Analysis of Spectrum Detection and Decision Using Machine Learning Algorithms in Cognitive Mobile Radio Networks

Pablo Palacios Játiva[1(✉)], Cesar Azurdia-Meza[1], Iván Sánchez[2], David Zabala-Blanco[3], and Milton Román Cañizares[4]

[1] Department of Electrical Engineering, University of Chile, Santiago, Chile
pablo.palacios@ug.uchile.cl, cazurdia@ing.uchile.cl
[2] Department of Telecommunications Engineering, Universidad de las Américas, Quito, Ecuador
ivan.sanchez.salazar@udla.edu.ec
[3] Department of Computing and Industries, Universidad Católica del Maule, Talca, Chile
[4] Departamento de Ingeniería de Comunicaciones, Universidad de Málaga, Málaga, Spain
0610744939@uma.es

Abstract. In this work, the performance of four Machine Learning Algorithms (MLAs) applied to Cognitive Mobile Radio Networks (CMRNs) are analyzed. These algorithms are Coalition Game Theory (CGT), Naive Bayesian Classifier (NBC), Support Vector Machine (SVM), and Decision Trees (DT). The numerical results of the performance analysis of these algorithms are presented based on two metrics. These metrics are commonly used in CMRNs which are Probability of Detection (P_d) and Probability of False Alarm (P_{fa}) against Signal-to-Noise Ratio (SNR). Furthermore, outcomes regarding the Classification Quality (CQ) and the simulation time are exposed. Theoretical and numerical results show that the SVM outperforms the rest of the algorithms in each of the metrics. The reasons behind this come from the SVM features, namely high precision, fast learning, and simplicity in the realization stage.

Keywords: Cognitive mobile radio networks (CMRNs) · Coalition game theory (CGT) · Support vector machine (SVM) · Decision tree (DT) · Machine learning algorithms (MLAs) · Naive bayesian classifier (NBC)

1 Introduction

Communications based on Cognitive Radio (CR) have been studied in recent years because they use the electromagnetic spectrum efficiently [1]. This use

X. Wu et al. (Eds.): QShine 2020, LNICST 381, pp. 142–153, 2021.
https://doi.org/10.1007/978-3-030-77569-8_11

efficiency occurs when performing frequency band jumps between wireless protocols and technologies. Furthermore, by complying with this paradigm, CR is considered an enabling technology for 5G communications. Another feature of CR is that its physical layer is a radio that changes its transmission characteristics depending on the communications environment. This adaptation occurs by detecting spectral holes and efficiently using the available frequencies. These benefits present CR in the short term as the best performing solution to achieve high data rates in wireless communications and enable large-scale user mobility. However, CR's biggest challenge is identifying primary users (PUs) who are using a wide range of spectrum, at a certain time, and in a specific geographic location. On the other hand, CR implementations must meet the following criteria: there is no interference between secondary (SU) (unlicensed) users and PUs [2].

For a communications system to be considered CR-based, it must fulfill a cognitive process, which requires four steps: spectrum detection, spectrum decision, spectrum sharing, and spectrum mobility [3]. In the last decade, investigations regarding the CR topic have been oriented in the field of spectrum detection and decision. Consequently, several techniques have been proposed, such as Energy Detection (ED), Cycle-Stationary Detection (CD), Singular Value Decomposition (SVD) [4], and Eigen-Value Decomposition (EVD). Nevertheless, in order to ensure that the CR devices to be truly conscious of the frequency changes that occurs in the mobility stage, just to improve the efficiency, is imperative that equipped with learning and reasoning functionalities.

In the search for mechanisms to mitigate problems in the spectrum detection and decision stages in CR systems, Machine Learning Algorithms (MLAs) have received a lot of attention from the scientific community. In the context of future networks (CR, femto/small cells, and heterogeneous networks), in [5], the authors present a problem formulation and methodology of several MLAs in terms of effectiveness in the testing stage. This analysis is done because MLAs present a new paradigm of proactive, self-aware, self-adaptive, and predictive networks. The authors conclude that the benefits of the MLAs will be verified in the next generation networks. In [6], the authors present a multiple antenna CR system, in which Support Vector Machine (SVM) algorithms are used to solve the spectrum detection problem. This work shows that the SVM algorithms applied to spectrum detection, specifically of spectral holes detection, are robust in terms of temporal and spatio-temporal detection. In [7], another application proposes an MLA-based solution for CR at the end-user device level. The authors conclude that in terms of the terminal service experience and user behavior, the complexity of the central CR network can be reduced.

