EAI Endorsed Transactions
on Wireless Spectrum
Research Article ICST.ORG

Spectrum Hole Identification in IEEE 802.22 WRAN using Unsupervised Learning

V.Balaji[1,*], S.Anand[1], C.R.Hota[1], G.Raghurama[2]

[1]Department of Computer science and Information Systems, BITS Pilani, Hyderabad Campus, 500078, India

[2]Department of Electrical and Electronics Engineering, BITS Pilani, Goa Campus, 403726, India

Abstract

In this paper we present a Cooperative Spectrum Sensing (CSS) algorithm for Cognitive Radios (CR) based on IEEE 802.22 Wireless Regional Area Network (WRAN) standard. The core objective is to improve cooperative sensing efficiency which specifies how fast a decision can be reached in each round of cooperation (iteration) to sense an appropriate number of channels/bands (i.e. 86 channels of 7MHz bandwidth as per IEEE 802.22) within a time constraint (channel sensing time). To meet this objective, we have developed CSS algorithm using unsupervised K-means clustering classification approach. The received energy level of each Secondary User (SU) is considered as the parameter for determining channel availability. The performance of proposed algorithm is quantified in terms of detection accuracy, training and classification delay time. Further, the detection accuracy of our proposed scheme meets the requirement of IEEE 802.22 WRAN with the target probability of falsealrm as 0.1. All the simulations are carried out using Matlab tool.

Received on 26 September 2015; accepted on 15 December 2015; published on 19 January 2016

Keywords: Cognitive radio, Dynamic Spectrum Access, Cooperative Sensing, TV white space, Machine Learning

doi:10.4108/eai.19-1-2016.150999

1. Introduction

A Cognitive Radio (CR) is a key technology [1] that allows wireless devices to dynamically access the available spectrum opportunities. Cognitive radio is a software defined radio [2] with the capability of identifying unused spectrum in a particular time, frequency and geographic location and utilizing it in opportunistic manner. The cognition capability of a CR is defined as the ability of CR transceiver to sense the surrounding radio environment, analyze the captured information and decide the best course of action in order to decide which spectrum bands are to be used and best transmission strategy to be adopted. CR is capable of making intelligent decisions and it's actions are based on observing the wireless connections and then using intelligent algorithms and computational learning to optimize their behavior. From the Definition of CR by Simon Haykin [3], it is clear that a CR device must have the attributes: awareness, intelligence, learning, adaptivity, reliability and efficiency. CR should intelligently sense the unused spectrum bands and learn without interfering with primary users. The experience gained through learning makes the CR to optimally reconfigure RF operating parameters and improve its decision. To perform this, CR must support the following functionalities [4]:

Spectrum awareness: It involves sensing the available spectrum bands and monitoring the activities of primary user with the help of spectrum sensing algorithms. These algorithms are used to identify the spectral activity pattern and estimate the characteristics of spectrum holes.

Learning: This phase acts as a knowledge base between spectrum sensing and decision phase. The gathered knowledge through learning can then be exploited to improve decision capability of CR.

Decisions and actions: The decision phase helps to choose appropriate spectrum band according to spectrum characteristics and user information. The actions are performed by effectively utilizing spectrum holes. The knowledge gathered during the learning phase acts as input to this module. The reconfiguration actions on RF operating parameters are performed

*Corresponding author.Email: p2011040@hyderabad.bits-pilani.ac.in

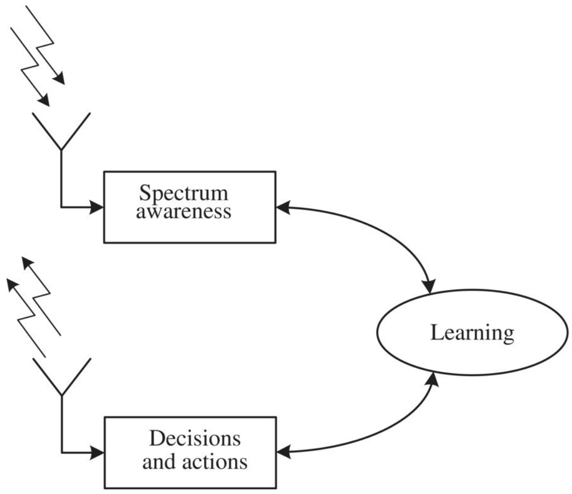

Figure 1. Functional model of CR

during this phase. The above sequence of operations by CR is schematically shown in Fig. 1.

