

A Novel Adaptive Fusion Scheme For Cooperative Spectrum Sensing

Imen Nasr, Sofiane Cherif

► **To cite this version:**

Imen Nasr, Sofiane Cherif. A Novel Adaptive Fusion Scheme For Cooperative Spectrum Sensing. VTC Fall 2012 - Wireless World 2020 Workshop, iee, 2012. <hal-00747079>

HAL Id: hal-00747079

<https://hal.inria.fr/hal-00747079>

Submitted on 30 Oct 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Novel Adaptive Fusion Scheme For Cooperative Spectrum Sensing

Imen NASR and Sofiane CHERIF

University of Carthage, Engineering College of communications of Tunis (SupCom), Tunisia

Emails: {nasr.imen, sofiane.cherif}@supcom.rnu.tn

Abstract—¹ In cognitive radio systems, the accuracy of spectrum sensing depends on the received primary signal strength at the secondary user (SU). In fact, a single node sensing would be compromised if the received signal strength is not high enough to be detected by this node. In this paper, we propose a cooperative decision fusion rule based on adaptive linear combiner. The weights which correspond to confidence levels affected to SUs, are determined adaptively using the Normalized Least Mean Squares (NLMS) and the Recursive Mean Squares (RLS) algorithms. The proposed algorithms combine the SUs decisions with the adaptive confidence levels to track the surrounding environment. Simulation results show a high adaptability of the proposed scheme, as the operating conditions change. Furthermore, the proposed algorithms do not necessitate a prior knowledge about the PU features and are very efficient compared to conventional decision fusion techniques.

Index Terms—Spectrum Sensing, Decision Fusion, NLMS algorithm, RLS algorithm.

I. INTRODUCTION

Cognitive radio has come up with solutions to the spectrum scarcity and under-utilization [1]. One of the proposed solutions is the access to the spectrum in an opportunistic way providing real time information about the band occupancy. Two major requirements have to be achieved. In the first hand, SUs have to detect spectrum access opportunities using efficient and accurate sensing techniques. In the second hand, they have to stop working once the primary user (PU) comes within an allowed time period. Besides, many SUs may cooperate with each other to ameliorate the system accuracy and reliability [2].

In distributed detection, some fusion algorithms such as OR, AND and Majority rules were studied in [3]. However, simply combining hard local decisions sent from cognitive terminals does not appear to be optimal when the sensors are localized in different environment conditions. This is why, a confidence level could be assigned to every secondary user regarding its decisions history and would be taken into account in the data fusion step [4]. This new parameter should reflect the variability of operating conditions and the secondary user detection efficiency and reliability. An optimal fusion rule was proposed in [5]. To implement this rule, a weight assigned to each SU decision has to be known. Iterative algorithms were proposed in [7] to estimate these weights. However, these

algorithms respond slowly when the system is changed which is not suitable for a time varying environment.

In this paper, an online learning process is presented to estimate adaptively a confidence level assigned to each SU. This confidence level is taken into account in the decision fusion step.

This paper is organized as follows: in Section II, the online learning process is presented. Then in Section III, the adapting algorithms (NLMS and RLS algorithms) to estimate the SUs confidence level are described. The simulation results are represented in Section IV. Finally, the last section concludes the paper.

II. FORMULATION OF THE ONLINE LEARNING

Let us consider a binary hypothesis testing problem with the following two hypotheses:

$$\begin{cases} H_1: \text{The signal is present} \\ H_0: \text{The signal is absent} \end{cases} \quad (1)$$

We consider a system of K SUs which are statistically independent. Every cognitive terminal performs a local sensing to make a local decision d_i , $i = 1, \dots, K$, where

$$d_i = \begin{cases} +1 & \text{if } H_1 \text{ is true} \\ -1 & \text{if } H_0 \text{ is true} \end{cases} \quad (2)$$