The main contribution of this manuscript is the performance evaluation of the Coalition Game Theory (CGT), Naive Bayesian Classifier (NBC), Support Vector Machine (SVM), and Decision Trees (DT) methods applied and adapted to a single Cognitive Mobile Radio Networks (CMRNs) by using the Network Simulator 3 (NS-3.23) modules [8]. This software validates several Machine Learning (ML) Approaches in a functional mobile network. The performance of these algorithms are analyzed in terms of the Probability of Detection (P_d), Probability of False Alarm (P_{fa}), Classification Quality (CQ) and simulation time, by employing numerical simulations and by obtaining Cumulative Probability Distributions (CDFs).

The organization of this manuscript is presented as follows: the CMRN, PUs and SUs models are explained in Sect. 2. In Sect. 3, the proposed MLAs are described with their respective mathematical formulation. Then, in Sect. 4, The performance of the CR with the MLAs applied and the numerical results of the simulations are discussed. Finally, the conclusions are presented in Sect. 5.

2 Cognitive Radio System Model

To evaluate the spectrum detection and spectrum decision, we use the standard MLA, which is composed by numerous input, hidden layers with different number of neurons, and various output applied to CMRNs, implemented to coexist with a primary wireless network composed of two state-of-the-art technologies, which are Wireless Fidelity (WiFi) and Long Term Evolution (LTE). We supposed an area covered by CMRN like the propose scenario, composed of m source-destination PUs pairs. The primary transmitters set is as follow $P_p = (P_{1p}, P_{2p}..., P_{mp})$, while the corresponding receivers set is $P_r = (P_{1r}, P_{2r}...., P_{mr})$. We assume the coexistence of l secondary transmitter in the set $S_s = (S_{1s}, S_{2s}..., S_{ls})$, and their corresponding receivers in the set $S_r = (S_{1r}, S_{2r}..., S_{lr})$. This scenario is presented in Fig. 1.

2.1 Secondary User Model

In a PU network, a single SU is considered to access the licensed bands without interfering with the communication of the PUs. We define t as the time slot and i as the frequency bin, where $t = 1, 2, ..., n$ and $i = 1, 2, ..., k$ respectively. We also define n as the number of time slots and k as the number of frequency bins. By using the SVD detection method [4], the spectrum sensing problem can be formulated as follows

$$x_i(t) = \begin{cases} n_i(t) & H_0 \\ h_i(t) * s_i(t) + n_i(t) & H_1 \end{cases}, \tag{1}$$

here, $x_i(t)$ is the signal received by the SU at the t^{th} time slot in the i^{th} frequency bin, $s_i(t)$ is the signal transmitted by the PU, $n_i(t)$ is the Additive White Gaussian Noise (AWGN), and $h_i(t)$ is the channel gain. H_0 and H_1 are the hypothesis test that indicates whether the SU is using the corresponding channel or not.

We used the SVD detection method as the spectrum detection technique because of its easy to design and efficiency in terms of P_d over other detection methods. The spectrum detection process carried out by the SVD method and its operating characteristics are explained in more detail in [4]. The spectrum status $SS_i(t)$ is given as follows