1.1. Motivation

In CR, the most important step is to obtain necessary observations about its surrounding RF environment, such as the presence of primary users and the appearance of spectrum holes. Spectrum sensing enables the detection capability of CR to measure, learn and be aware of the radio's operating environment. The spectrum sensing problem in CR consists of three sub-problems [5]:

- Decide which channel to sense (Channel sensing is a decision making problem)
- Decide whether the channel is idle/busy based on local observations of the sensed channel (Primary signal detection or channel-state detection problem)
- Decide collaboratively whether to access the channel or not if it is indeed idle (Cooperative decision making problem)

Cooperative communication in wireless networks addresses the problem of channel impairments (i.e. multipath fading/shadowing) and improves the spatial diversity gain [6] of wireless receivers. The wireless nodes can make collaborative decision strategies to access channels with the help of cooperative communication techniques. The idea of CSS has been adopted from this cooperative communication technique. These CSS schemes greatly improve the received Signal-to-Noise ratio (SNR) under deep fading. Through this cooperation, the SUs can share their locally observed information about spectrum holes and make more accurate collaborative decision. It is noted that the cognitive capability [7] of CR enhances the decision quality of CSS algorithms and improves cooperative sensing accuracy. Recent advances in spectrum exploration and exploitation are discussed in [8].

1.2. Related work

Although cooperative communication [9] has lot of benefits to cognitive radios, there are still numerous theoretical and technical problems that remain unsolved. In [10], author's discuss various cooperative sensing techniques with their emphasis on spectrum sensing and access based cooperation, interference constraint based adaptive cooperative feedback, cooperative transmission based on rate-less network coding and interference coordination based on limited cooperation. Cooperative sensing is proposed in the literature [11] and its performance has been investigated extensively. In a widely studied form of cooperative spectrum sensing, the Secondary Users (SU) provide locally-sensed information on the primary users activity to a decision-making fusion center (FC) which can be an access point or base station or one of the SUs [12]. The FC analyzes the information and determines the activity status of primary user. The cooperative sensing can be categorized based on the type of fusion scheme used at the FC. Hard decision combining schemes such as AND, OR and k-out-of-N rule are considered in [13]. A cooperative sensing scheme based on linear combination of the local test statistics was proposed in [14] where the combining weights were optimized to improve the detection performance. Relay based CSS schemes are studied in [15].

As already discussed, cooperative learning [16] can help a cognitive radio to learn the surrounding environment and improve its sensing accuracy. However, in recent years there has been a growing interest in applying machine learning algorithms to CSS. In [17], the author has proposed CSS scheme based on supervised learning approach such as Support Vector Machine (SVM) and K-Nearest Neighbour (KNN) classification algorithms. The same author in another paper [18] has done a comparative study of Supervised (i.e. SVM and weighted KNN) and Unsupervised learning techniques (i.e. K-means clustering and Gaussian Mixture Model) for CSS schemes. The comparison of various CSS classifiers has been carried out based on training duration, classification delay and Receiver Operating Characteristics (ROC) performance. The result concludes that unsupervised K-means clustering is a promising approach for CSS due to its high ROC performance with low classification delay and training duration. However, in [18] the author assumed that the SUs are immobile and the SNR of each SU has been normalized to Gaussian distribution. In our work, we deployed SUs randomly with mobility in a grid topology and the SNR values are changed during iterations according to distance coordinates from the primary transmitter.

A survey on state-of-the art machine learning techniques and role of learning in cognitive radio is presented in [19]. In our recent work [20] we have used Perceptron learning module in which the fusion centre collects local sensing results of each SU and makes the final decision based on soft combination of the local decisions (weighted average method). The weights corresponding to each SU is computed using energy values captured by individual SU. The weight assigned to every secondary user is multiplied to the local decision value and the cumulative sum obtained from all the secondary users is used to determine the final decision of the FC. These weighted linear combinations of the local decision vectors produce the Target Output. Then, the hard-limit function determines the final decision of FC about availability of primary channel. Due to the dynamic channel environment, feature vectors are scattered in decision boundary which affects the detection accuracy of FC. To overcome this, we have developed in this work unsupervised K-means clustering approach which partitions set of training energy vectors into K disjoint clusters. This unsupervised K-means clustering is a promising approach due to its higher detection accuracy and less training and classification delays.