As shown in Fig. 1, an overall decision d_c is then computed

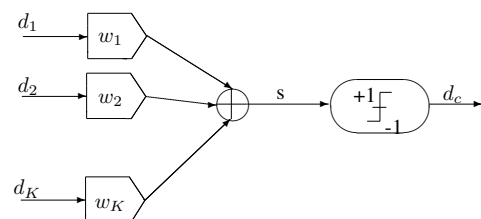


Fig. 1. Fusion rule description.

based on the following fusion rule:

$$d_c = f(d_1, \dots, d_K) = \begin{cases} +1 & \text{if } \sum_{i=1}^K w_i d_i \geq 0 \\ -1 & \text{otherwise} \end{cases} \quad (3)$$

where w_i is a confidence level affected to every secondary user i . Cooperative detection techniques, including Likelihood ratio criterion approaches (Bayesian test [5], Neyman Pearson Test [6], etc), correspond in fact to a suitable weighting of SUs decisions. However, the proposed weights expressions

¹This publication was made possible by NPRP grant No.08-577-2-241 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

supposes that the local detection and false alarm probabilities of each SU are known. An estimation technique of these two parameters was presented in [7], but it needs long training sequences and responds slowly to time varying environment.

The proposed structure may be also seen as the simplest kind of a feedforward neural network (a linear classifier), in particular, the perceptron [8]. In this type of networks, reference signals are given to conduct the secondary users weights, w_i , $i = 1, \dots, K$ to converge to the optimal value allowing to make better decision.

Such a reference is not available in practice because, for our application, no a prior information is given about the PU. Insofar as the fusion center decision is more reliable than local decisions [7], we propose to use this information as a training signal that allows the system to migrate from ill-functioning area to an operating zone of acceptable quality. The objective is to converge later to more powerful functioning zone.

Then, to estimate the coefficients w_i , $i = 1, \dots, K$, we propose an adaptive scheme which is monitored by a conventional cooperation technique. The adaptive scheme is chosen for the system optimization but also for a possible tracking of the environment.

III. ADAPTIVE WEIGHTS ESTIMATION PROCESS

One of the basic component of feedforward networks is the adaptive linear combiner. The proposed scheme is implemented in Fig. 2 to estimate adaptively the SUs weights. The system receives at each time n , an input vector $\mathbf{d}_n =$

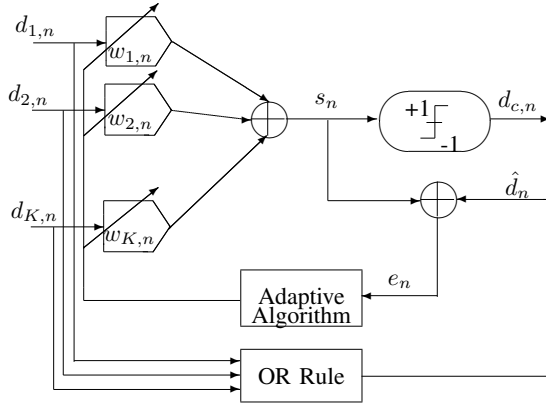


Fig. 2. Adaptive linear combiner.

$[d_{1,n}, \dots, d_{K,n}]^T$ representing all secondary users decisions, and a desired response \hat{d}_n . The decision input vector components are weighted by a set of adaptive coefficients, representing the SUs decisions confidence levels, given by $\mathbf{w}_n = [w_{1,n}, \dots, w_{K,n}]^T$. The inner output s_n is a sum of the weighted inputs.

$$s_n = \mathbf{w}_n^T \mathbf{d}_n \quad (4)$$

The updating error is

$$e_n = \hat{d}_n - s_n. \quad (5)$$

The reference signal \hat{d}_n is obtained by combining the local decisions sent from different SUs using a conventional cooperation technique. Given that accurate and fast PU detection is required in cognitive radio systems, we have chosen the OR rule to compute \hat{d}_n . This fusion rule is known for its high detection probability and low complexity [2]. To adapt the SUs weights we propose to minimize the updating the mean square of error function e_n .

