Sequential Learning for Multi-Channel Wireless Network Monitoring With Channel Switching Costs For Non-stationary Users

Binayak Ranjan Das, Aman Srivastava

https://github.com/binayakranjan/Sniffer-Channel-Assignment-In-Wireless

May 17, 2020

Accurate and timely estimates of wireless network conditions and performance characteristics can yield better performance in a number of applications such as network resource management,wireless advisory,troubleshooting etc.However,utilization of multiple contiguous or non-contiguous channels or bands for estimation gives rise to the Sniffer-channel assignment problem.Sniffer-channel assignment with no prior knowledge of user activity is closely related to the multi-armed bandit problem (MAB) which can be optimized by Sequential Learning. While prior works have explored ways to use variants of the Upper Confidence Bound(UCB) algorithm for this purpose, they have assumed the users are static.We analyze two algorithms: the discounted UCB and the sliding-window UCB to cater to the wireless network with non-stationary users i.e mobile users.

1 Introduction

Today's wireless usage is characterized by a diverse set of QoS requirements which range from best-effort data services, to VOIP and streaming applications. The QoS sensitive services have additional constraints which make it even more difficult and challenging to manage the wireless infrastructure. Accurate and timely estimates of wireless network conditions and performance characteristics are critical to many system administrative tasks including fault diagnosis, resource management, and critical path analysis for infrastructure upgrades. *Passive monitoring* is a widely-used and effective technique to monitor wireless networks where a dedicated set of hardware devices called sniffers, or monitors, are used to monitor activities in wireless networks. In this, sniffers (i.e., software or hardware devices that intercept and log packets) are used to capture transmissions of wireless devices or activities of interference sources in their vicinity and analyze traffic between other nodes and store the information in trace files which can be used to estimate network conditions and performance. Such estimates are utilized for efficient network operations including network resource management, network configuration, fault detection or diagnosis, and network intrusion detection.Often the logs help in maximizing the QoM in multichannel infrastructure wireless networks with different apriori knowledge obtained from sniffers. Hence, the requiremt and utilization of *sniffer channel assignment* is prolific. Let's focus on some of it's use case scenarios.

Cognitive radio network (CRN) is a promising paradigm to solve the contradiction between the limited wireless spectrum resources and the growing number of mobile applications. By exploiting the spectrum in an opportunistic fashion, CRNs allow the secondary users (SUs) to use the licensed bands without generating excessive interference to primary users (PUs). In

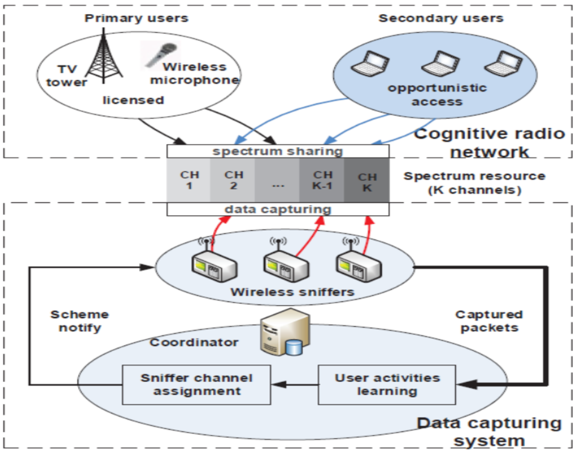

Figure 1: *Sniffer channel assignment in CRN by Jing et.al.[6] with an objective is to capture as much SU traffic as possible , so as to support applications like network monitoring and network forensics*