1.3. Contribution

This paper discusses a framework of CSS scheme using unsupervised K-means clustering algorithm to meet the functional requirement of IEEE 802.22 WRAN standard. The key contributions of this paper are as follows:

- The simulation scenario of CSS scheme has been formulated using machine learning techniques to meet the requirements of IEEE 802.22 WRAN standard.
- Local sensing phase is carried out using energy detection to scan the complete available channel set from 54MHz-682MHz with channel bandwidth of 7MHz.
- The Cooperative Spectrum Sensing (CSS) phase is based on unsupervised K-means clustering classification algorithm. The reason for adopting learning algorithm in CSS is because of its ability to dynamically adapt and train at any time, ability to 'learn' features and attributes of the system which is often difficult to formulate analytically. The performance of our proposed algorithms are evaluated using training duration, classification delay and detection accuracy.

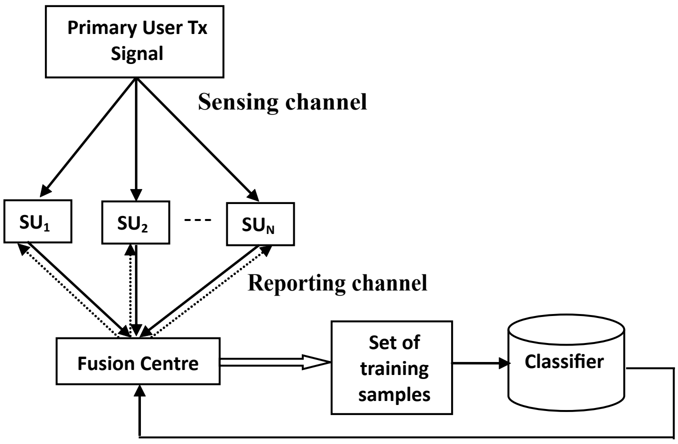

Figure 2. Working model of proposed CSS scheme

2. System model

2.1. Working model of the proposed CSS scheme

The Fig.2 shows the working model of the proposed CSS scheme. Each SU senses the primary user transmitted signal through sensing channel using energy detection scheme. The operation of energy detector is based on received signal power and noise power and comparing with local threshold to decide presence (H_1)/absence (H_0) of PU. The statistical inference drawn from energy detector which acts as local decisions X_i(n) of ith SU at sample index n is given by,

$$\begin{aligned} H_0 &: x_i(n) = w_i(n) \\ H_1 &: x_i(n) = h_i(n) \times s(n) + w_i(n) \end{aligned} \tag{1}$$

where w_i(n) is the additive white-Gaussian noise (AWGN), s(n) is the primary user signal and h_i(n) is the gain of the sensing channel between PU and SU.

The decision metric for the energy detector can be written as,

$$M_i = \sum_{n=0}^{N} | x_i(n) |^2 \tag{2}$$

where N is the observation vector. The performance of energy detector can be evaluated by using two probabilities: Probability of detection 'P_d' and Probability of false alarm 'P_f'. The probability of detection is to decide the presence of primary user when it is truly present. In contrary, the 'P_f' is to decide the presence of PU when it is actually not present. It can be formulated as,

$$\begin{aligned} P_d &= P_r(M_i > \lambda/H_1) \\ P_f &= P_r(M_i > \lambda/H_0) \end{aligned} \tag{3}$$

where 'λ' is decision threshold which can be selected for finding the optimum balance between 'P_d' and 'P_f'. By setting a desired probability of false alarm and calculating the variance of a data set, the system sets a threshold to indicate signals above the noise level.