A. Normalized Least Mean Squares (NLMS) algorithm

The LMS algorithm, which is based on the steepest descent method, is commonly utilized to reduce the mean square error [9]. This algorithm could be used to update the local secondary user weights according to the following equation :

$$\mathbf{w}_{n+1} = \mathbf{w}_n - \frac{1}{2} \mu \frac{\partial e_n^2}{\partial \mathbf{w}_n} \quad (6)$$

So the updating equation is:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu e_n \mathbf{d}_n^T \quad (7)$$

The main drawback of the "pure" LMS algorithm is that it is sensitive to the scaling of its input. This makes it very hard to guarantee the stability of the algorithm [10]. The Normalized least mean squares algorithm (NLMS) is a form of the LMS algorithm that solves this problem by normalizing the step size μ with the input power. The updating equation would be:

$$\mathbf{w}_{n+1} = \mathbf{w}_n + \mu e_n \frac{\mathbf{d}_n^T}{\mathbf{d}_n^T \mathbf{d}_n} \quad (8)$$

In our application $\mathbf{d}_n^T \mathbf{d}_n = K$.

B. Recursive Least Squares (RLS) algorithm

The RLS algorithm finds recursively the \mathbf{w} coefficients that minimize a weighted linear least squares cost function $J_{\mathbf{w}}(n)$ related to the input decision vector, where

$$J_{\mathbf{w}}(n) = \sum_{i=0}^n \delta^{n-i} e_i^2 \quad (9)$$

δ is a forgetting factor which helps provide a tradeoff between the algorithm tracking capability and steady state performance. It is typically chosen such that $0 \leq \delta < 1$. If no forgetting factor is needed, we chose $\delta = 1$.

The optimal weight is given by solving the following equation:

$$\nabla_{\mathbf{w}} J_{\mathbf{w}}(n) |_{\mathbf{w}=\mathbf{w}_n} = 0 \quad (10)$$

By solving eq. 10 we lead to the following updating equation:

$$\mathbf{w}_n = \mathbf{Q}_n \mathbf{p}_n \quad (11)$$

where

$$\mathbf{p}_n = \sum_{i=0}^n \delta^{n-i} \hat{d}_i \mathbf{d}_i = \delta \mathbf{p}_{n-1} + \hat{d}_n \mathbf{d}_n, \quad (12)$$

$$\mathbf{Q}_n = \mathbf{R}_n^{-1} \quad (13)$$

and

$$\mathbf{R}_n = \sum_{i=0}^n \delta^{n-i} \mathbf{d}_i \mathbf{d}_i^T = \delta \mathbf{R}_{n-1} + \mathbf{d}_n \mathbf{d}_n^T. \quad (14)$$

The matrix \mathbf{Q}_n is computed using (14) and the Matrix Inversion Lemma [11], leading to the following expression:

$$\mathbf{Q}_n = \delta^{-1} \mathbf{Q}_{n-1} - \delta^{-1} \mathbf{k}_n \mathbf{d}_n^T \mathbf{Q}_{n-1} \quad (15)$$

where $\mathbf{k}_n = \frac{\delta^{-1} \mathbf{Q}_{n-1}}{1 + \delta^{-1} \mathbf{d}_n^T \mathbf{Q}_{n-1} \mathbf{d}_n}$ (called Kalman gain).

By using (12) and (11) we obtain the following updating equation:

$$\mathbf{w}_n = \mathbf{w}_{n-1} + \mathbf{k}_n (\hat{d}_n - \mathbf{w}_{n-1}^T \mathbf{d}_n) \quad (16)$$

IV. SIMULATION RESULTS

We consider a centralized cognitive radio network with K secondary users. We assume that the received signal to noise ratios (SNR) for each SU are different. Spectrum sensing is performed using energy detection technique, where decision thresholds are all set identical for each SU. Sensing performance at each SU is evaluated by the probability of detection corresponding to hypothesis H_1 and the probability of false alarm corresponding to H_0 . These performances depend on the SNR determined at each SU.