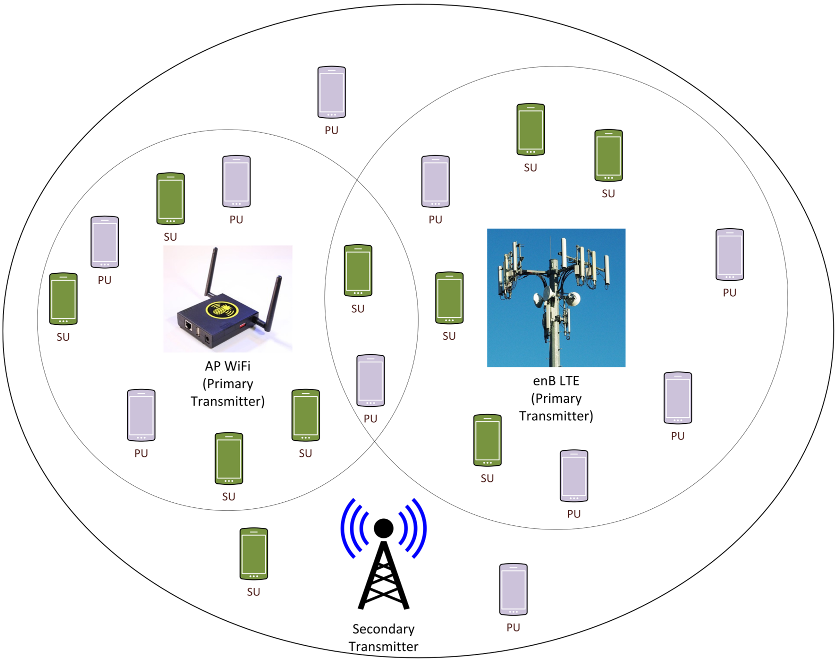

Fig. 1. Scheme of the proposed CMRN.

$$SS_i(t) = \begin{cases} 1 \; x_i(t) > \lambda \\ 0 \; x_i(t) < \lambda \end{cases}. \tag{2}$$

The occupancy O_c^i for the i^{th} time slot acquires the form of

$$O_c^i = (\sum_{t=1}^{k} SS_i(t))/k, \tag{3}$$

which means the average state of the spectrum between the used frequencies bins. When a PU uses some frequency bin that is being sensed by some SU, the $SS_i(t)$ is increased. Consequently, the O_c^i will take large values, by highlighting that the probability of using that frequency bin for the SU is low. Conversely, if $SS_i(t)$ decreases, O_c^i also decreases and it is more likely that this frequency bin is idle and can be used.

2.2 Primary User Model

The PU status is represented as PU^i. This status, for the i^{th} time slot, can be chosen according the rules that follow

$$PU^i = \begin{cases} 1 & (O_c^i > U_{oc}) \ or \ (L_{oc} \leq O_c^i \leq U_{oc} \\ & and \ ffb^i < \overline{ffb}) \\ 0 & (O_c^i < L_{oc}) \ or \ (L_{oc} \leq O_c^i \leq U_{oc} \\ & and \ ffb^i \geq \overline{ffb}) \end{cases}, \tag{4}$$

The maximum and minimum values of occupancy for the n time slots are represented by U_{oc} and L_{oc}. The consecutive free frequency bins in the i^{th} time slot are denoted by ffb^i, whose maximum value when the PU is present is denoted as $\overline{ffb}$.

2.3 Steps for an Overall Machine Learning

First, MLA constructs a classifier to map SS_i to PU^i, where $SS_i = (SS_i(1), SS_i(2)..., SS_i(k))$ represents the feature vector. There are three steps for constructing a classifier, which are:

Training: For the training stage, we denote a training vector for the spectral state, $SS_{i(train)} = (SS_{i(train)}(1), SS_{i(train)}(2)....., SS_{i(train)}(k))^T$. Then, we denote a variable for the training PU status, PU^i_{train}. For the total cases, $i = 1, 2, ..., n_1$, where n_1 is the number of training time slots fed into the classifier.

Testing: For the training stage, we denote a testing vector for the spectral state, $SS_{i(test)} = (SS_{i(test)}(1), SS_{i(test)}(2)....., SS_{i(test)}(k))^T$, Then, we denote a variable fot the testing PU status, PU^i_{test}. For the total cases, $i = n_1 + 1, n_1 + 2, ..., n_2$, n_2 refers to the length of testing sequence. In this work, the matrix of size $n * k$ is divided into 10% training data matrix of size $n_1 \times k$ and 90% testing data matrix of size $n_2 \times k$ [9].

Classification Quality: For the classification quality stage, we denote the PU status for the i^{th} time slot as PU^i_{eval}. In this stage it is categorized the testing vector $SS_{i(test)}$ as an occupied class $(PU^i_{eval} = 1)$ or unoccupied class $(PU^i_{eval} = 0)$. The PU status is correctly determined, when $PU^i_{eval} = PU^i_{test}$, by producing $CQ^i = 1$. This scenario will be represented as $P_d = 1$ or 0 depending on the value of PU^i_{eval}. The no-detection occurs when $PU^i_{eval} = 0$ and $PU^i_{test} = 1$, whereas false alarm occurs when $PU^i_{eval} = 1$ and $PU^i_{test} = 0$, by giving $CQ^i = 0$. This situation will be represented as P_{fa}.