Each SU processes the received energy and compares it with a local threshold. The received signal strength of each SU depends based on its distance from Primary transmitter. The collection of energy vectors of each SU is represented using a matrix shown below. In this matrix, the row vectors and column vectors are considered as secondary users and number of channels respectively. Each secondary user has an array of values specifying the availability of each of the 92 channels. The local decision observations of all SUs are denoted as Y_i(t) and represented in matrix form as,

$$Y_i(t) = \begin{pmatrix} x_1(n) & x_1(n) & \cdots & x_1(n) \\ x_2(n) & x_2(n) & \cdots & x_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ x_N(n) & x_n(n) & \cdots & x_N(n) \end{pmatrix} \quad (4)$$

Based on the local decisions of the N SUs, the fusion center will take a final decision. The local sensing phase is described in Algorithm 1. First, the primary user signal is added with noise according to the distance from the primary user. This noise added signal, 'signal_at_node' acts as input to different SUs. For each of the 10 secondary users, periodograms are calculated for 'signal_at_node', and based on that a Power Spectral Density (PSD) graph is obtained. The frequency range is considered as (54-698MHz) and divided into chunks of 7MHz channel bandwidth which is scanned in steps of channel width giving around 92 channels whose status can be either 'occupied' or 'available'. The average energy values at each channel are compared to a threshold value based on a random probability of false alarm. If the energy value of the channel is greater than the threshold, the channel is specified as 'occupied', otherwise it is 'available'.

2.2. Unsupervised learning algorithm for proposed CSS scheme

Learning ability is important in cognitive radios for effective decision making. Learning algorithms are implicitly built into spectrum knowledge acquisitions and decision-making algorithms in the sense that they convert information (current and past observations) in to decisions and actions. As mentioned in [3], a CR is an intelligent wireless communication system using the attributes of intelligence and cognitive abilities that enables self-learning and self-awareness.

Learning algorithms can broadly be categorized as either Supervised or Unsupervised learning. In the recent literature on CR [15], both supervised and unsupervised techniques have been proposed for various learning tasks. Unsupervised learning may particularly be suitable for diverse RF environment to make decisions and actions without prior knowledge. In this framework, we propose to use unsupervised

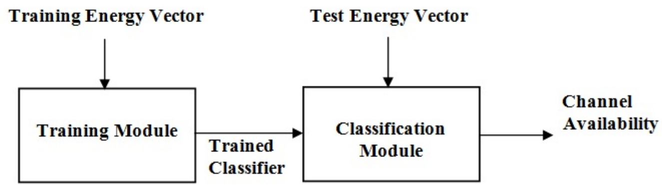

Figure 3. Schematic representation of learning module for proposed CSS scheme

K-means clustering algorithm to make cooperative decisions about channel availability. Before discussing the algorithm, it is necessary to look into the schematic representation of the learning module shown in Figure.3. It consists of training module and classification module. The training energy samples are fed into the training module which provides trained energy vectors to the classification module.

Generally, the training procedure of machine learning takes long time. To overcome this, the training module can be activated only during the initial CR deployment and any changes in primary network radio configurations. The classification module helps to determine the channel availability with the help of test energy vector. In order to achieve low classification delay, it is necessary to choose suitable classification algorithm with low complexity.

K-means clustering is an iterative, data partitioning algorithm that assigns number of observations to exactly one K clusters defined by centroids, where K is chosen before the algorithm starts. It partitions data into K mutually exclusive clusters, and returns the index of the cluster to which it has assigned each observation. It finds a partition in which objects within each cluster are as close to each other as possible, and as far from objects in other clusters as possible. Each cluster in the partition is defined by its member objects and by its Centroid. The centroid for each cluster is the point to which the sum of distances from all objects in that cluster is minimized. The Centroid of each cluster is used for classification. Once the classifier is trained, it is ready to receive test energy vectors for classification. K-means clustering aims to partition the observed energy vectors into K clusters (c1, c2,..ck) so as to minimize the distance of vectors within cluster by using distance measure. The partitioned clusters are passed using 'argmin' function as mentioned in equation (5).

$$\underset{c_1,c_2,\ldots..c_k}{argmin} \sum_{k=1}^{K} \sum_{Y^L \epsilon C_k} \left\| Y^L - \alpha_k \right\|^2 \quad (5)$$

where Ck is the set of training energy vectors that belong to cluster K, Y^L is complete training