A. Check for the algorithms convergence

In this paragraph, we evaluate the proposed algorithm in terms of convergence rapidity and correct weights estimation. Fig. 3 shows the evolution of the confidence levels versus time.

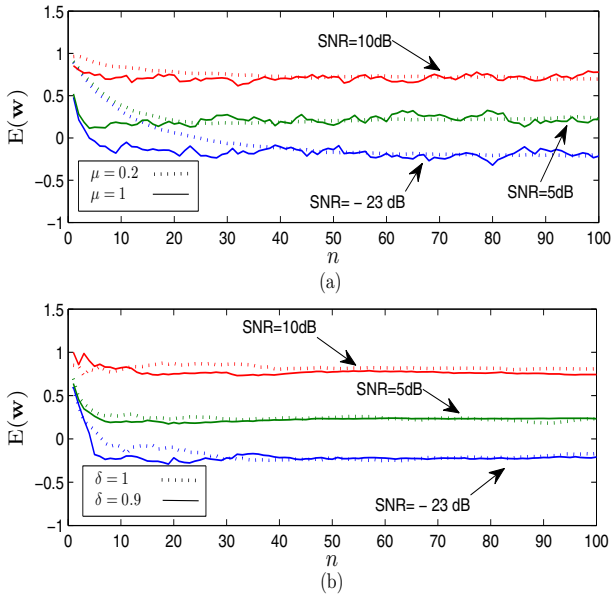


Fig. 3. Mean of the confidence level vector versus time ($K = 3$): (a) NLMS algorithm (b) RLS algorithm.

We have chosen $K = 3$ SUs with different SNRs: 10, 5 and -23 dB, respectively. The NLMS and RLS algorithms both converge to the same weights vector. When changing the step

size μ , the NLMS algorithm convergence speed and fluctuation around the mean value change. The same effect is observed by adding a forgetting factor $\delta < 1$ to the RLS algorithm. It is also clear that the SU with the highest SNR has the highest weight. In Fig. 4 we present the Mean Square Deviation of \mathbf{w}

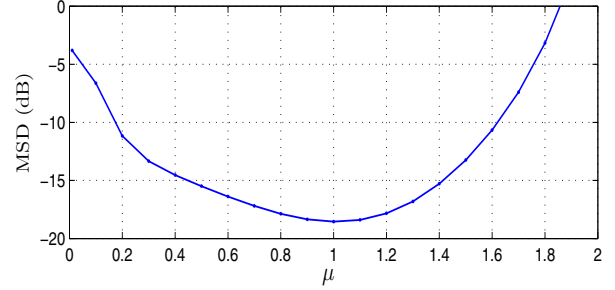


Fig. 4. MSD versus μ for NLMS algorithm.

($\text{MSD}(\mathbf{w}) = E \left((\mathbf{w} - E(\mathbf{w}))^T (\mathbf{w} - E(\mathbf{w})) \right)$) versus μ for the NLMS algorithm. The MSD is evaluated at time $n = 100$. Fig. 4 shows that The MSD is minimized for $\mu = 1$. The MSD was also evaluated in terms δ for the RLS algorithm. The minimum value is reached for $\delta = 0.98$. The corresponding values lead to a faster and less fluctuant convergence.

B. System performances evaluation

To evaluate the system efficiency and reliability, we need to compute detection and false alarm probabilities at the fusion center, Q_d and Q_f . Fig. 5 shows the fusion center