3 Proposed Machine Learning Algorithms

In this work, four MLA are analyzed to predict the PU status via the occupancy data. The motivation to use these algorithms is to find the most efficient MLA for predicting future status.

3.1 Coalition Game Theory

We formulate the cooperative problem as a coalition game $G = (N; u)$, where $N = SS_i$ and u represents the payoff function that transform a user contribution in a coalition into its profit. We formulate this scheme via two steps, as follows [2].

The Cooperation Phase: First, local detection is required, which is done using the SVD method. In a Rayleigh fading environment, the P_d and P_{fa} of the i-th SU are respectively represented as $P_{d,i,j}$ and $P_{f,i,j}$, which are given by [2]

$$P_{d,i,j} = [PY_{i,j} > \lambda | H_1] = e^{-\frac{\lambda}{2}} \sum_{n=0}^{w-2} \frac{1}{n!} \left(\frac{\lambda}{2}\right)^n + \left(\frac{1+\gamma_{i,j}}{\gamma_{i,j}}\right)^{w-1} \left[e^{-\frac{\lambda}{2(1+\gamma_{i,j})}} - e^{-\frac{\lambda}{2}} \sum_{n=0}^{w-2} \frac{1}{n!} \left(\frac{\lambda * \gamma_{i,j}}{2(1+\gamma_{i,j})}\right)^n \right], \tag{5}$$

and

$$P_{fa,i,j} = [PY_{i,j} > \lambda | H_0] = \frac{\Gamma\left(w, \frac{\lambda}{2}\right)}{\Gamma(w)}, \tag{6}$$

where $Y_{i,j}$ is the normalized output of the i-th SU sensing the status of the j-th PU, λ is the detection threshold for the j-th PU, w is the time-bandwidth product, and $\gamma_{i,j}$ denotes the average SNR of the received signal from the PU to the SU. Furthermore, $\Gamma(.,.)$ and $\Gamma(.)$ are the incomplete and complete Gamma functions respectively.

In addition, the missing probability P_m for the i-th SU is considered as follows

$$P_{m,i} = 1 - P_{d,i,j}. \tag{7}$$

By reducing the $P_{m,i}$ directly maps to increasing the P_d and, consequently, interference on the PU decreases.

Within each coalition Ω, a single SU, named as the coalition head, k, collects the sensing bits from the coalition SUs, and acts as a fusion center to decide on the presence or not of the PUs in the channel.

The missing and false alarm probabilities are as follows

$$Q_{m,\Omega} = \prod_{i \in \Omega} [P_{m,i} * (1 - P_{e,i,k}) + (1 - P_{m,i}) * P_{e,i,k}], \tag{8}$$

and

$$Q_{f,\Omega} = 1 - \prod_{i \in \Omega} [(1 - P_{fa}) * (1 - P_{e,i,k}) + P_{fa} * P_{e,i,k}]. \tag{9}$$

A suitable function acquires the form of

$$u(\Omega) = Q_{d,\Omega} - C(Q_{f,\Omega}) = (1 - Q_{m,\Omega}) - C(Q_{f,\Omega}), \tag{10}$$

where $Q_{d,\Omega}$ denotes the probability detection of the coalition Ω and $C(Q_{f,\Omega})$ represent a cost function of the P_{fa} within the coalition Ω. The latter can be written as

$$C(Q_{f,\Omega}) = \begin{cases} -\alpha^2 log(1 - \left(\left(\frac{Q_{f,\Omega}}{\epsilon}\right)^2\right) & Q_{f,\Omega} < \epsilon \\ +\infty & Q_{f,\Omega} \geq \alpha \end{cases}, \tag{11}$$

where ϵ denotes a false alarm constraint per coalition, namely per SU.

The SU Transmission Phase. For the SU transmission phase we assume a time division multiplexing [10]. Then, the transmission is divided according the SU contribution in Ω. The time allocated for SU is given by $(1 - \alpha_P) * t_i^{\Omega}$. Its reward is made proportional to the energy spent by the SU.