Algorithm 1: Local Sensing Based on Energy Detection

```
No_of_Nodes N;
Data: energyDetection()
Result: energy vector
begin
    for user ← 1 to N do
        Signal_at_node ← Primary_user_Signal + AWGN
        L ← size(Primary_User_Signal)
        Threshold ← qfuuncinv(Pf(user))/sqrt(L)+1
        Periodogram_at_node ← periodogram(Signal_at_node)
        Occupied[length(Periodgram_at_node )] ← 0
        while i < lengthPeriodgram_at_node do
            if Periodgram_at_node i > Threshold then
                occupied(i) ← 1
            i = i +1
        Channel_width ← 7 MHz
    Energy ← 0
    Sum ← 0
    if (occupied == 1) then
        Sum ← Sum + 1
        Energy ← Energy + Periodgram_at_node (freq)
    if Sum > width/2 then
        Channel ← 1
    else
        Channel ← 0
Data: changeVelocities(velocity_i, velocity_j, X)
begin
    if (mod(x, 4) == 0) then
        Reverse the Velocity_i
    if (mod(x, 2) == 0) then
        Reverse theVelocity_i
    else if (mod(x, 4) == 1) then
        Reverse the Velocity_j
        if (mod(x, 2) == 0) then
            Reverse the Velocity_j
Data: changeDistances(velocity_i, velocity_j, X, Y)
begin
    X ← X + Velocity_i
    Y ← Y + Velocity_j
    if (X, Y) > (100, 100) then
        (X, Y) ← (X, Y) − 2 × (Velocity_i, Velocity_j)
    if (X, Y) > (0, 0) then
        (X, Y) ← (X, Y) + 2 × (Velocity_i, Velocity_j)
```

energy vectors, α_k is called Centroid of cluster K and $||.||^2$ is known as Square of Euclidean distance. After training, the classifier receives test energy vector for classification. The classifier classifies based on the following condition,

$$\frac{||Y^* - \alpha_1||}{min_{k=1,2,...K} ||Y^* - \alpha_k||} \geq \beta \tag{6}$$

where Y^* is known as test energy vector received by classifier, α_k is the Centroid for cluster K and β

Algorithm 2: Proposed CSS scheme using k-means clustering algorithm

Input energy(i,j) //Stores energy values of j^th SU for the i^th band;
Initialize local decision(j,i);
$Y^L \leftarrow$ Training energy vectors;
$Y^{L,k} \leftarrow C_k$ //partitions training vectors into K disjoint clusters (C);
$\alpha_k \leftarrow \mu_i$ //Initialize centre of cluster to determine Centroid α_k, where $i = 1, 2, .., k$;

for *each cluster k* **do**
  $Y^{L,K} \leftarrow |\alpha_k|^- 1 \sum_{Y^L} \epsilon \alpha_k Y^L, \forall_k = 1, 2, .., k$
  //calculating mean of all training energy vectors in cluster k
  $Distmeasure \leftarrow$ Euclidean || Cityblock
  // for minimizing distance of energy vectors to local minima
$CH \leftarrow H_0|H_1$
// each SU reports its sensing decision to FC
$CH \rightarrow$ global decision
// FC declares final decision based on suboptimal solution through convergence

is called threshold to control trade-off between false alarm and detection probabilities. The algorithm works as follows. First, it Partitions the set of energy data into k disjoint clusters. The Centroid of first cluster (for which the class is available) is the mean of the data for which class is available. All the other data is divided into separate K clusters such that within squares sum of distances is minimized for all these K clusters. For the given training energy vectors, the data is first divided into two parts. One is for those for which the class is available, and the other for those for which class is unavailable. All the other data is divided into K clusters, where K varies from 1 to 10. For classification of test energy vectors, the classifier determines if the test energy vector belongs to cluster 1 or other clusters, based on the distance of the test energy vector to the centroids. We have considered two distance measures namely Euclidean and Cityblock. The Euclidean distance examines the root of square differences between coordinates of a pair of objects. Similarly, the cityblock distance examines the absolute differences between coordinates of a pair of objects. The classifier classifies the test vector as channel unavailable if the distance d is greater than β which is a tunable parameter. The value of this tuning parameter varies from 0.1 to 0.3 which indicates the permissible value of 'P_f' as per IEEE 802.22. The steps involved in unsupervised K-means clustering based CSS scheme is shown in Algorithm 2.