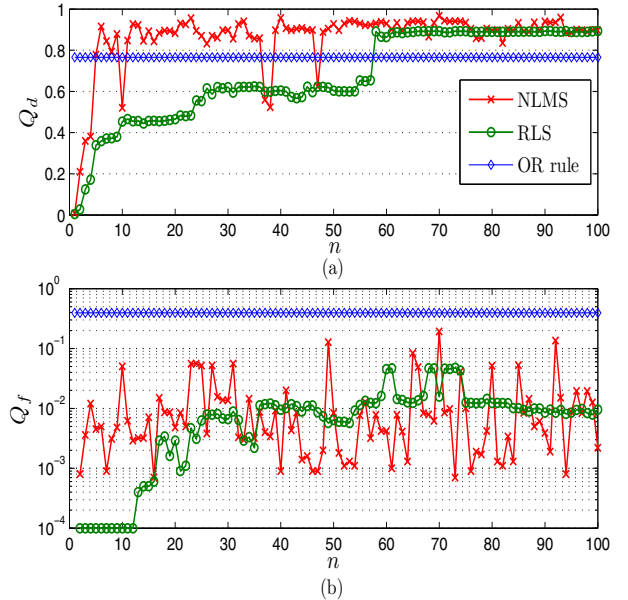


Fig. 5. Performance of the proposed approach: Q_d and Q_f versus time ($K = 10$).

performances using the RLS and NLMS algorithms for $\mu = 1$ and $\delta = 1$. The algorithms begin with an initialization phase

among the first 60 iterations then converge to the optimal weights value (the same behavior as in Fig. 3). At time $n = 100$, we obtain $Q_d = 0.9$ for the adaptive scheme and 0.8 for the OR rule, while, $Q_f = 10^{-2}$ for the adaptive scheme and 0.4 for the OR rule. Then, the proposed technique may reach a high detection level and reduce the false alarms rate compared to the OR fusion rule (used to compute the reference signal). In Fig. 6, we plotted the ROC curves related

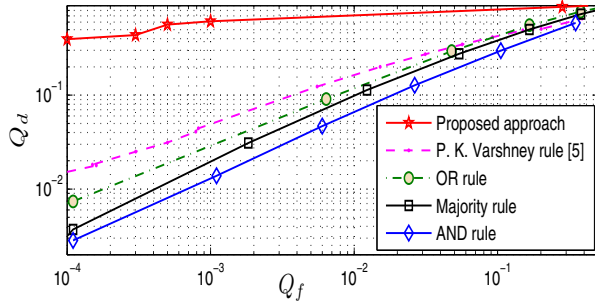


Fig. 6. ROC curves ($K = 10$ and $P_f = 0.1$).

to the proposed adaptive algorithm, the P. K. Varshney Rule [5] and the conventional fusion rules [2] (OR, AND and Majority rules). The performances of the adaptive algorithm are obtained by computing the value of \mathbf{w} after 100 iterations (both algorithms, NLMS and RLS, converge to the same weight vector). We assume having $K = 10$ SUs whose SNRs are different and take values in $[-10 \text{ dB}, 10 \text{ dB}]$. The local decision threshold is computed for a local false alarm probability $P_f = 0.1$. Under these conditions, the detection probability at the fusion center for $Q_f = 10^{-3}$ is, $Q_d = 0.6$ for the proposed scheme, 0.05 for P. K. Varshney Rule [5] and 0.03 for the OR rule. It is obvious that the proposed technique outperforms the other fusion rules.

C. Environment tracking

The SU environment varies when the received SNR changes or when the PU state switches from idle to active and vice versa. In order to study the tracking behavior of the algorithms,

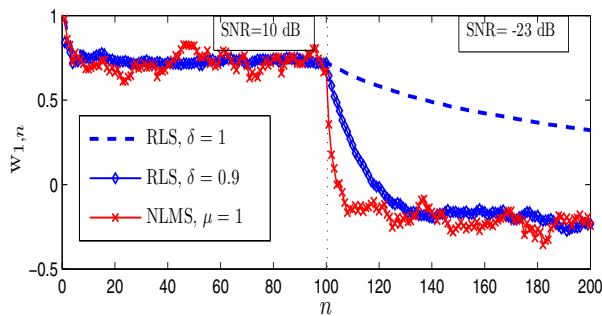


Fig. 7. Received SNR at the 1st SU changes ($K = 3$ and $P_f = 0.1$).