3.2 Naive Bayesian Classifier

This algorithm is named as the "independent feature model" because it does not consider the features interdependence. In this model, the total samples are contained in the feature vector for the i^{th} time, Furthermore, these samples are independent of each other, because of every feature represents a specific frequency bin. However, the variable of the PU status PU^i, results a function of the frequency bin. The probability of SS_i by using the Bayes theorem is defined as [11]

$$\mathrm{p}(\mathrm{PU}^i, \mathrm{SS}_i) = \mathrm{p}(\mathrm{PU}^i) * \mathrm{p}(\mathrm{SS}_i|\mathrm{PU}^i). \tag{12}$$

When $PU^i = 0$, SS_i is classified as an idle class; otherwise SS_i is an occupied class. The goal is to obtain the class with the largest posterior probability in the classification phase. The classification rule is represent as follows

$$classify(S\hat{S}_i) = \underset{SS_i}{\arg\max} \left\{\mathrm{p}(\mathrm{PU}^i, \mathrm{SS}_i)\right\}, \tag{13}$$

where $S\hat{S}_i = \{S\hat{S}_i\,(1)\,,\ S\hat{S}_i\,(2)\,.\,.\,.\ \{S\hat{S}_i(k)\}$.

3.3 Support Vector Machine

This algorithm results in a discriminative classifier with high accuracy. In addition, SVM tends to be resistant to over-fitting and. Generally, two types of classifiers in SVM are presented in the literature: linear and non-linear SVM. In this work, for simplicity but without losing the generality, linear SVM is employed.

The training feature and response vectors are represented as $D = \left(\mathrm{PU}^i,\ \mathrm{SS}_i\right)$, where $PU^i \in \{0,\ 1\}$. By definition, the two classes of SVM are separated defining a hyperplane H, which is represented as $x * SS_i = \rho$, where x represents the normal vector and ρ represents the constant separating occupied and idle classes ($PU^i \in \{0,\ 1\}$), which in turn is defined as [12]

$$PU^i = \begin{cases} 1 \rightarrow x * SS^i > \rho(\text{Occupied class}) \\ 0 \rightarrow x * SS^i < \rho(\text{Idle class}) \end{cases} \tag{14}$$

3.4 Decision Trees

In this work, decision trees are represented with a classificatory approach, where the leaves of the tree define the class labels. A benefit of DTs is that they can handle interactions and feature dependency. Regarding the decision made by this algorithm, it is made at each node internally, which allows the data division into two own subsets. The data is represented ad follows

$$(\mathrm{SS}_i, \mathrm{PU}^i) = \{(SS_i\,(1)\,,\ SS_i\,(2)\ldots\ SS_i(k)), \mathrm{PU}^i\}, \tag{15}$$

where PU^i is the dependent variable, which is assigned by calculating the entropy of the feature as follows [13]

$$\text{Entropy(t)} = -\sum_{idi=0}^{Z} p(idi|t) * log_2(p(idi|t), \tag{16}$$

where $p(idi|t)$ is the fraction of records belonging to class idi for a certain node t, and Z represents the total classes.

4 Results

4.1 Simulation Parameters

Each of the MLAs were implemented in a simulated CR environment. To generate the simulations, we use the NS-3 software, because it provides executable models of signal propagation and user mobility. We have chosen for the propagation model, the range propagation loss model, due to its unique end-user and transmitter dependency. For the mobility model we chose the random waypoint model. To create a more realistic environment, we created two types of SU: SUs with cognitive capacity to work only on LTE or WiFi and SUs with cognitive capacity for both technologies (dual SU). The most important technical parameters used in the simulation scenarios are shown in Table 1.

Table 1. Technical parameters of the simulation

Parameter	Value
AP coverage	50 m
Channel model	Slow Rayleigh fading
CR LTE/WiFi (SU)	5
Dual CR (SU)	10
eNB cells	3
eNB coverage	350 m
LTE frequency	729 MHz [4]
LTE bandwidth	20 MHz [4]
Mobility model	Random way-point
Noise model	AWGN
Propagation model	Range propagation loss
PU LTE/WiFi	5
Receiver power	0.06 mW
Samples	Variable
Transmitter power	0.037 mW
WiFi bandwidth	20 MHz [4]
WiFi frequency	2400 MHz [4]