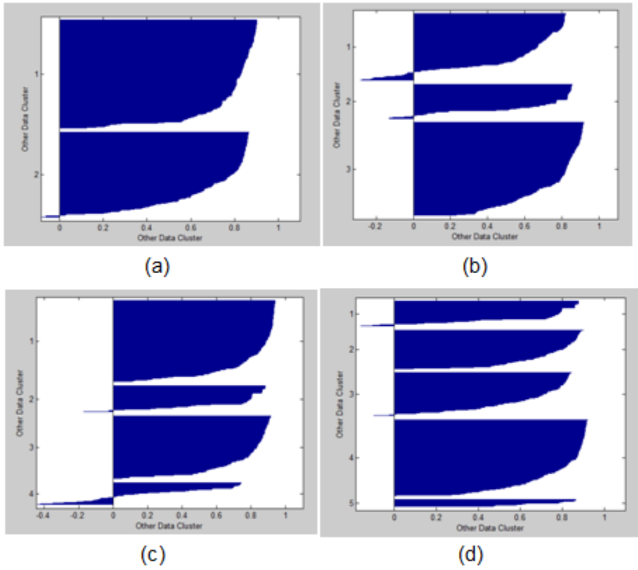

Figure 4. Silhouette plot for Euclidean distance measure: a)K=2 b)K=3 c)K=4 d)K=5

3. Simulation Setup and Results

The performance of the unsupervised K-means clustering algorithm for CSS has been analyzed by calculating delay of training as well as testing energy vectors and detection accuracy. We consider a CR simulation scenario with one primary transmitter and 10 SUs which operate in the frequency range of (54-698)MHz divided into 7MHz of channel bandwidth. Multiple secondary users are randomly deployed in a grid topology of area 120×120 Sq.km, using one FC. The distance coordinates of each SU varies during each iteration. The value of SNR for each SU changes based on the distance from the primary transmitter. The other important simulation parameters are as follows: Primary transmitter power is 200 MW, Primary signal type is BPSK modulated signal, Noise model is Additive White Gaussian Noise (AWGN). The simulation scenarios are performed using MATLAB 7.14 (R2012a) in a 64-bit computer with core i3 processor, clock speed 2.4 GHz, and 4GB RAM.

The performance of unsupervised K-means clustering algorithm for CSS scheme has been summarized on Table.1 and 2 using Euclidean and Cityblock distance metrics. The following observations can be made from the above summary. The variation of training delay with respect to number of clusters is less under Cityblock than Euclidean. There is less deviation on delay time for test energy vectors under both distance metrics. It is important to note that the detection accuracy remains same under both distance metrics. Also, the rate of detection accuracy satisfies the permissible limit given by IEEE 802.22.

Table 1. Performance Summary of k-means clustering based CSS scheme using Euclidean Distance metric

No of clusters	Training observation	Training Delay	Test observation	Test delay	Detection accuracy
2	1634	0.0097	86	0.0289	69.76
3	1634	0.0117	86	0.0470	69.76
4	1634	0.0157	86	0.0543	69.76
5	1634	0.0168	86	0.0750	69.76

Table 2. Performance Summary of k-means clustering based CSS scheme using Cityblock Distance metric

No of clusters	Training observation	Training Delay	Test observation	Test delay	Detection accuracy
2	1634	0.0116	86	0.0369	60
3	1634	0.0119	86	0.0506	60
4	1634	0.0121	86	0.0661	60
5	1634	0.0129	86	0.0757	60

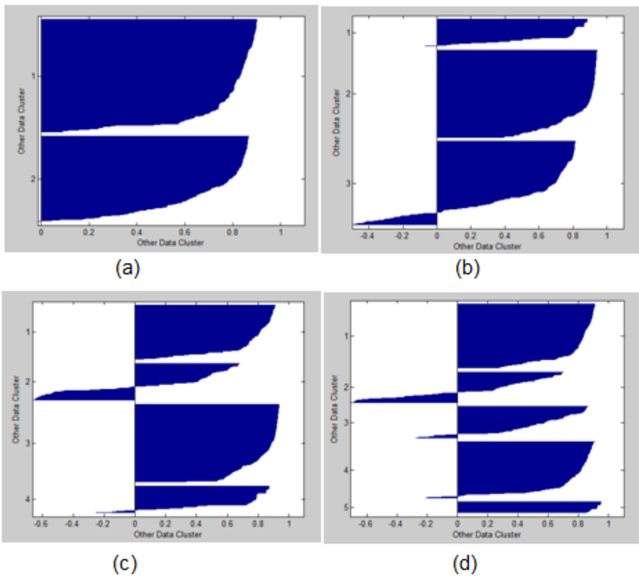

Figure 5. Silhouette plot for Cityblock distance measure: a)K=2 b)K=3 c)K=4 d)K=5