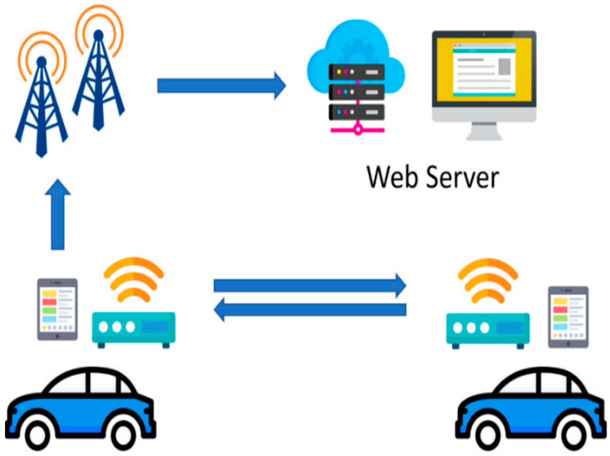

Figure 2: *Sniffer channel assignment in Ad Hoc Network by Zhang et.al[7].The experiment was carried out by two midsize vehicles, and each roof mounts a dedicated short-range communications (DSRC) on-board units (OBUs) at an approximate height of 1.48 meters. The GPS antennas had to be mounted outside of the vehicles in all weather conditions; otherwise, the GPS position would stop updating. An additional micro-controller (Raspberry Pi) was connected with each of the DSRC OBUs via a RS-232 serial interface for data exchange*

light of the volatile characteristics of SU activities,the sniffer channel assignment problem can be formulated as a nonstochastic/ adversarial MAB problem, without probabilistic models of the SU activities. Imperfect monitoring and channel switching costs, often non-negligible in realistic data capture applications can be incorporated to cater the time-varying unreliability in monitoring.

Wireless mesh networks (WSNs) are being used on the last mile for extending or enhancing Internet connectivity for mobile clients located on the edges of the wired network. A WSN is a communications network made up of radio nodes organized in a mesh topology.One major problem in them is capacity reduction due to interference among multiple simultaneous transmissions. Heuristic approaches on channel assignments are proposed to improve the aggregate throughput of WMNs. The aggregate traffic demands and network topology do not change frequently in WMNs. Hence, it is feasible to have optimizations using measured traffic demands. Optimal sniffer channel assignment can be used to compute the optimal channel assignment and routing and configure each elements periodically. One can use it to find the necessary and sufficient conditions under which interference-free link communication can take place. Using this, effective algorithms can be designed which can help us to exploit the increased number of channels and radios.

The capacity of *ad hoc wireless networks* can be significantly increased by equipping each network node with multiple radio interfaces that operate on multiple non-overlapping channels. The entire frequency band is divided into multiple channels and each radio can access only one channel at a time. Therefore, multiple radio interfaces will help a network node utilize a larger amount of radio bandwidth, which in turn will increase the system capacity and throughput. Therefore, optimal sinffer channel assignment can be used to solve problems like determining the set of channels that each link should operate on, when should each link be activated at each channel and how to select paths that minimize interference and increase throughput.

2 Related Work

[1] Arora etc used the linear bandit model to capture the dependency amongst the arms and develop two policies that take advantage of this dependency and enjoy logarithmic regret bound of time-slots with a term that is sub-linear in the number of arms.

[2] Garivier and others analyzed two algorithms the discounted UCB and the sliding-window UCB for Non-Stationary Bandit Problems.

[3] Le et.al proposed a novel Upper Confident Bound-based (UCB) policy that enjoys a logarithmic regret bound in time t and depends sublinearly on the number of arms, while its total switching cost grows in the order of O(loglog(t)).

[4] D. Shin, S. Bagchi and C. Wang, "Toward optimal sniffer-channel assignment for reliable monitoring in multi-channel wireless networks," 2013 IEEE International Conference on Sensing, Communications and Networking (SECON), New Orleans, LA, 2013, pp. 203-211.

[5] Yevgeny Seldin, Csaba Szepesvári, Peter Auer and Yasin Abbasi-Yadkori present an empirical evaluation and improved analysis of the performance of the EXP3 algorithm in the stochastic setting and

its modification capable of achieving "logarithmic" regret in stochastic environments.

[6] Jing and others propose two online learning algorithms for the SCA scenarios with and without channel switching costs, respectively, and their regret performances are proven uniformly sublinear in time and polynomial in the number of channels.