4.2 Numerical Simulation Results

The MLA curves were obtained through the implementation and simulation of the algorithms in NS-3. The P_d vs SNR and P_{fa} vs SNR are presented as a Cumulative Distribution Function (CDF) for the MLAs, as shown in Fig. 2 and Fig. 3. As can be seen in both figures, the algorithm that presents the worst performance is the NBC, due to its features of not having complete information and making decisions based on statistics. CGT and DT algorithms have a similar behavior, because the first takes a cooperative detection between nodes, and the second divides the decisions into subsets, which are similar processes for the system. Finally, the algorithm with the best performance is the SVM, due to its high precision and accuracy when recognizing the use patterns of frequency bins for detection. For low levels of SNR, specifically −10 dB, SVM has a P_d of 50% and a P_{fa} of 5%, while NBC has a P_d of 20% and a P_{fa} of 7%.

Figure 4 shows the plot of the Classification Quality as a function of Number of Samples. As N_s increases, they have more chance to sense and sensing accuracy improves. However, by increasing the N_s, the algorithms have a greater amount of data to process and become slower.

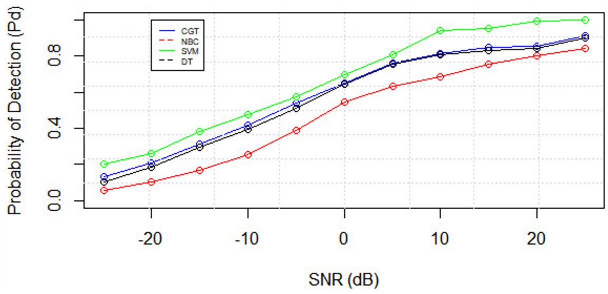

Fig. 2. Probability of detection CDF.

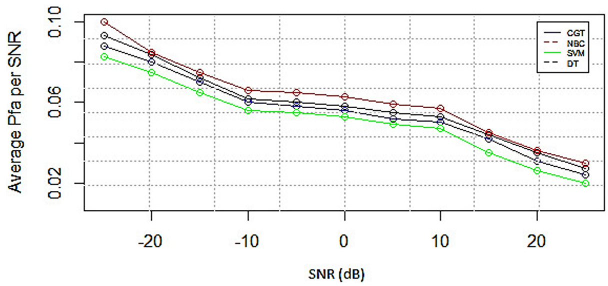

Fig. 3. Probability of false alarm CDF.

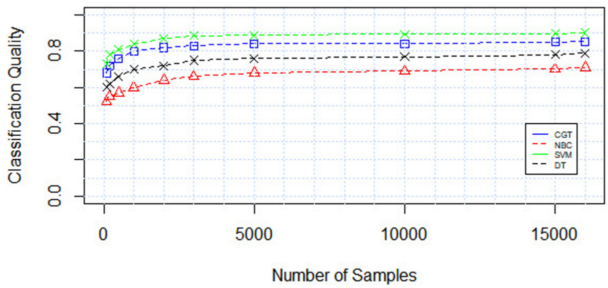

Fig. 4. Classification quality curve.

An important factor that must be considered by the simulator to develop the experiments in a controlled way is the simulation time, since it is not the same magnitude as the real time. To determine these times, several simulations were run with different N_s, maintaining the basic technical parameters indicated in Table 1 in each of them. We defined the number of simulations for the experiment using the Monte Carlo method, with 21 iterations for all variations of SUs. This process was done to have reliable and valid statistics of the generated data [14]. We observe the linear and increasing behavior of Simulation time vs Number Of Samples in Fig. 5. The SVM algorithm presents slightly better

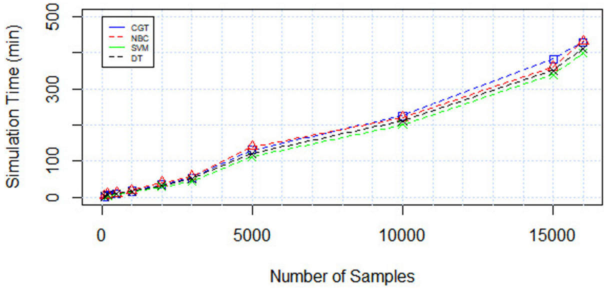

Fig. 5. Simulation time as a function of number of samples.

performance than the other MLAs, due to its simplicity of learning. Specifically the biggest difference is found when N_s is 15000, the SVM has a simulation time of 340 min approximately, while the CGT that presents the worst performance, has a simulation time of 390 min.