To get an idea of how well-separated the resulting clusters are, we can make a silhouette plot using the cluster indices output from K-means. The silhouette plot displays a measure of how close each point in one cluster is to points in the neighboring clusters. This measure ranges from +1, indicating points that are very distant from neighboring clusters called 'well-formed clusters', through 0, indicating points that are not distinctly in one cluster or another called 'ill-formed clusters', to -1, indicating points that are probably assigned to the wrong cluster called 'outliers'. Silhouette returns these values in its first output. The Silhouette plots using Euclidean distance metric for different values of K are shown in Fig.4.

The Silhouette graph shows two well-formed clusters, with a little fraction of data points as outliers. Since, most of the data points from both the clusters have their index greater than 0.6, this shows that the data points are tightly bound in the two clusters. With K = 3, clusters are formed from the data for which the channel is unavailable; the data points are scattered into 3 clusters as shown above. The Silhouette Graph show that the cluster 3 formed from above data is a well-formed cluster. Cluster 2 and Cluster 3 can also be thought of as well-formed, however the number of data points that are classified as outliers are more in these cases. With K = 4 clusters are formed from the data for which the channel is unavailable, the data points are scattered into 4 clusters as shown above. The Silhouette Graph shows that the cluster 4 formed from above data are an ill-formed data cluster. Cluster 1, 2 and 3 are well formed data clusters; however, cluster 2 has some ill classified points. With K = 5 clusters are formed from the data for which the channel is unavailable, the data points are scattered into 4 clusters as shown above. The Silhouette Graph shows that all the clusters formed with K = 5 are well-formed data clusters. Cluster 1 and 3 have some outliers classified under them, but all the other clusters have the index of most of the data-points above 0.6, which makes all of them distinct.

Similarly, the Silhouette plots for Cityblock distance measure are shown in Fig.5 for various cluster values. With K = 2 clusters are formed from the data for which the channel is unavailable, the data points are scattered into two clusters as shown above. The Silhouette Graph shows that both clusters are very well-formed with no outliers classified. Also, majority of data points in each of the clusters have their index greater than 0.6, which shows that the clusters have been formed tightly by the data points. With K = 3 clusters are formed from the data for which the channel is unavailable, the data points are scattered into 3 clusters as shown above. The Silhouette Graph shows that cluster 1 and 2 are well-formed, with cluster 1 having some outlier data points. However, cluster 3 has a large number of outliers, hence cannot be classified as well-formed. With K = 4 clusters are formed from the data for which the channel is unavailable, the data points are scattered into 4 clusters as shown above.

The Silhouette Graph shows that only cluster 1 and cluster 3 can be thought as well-formed. However, cluster 2 has large number of data points specified as outliers. Also, cluster 4 is not well-formed because of the outliers shown in the Fig.5. With K = 5 clusters are formed from the data for which the channel is unavailable, the data points are scattered into 5 clusters as shown above. The Silhouette Graph shows that only cluster 1 and cluster 5 are well-formed. However, cluster 5 is small as it contains less number of data points. All other clusters have outliers classified within them and hence cannot be thought of as well-formed.

4. Conclusion

In this paper, we presented cooperative spectrum sensing (CSS) scheme using unsupervised k-means clustering algorithm. The proposed CSS scheme has the capability to learn from the radio environment to achieve cognitive tasks. The received signal strength of each SU is measured using energy detection scheme and considered as feature input to the classifier module to determine channel availability. The simulation scenario has been formulated to meet the requirements of IEEE 802.22 WRAN standard. The simulation results show that the unsupervised k-means clustering algorithm significantly improves detection accuracy with training and testing delay of 16.8 and 75 milliseconds respectively. As future work, it can be extended further to various cooperation scenarios to support different wireless standards and specifications which will help to improve the cognition capability and cooperative sensing accuracy.

