2N$).

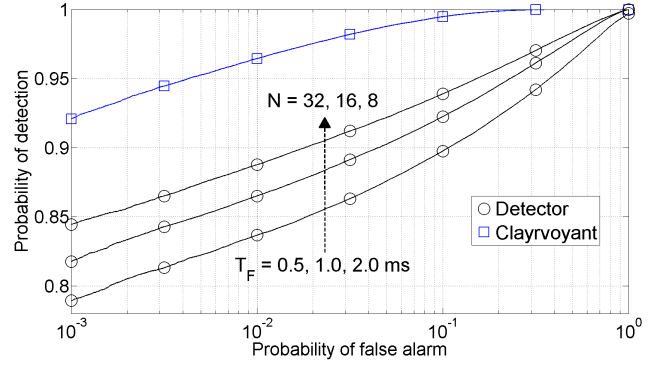Finally, Figure 6 shows the ROC curves for PU networks

Fig. 5. ROC curves for $N_U = 3$, $N_S = 4$, $N_o = 2$ and different sliding window sizes.

with different activity characteristics. The activity of the network is characterized by parameter π_0, that is, is the probability that none of the PUs is transmitting. The more active are the PU users, the lower the π_0. It can be observed that it has a significant influence on the detector performance. The more activity in the network, the better the performance. This is because the clustering algorithm has to learn the parameters of the clusters associated with network states $\mathbf{s} \neq \mathbf{0}$, whereas the parameters of the cluster $\mathbf{s} = \mathbf{0}$ are known in advance. Therefore, the more energy vectors are generated under active network states ($\mathbf{s} \neq \mathbf{0}$), the better the algorithm estimates the parameters of their clusters. The different values of π_0 were obtained by choosing the appropriate values of the transition probability $p_{0,0}$ while keeping $p_{1,1} = 0.5$ fixed.

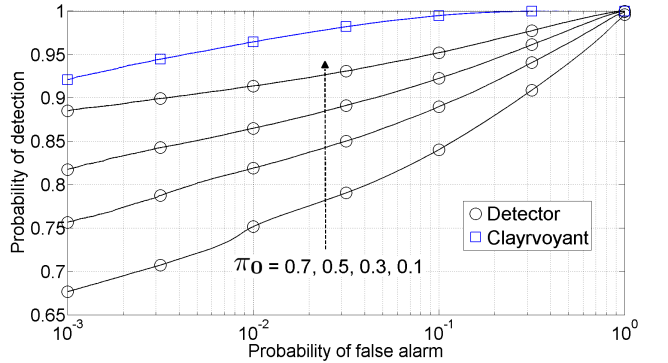**Fig. 6**. ROC curves for $N_U = 3$, $N_S = 4$, $N_o = 2$ and different probability that the channel is idle.

6. CONCLUSIONS

In this work we have presented a novel method for centralized cooperative spectrum sensing in mobile environments where the sensors or cognitive radios perform energy detection. We have posed the detection problem as a composite hypothesis testing problem where the unknown parameters are estimated by means of an adaptive clustering algorithm in the energy space. The input to the clustering algorithm are the most recent local energy levels at the sensors. Due to its adaptive nature, the algorithm is suited for time varying scenarios. In addition, the algorithm is able to deal with missing values, which makes it suited to be applied with any sensor selection strategy to reduce the overhead in the control network. The simulation results show that the proposed method is feasible and efficient in slow fading environments.

7. REFERENCES

[1] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: a survey," *Physical Communications*, vol. 4, no. 1, pp. 40–62, March 2011.

[2] Y. Zeng, Y. C. Liang, A. T. Hoang Liu, and R. Zhang, "A review on spectrum sensing for cognitive radio: challenges and solutions," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, no. 1, January 2010.

[3] J. Ma, G. Zhao, and Y. Li, "Soft combination and detection for cooperative spectrum sensing in cognitive radio networks," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4502–4507, December 2008.

[4] Z. Quan, S. Cui, and A. H. Sayed, "Optimal linear cooperation for spectrum sensing in cognitive radio networks," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 1, pp. 28–40, February 2008.

[5] K. W. Choi, E. Hossain, and D. I. Kim, "Cooperative spectrum sensing under a random geometric primary user network model," *IEEE Transactions on Wireless Communications*, vol. 10, no. 6, pp. 1932–1944, June 2011.

[6] K. M. Thilina, K. W. Choi, N. Saquib, and E. Hossain, "Machine learning techniques for cooperative spectrum sensing in cognitive radio networks," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 10, pp. 2209–2221, November 2013.

[7] S. M. Kay, *Fundamentals of Statistical Signal Processing. Detection Theory*, Prentice Hall. Signal Processing Series, 1998.

[8] T. J. Lim, R. Zhang, Y. C. Liang, and Y. H. Zeng, "GLRT-based spectrum sensing for cognitive radio," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM)*, December 2008, pp. 4391–4395.

[9] J. Font Segura and X. Wang, "GLRT-based spectrum sensing for cognitive radio with prior information," *IEEE Transactions on Communications*, vol. 58, no. 7, pp. 2137–2146, July 2010.

[10] S. Zarrin and T. J. Lim, "Composite hypothesis testing for cooperative spectrum sensing in cognitive radio," in *Proceedings of the IEEE International Conference on Communications (ICC)*, June 2009, pp. 1–5.

[11] S. Zarrin and T. J. Lim, "Cooperative spectrum sensing in cognitive radios with incomplete likelihood functions," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3272–3281, June 2010.

[12] H. Urkowitz, "Energy detection of unknown deterministic signals," *Proc. IEEE*, vol. 55, pp. 523–531, April 1967.

[13] W. C. Jakes, *Microwave Mobile Communications*, John Wiley and Sons, 1975.

[14] T. S. Rappaport, *Wireless Communications*, Prentice Hall, 2002.