5 Conclusions

In this work, some MLA implemented in the detection and decision stage of a CMRN have been analyzed and compared. We modeled the detection and decision stage of the "Cognitive Cycle" as a Coalition Game (CG), Naive Bayesian Classifier (NVC), Support Vector Machine (SVM) and Decision Tree (DT) problem with their respective algorithms. We characterized the network structure resulting from the proposed techniques, its stability and performance was studied and observed in terms of P_d, P_{fa}, CQ and simulation time. Simulation results showed that SVM algorithm, compared with CG, NVC and DT, outperform the CMRN system, based on the parameters studied, specifically in 40 min less in simulation time, compared to the worst performance algorithm (CGT), keeping the N_s fixed at 15000. It also increases the P_d by 30% and decreases the P_{fa} by 2%, compared to the NBC algorithm, which showed the worst performance in these parameters, keeping the SNR fixed at -10 dB.

Acknowledgment. This work was supported by ANID PFCHA/Beca de Doctorado Nacional/2019 21190489, SENESCYT "Convocatoria abierta 2014-primera fase Acta CIBAE-023-2014", UDLA Telecommunications Engineering Degree, Project FONDECYT No. 11160517, and Grupo de Investigación en Inteligencia Artificial y Tecnologías de la Información (IA&TI).

References

1. Sasipriya, S., Vigneshram, R.: An overview of cognitive radio in 5G wireless communications. In: 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1–5, December 2016

2. Palacios, P., Saavedra, C.: Coalition game theory in cognitive mobile radio networks. In: Botto-Tobar, M., Pizarro, G., Zúñiga-Prieto, M., D'Armas, M., Zúñiga Sánchez, M. (eds.) CITT 2018. CCIS, vol. 895, pp. 3–15. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-05532-5_1
3. Palacios Játiva, P., Saavedra, C., Freire, J.J., Román Cañizares, M., Zabala-Blanco, D.: Comparative analysis of cooperative routing protocols in cognitive radio networks. In: Botto-Tobar, M., Zambrano Vizuete, M., Torres-Carrión, P., Montes León, S., Pizarro Vásquez, G., Durakovic, B. (eds.) ICAT 2019. CCIS, vol. 1195, pp. 43–56. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-42531-9_4
4. Palacios, P., Castro, A., Azurdia-Meza, C., Estevez, C.: SVD detection analysis in cognitive mobile radio networks. In: 2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN), July 2017, pp. 222–224 (2017)
5. Kibria, M.G., Nguyen, K., Villardi, G.P., Zhao, O., Ishizu, K., Kojima, F.: Big data analytics, machine learning, and artificial intelligence in next-generation wireless networks. IEEE Access **6**, 32 328–32 338 (2018)
6. Awe, O.P., Deligiannis, A., Lambotharan, S.: Spatio-temporal spectrum sensing in cognitive radio networks using beamformer-aided SVM algorithms. IEEE Access **6**, 25 377–25 388 (2018)
7. Perez, J.S., Jayaweera, S.K., Lane, S.: Machine learning aided cognitive RAT selection for 5G heterogeneous networks. In: 2017 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), June 2017, pp. 1–5 (2017)
8. Patel, R., Kamboj, P.: Investigation of network simulation tools and comparison study: NS3 vs NS2. Transactions **14**, 15 (2015)
9. Shavlik, J.W., Dietterich, T., Dietterich, T.G.: Readings in Machine Learning. Morgan Kaufmann (1990)
10. Lee, J., Noh, H., Lim, J.: TDMA-based cooperative MAC protocol for multi-hop relaying networks. IEEE Commun. Lett. **18**(3), 435–438 (2014)
11. Shen, B., Su, X., Greiner, R., Musilek, P., Cheng, C.: Discriminative parameter learning of general Bayesian network classifiers. In: Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence, November 2003, pp. 296–305 (2003)
12. A. Ben-Hur and J. Weston, "A user's guide to support vector machines", in Data mining techniques for the life sciences. Springer, 2010, pp. 223–239
13. Dahan, H., Cohen, S., Rokach, L., Maimon, O.: Proactive Data Mining with Decision Trees. Springer, Heidelberg (2014)
14. Alfonso, U.M., Carla, M.V.: Modelado y simulación de eventos discretos. Editorial UNED (2013)