References

[1] Marcus M, Burtle J, Franca B, Lahjouji A, McNeil N. Federal communications commission spectrum policy task force. Report of the Unlicensed Devices and Experimental Licenses Working Group. 2002;.

[2] Mitola III J, Maguire Jr GQ. Cognitive radio: making software radios more personal. Personal Communications, IEEE. 1999;6(4):13–18.

[3] Haykin S. Cognitive radio: brain-empowered wireless communications. Selected Areas in Communications, IEEE Journal on. 2005;23(2):201–220.

[4] Akyildiz IF, Lee WY, Vuran MC, Mohanty S. NeXt generation/dynamic spectrum access/cognitive radio wireless networks: a survey. Computer Networks. 2006;50(13):2127–2159.

[5] Liang YC, Chen KC, Li GY, Mahonen P. Cognitive radio networking and communications: An overview. Vehicular Technology, IEEE Transactions on. 2011;60(7):3386–3407.

[6] Yücek T, Arslan H. A survey of spectrum sensing algorithms for cognitive radio applications. Communications Surveys & Tutorials, IEEE. 2009;11(1):116–130.

[7] Wang B, Liu K. Advances in cognitive radio networks: A survey. Selected Topics in Signal Processing, IEEE Journal of. 2011;5(1):5–23.

[8] Lunden J, Koivunen V, Poor HV. Spectrum Exploration and Exploitation for Cognitive Radio: Recent Advances. Signal Processing Magazine, IEEE. 2015;32(3):123–140.

[9] Akyildiz IF, Lo BF, Balakrishnan R. Cooperative spectrum sensing in cognitive radio networks: A survey. Physical communication. 2011;4(1):40–62.

[10] Chen X, Chen HH, Meng W. Cooperative communications for cognitive radio networksâĂŤfrom theory to applications. Communications Surveys & Tutorials, IEEE. 2014;16(3):1180–1192.

[11] Ghasemi A, Sousa ES. Collaborative spectrum sensing for opportunistic access in fading environments. In: New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005. 2005 First IEEE International Symposium on. IEEE; 2005. p. 131–136.

[12] Lee WY, Akyildiz IF. Optimal spectrum sensing framework for cognitive radio networks. Wireless Communications, IEEE Transactions on. 2008;7(10):3845–3857.

[13] Zhang W, Mallik RK, Letaief K. Optimization of cooperative spectrum sensing with energy detection in cognitive radio networks. Wireless Communications, IEEE Transactions on. 2009;8(12):5761–5766.

[14] Peh EC, Liang YC, Guan YL, Zeng Y. Cooperative spectrum sensing in cognitive radio networks with weighted decision fusion schemes. Wireless Communications, IEEE Transactions on. 2010;9(12):3838–3847.

[15] Ganesan G, Li Y. Cooperative spectrum sensing in cognitive radio, part I: Two user networks. Wireless Communications, IEEE Transactions on. 2007;6(6):2204–2213.

[16] Bkassiny M, Li Y, Jayaweera SK. A survey on machine-learning techniques in cognitive radios. Communications Surveys & Tutorials, IEEE. 2013;15(3):1136–1159.

[17] Thilina KM, Choi KW, Saquib N, Hossain E. Pattern classification techniques for cooperative spectrum sensing in cognitive radio networks: SVM and W-KNN approaches. In: Global Communications Conference (GLOBECOM), 2012 IEEE. IEEE; 2012. p. 1260–1265.

[18] Thilina KM, Choi KW, Saquib N, Hossain E. Machine learning techniques for cooperative spectrum sensing in cognitive radio networks. Selected Areas in Communications, IEEE Journal on. 2013;31(11):2209–2221.

[19] Abbas N, Nasser Y, El Ahmad K. Recent advances on artificial intelligence and learning techniques in cognitive radio networks. EURASIP Journal on Wireless Communications and Networking. 2015;2015(1):1–20.

[20] Balaji V, Kabra P, Saieesh P, Hota C, Raghurama G. Cooperative Spectrum Sensing in Cognitive Radios Using Perceptron Learning for IEEE 802.22 WRAN. Procedia Computer Science. 2015;54:14–23.