we changed, in Fig 7, the SNR of the 1st SU from 10

to -23 dB at iteration 100. The local decision threshold is computed for a local false alarm probability $P_f = 0.1$. We notice that the RLS algorithm is able to adapt faster to the changes by decreasing δ . The NLMS algorithm has excellent tracking performance and may outperform the RLS algorithm. A similar result is obtained in Fig. 8. When changing the

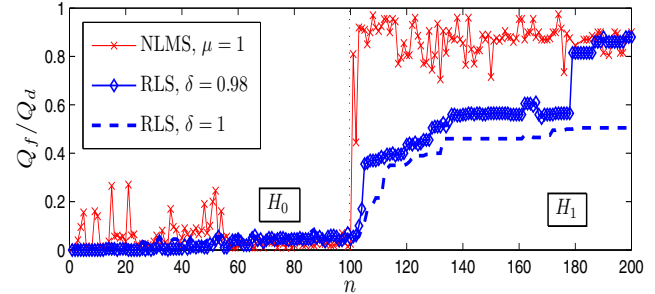


Fig. 8. PU status changes ($K = 10$ and $P_f = 0.1$).

hypothesis from H_0 to H_1 (at time $n=100$), the probability of detection increases from low values when the PU is absent (it corresponds in this case to the false alarm probability: Q_f) to much higher values when the PU is present (it corresponds in this case to the detection probability: Q_d).

V. CONCLUSION

In this paper, we proposed an adaptive cooperative spectrum sensing scheme where confidence levels are assigned to every secondary decision. This technique seems to be good due to its ability to track the SU environment.

In these algorithms, we tried to minimize the mean square error on decision making. We pointed an amelioration in terms of detection and false alarm probabilities compared to the optimal fusion rule [5] and the conventional fusion rules [2]. The proposed scheme is then robust in time varying environment presenting high detection performances.

REFERENCES

- [1] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol.23, no.2, pp. 201- 220, Feb. 2005.
- [2] K. Ben Letaief and W. Zhang, "Cooperative Communications for Cognitive Radio Networks," *Proceedings of the IEEE*, vol.97, no.5, pp.878-893, May 2009
- [3] E. Peh and Y. Liang, "Optimization for Cooperative Sensing in Cognitive Radio Networks," *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, pp.27-32, 11-15 March 2007
- [4] P. Qihang, Z. Kun, W. Jun and Li Shaoqian, "A Distributed Spectrum Sensing Scheme Based on Credibility and Evidence Theory in Cognitive Radio Context," *IEEE 17th International Symposium on Personal Indoor and Mobile Radio Communications*, pp.1-5, 11-14 Sept. 2006.
- [5] Z.Chair and P.K. Varshney, "Optimal Data Fusion in Multiple Sensor Detection Systems", *IEEE Transactions on Aerospace and electronic systems*, vol.AES-22, no.1, pp.98-101, Jan. 1986.
- [6] J. Hillenbrand, T. A. Weiss, and F. K. Jondral, "Calculation of detection and false alarm probabilities in spectrum pooling systems", *IEEE Communications Letters*, vol.9, no.4, pp. 349- 351, April 2005.
- [7] N. Ansari, E.S.H. Hou, B. Zhu and J. Chen, "Adaptive fusion by reinforcement learning for distributed detection systems", *IEEE Transactions on Aerospace and Electronic Systems*, vol.32, no.2, pp.524-531, April 1996.

- [8] M. Minsky and S. Papert, *Perceptron Expanded Edition*. Cambridge, MA: MIT Press, 1988.
- [9] B. Widrow and M. A. Lehr, "Feedforward Networks", *INNS World Congress On Neuron Networks*, Dec. 1992.
- [10] Simon Haykin, "Adaptive Filter Theory, Prentice Hall", 2002, ISBN 0-13-048434-2.
- [11] D.J. ylavsky and G.R.L. Sohie, "Generalization of the matrix inversion lemma", *Proceedings of the IEEE* , vol.74, no.7, pp. 1050- 1052, July 1986