[7] Zhang et.al presents a real-time channel prediction model, which could predict channel parameters such as path loss (PL) and packet drop (PD), for dedicated short-range communications (DSRC).

3 Overview

This work focuses on the problem to determine sniffer-channel assignment with switching costs as a linear partial monitoring problem. This is a super-class of multi-armed bandits(MAB). A partial monitoring problem is one in which an action is chosen repeatedly and the environment responds with an outcome. The learner suffers a loss and receives a feedback signal, both of which are fixed functions of the action and the outcome. The goal of the learner is to minimize his regret which is the difference between his total cumulative loss and the total loss of the best fixed action in hindsight. The problem can arise in two types of environments - stationary and non - stationary. As opposed to stationary environments the user activities change over time in non - stationary environments, for example, due to the mobility of the user and the consideration of adversarial settings where an adversary has complete control over the payoffs which makes it more challenging.

4 Approach Details

4.1 For Stationary User Environments

We suppose that there are 'p' sniffers monitoring user activities in 'K' channels. A user 'u' is operating in one channel whose transmission probability is denoted by 'P(u)'. The relation between users and sniffers can be represented by an undirected bi-partite graph having 2 sets - one for users 'u' and one for sniffers 's'. There exists an edge between a sniffer and user if the user is in the reception range of the sniffer. Our problem is based on the assumption that one sniffer can observe one user at a time because a sniffer can only observe transmissions over a single channel at one time. Let κ denote the set of possible channel assignments of sniffers to channels. Firstly, we consider the baseline model for optimal channel assignment without uncertainty whose primary objective is to maximize the expected number of 'active' users monitored known as 'Max-effort-cover(MEC)'. Each sniffer can monitor one of a set of 'k' channels. Each sniffer is associated with a set of binary decision variables z_{s_k}, where $z_{s_k} = 1$ if the sniffer is assigned to channel k; 0, otherwise. Further, y_u is a binary variable indicating whether user u is monitored, and p_u is the weight associated with user u. The MEC problem is NP-hard with respect to the number of sniffers, even for K = 2. When the graph G and the user activity probabilities are given, the optimal sniffer channel assignment is stationary in time. However, when the user activity probabilities are unknown, a learning strategy must try different assignments just like a partial monitoring problem. The second model is Linear bandit for optimal channel assignment with uncertainty. Our assumption is consistent with FDMA and TDMA that one user can be observed by only one sniffer at a time. It is observed that the expected payoff is maximized by putting different sniffers to different channel. Thus, the joint payoff is the number of distinct users observed over all channels, k = $(k_1, k_2, ..., k_p)$. A reasonable way to estimate the parameter vector θ is to keep a running average for the components of θ . If at time t agent chooses k(t) ϵ κ then the current estimate, θ (t-1), can be updated based on past value. Moreover, sniffers are deployed distributedly. Their observations are typically correlated but non-identical. The learning efficiency or performance of a policy is evaluated in terms of its regret. It is composed of sampling regret (due to the play of suboptimal arms) and switching regret capturing the cost of switching assignments. A good expected policy will be one which minimizes both. Some arms reveal information about other arms, hence, we identify a restricted set smaller than κ called **spanning set or spanner** denoted as ε by assigning all the sniffers to the same channel so that arms played in this set help us reach the optimal answer. For optimal sniffer channel assignment, we are making use of Upper confidence bound(UCB) based policy which states that when switching costs are not negligible, there should be less changing of joint action. An action once selected persists for a period of time (epoch). This time is divided into blocks which are small at first when the uncertainty is high and the length increases once the system gains more knowledge. The choice of arm to be played depends on - estimated payoff and number of times a component has been observed. The epoch length grows exponentially and at the end of each epoch the likeliness of choosing sub-optimal arms decreases.

4.2 For Non - Stationary User Environments

These observations above can be extended to non-stationary environments where the user activities change with time, for example, a user may be mobile, a user may switch to an empty channel in a multi channel system. The decision to play the most optimal arm is comparatively more challenging due to the changing environment. The main reason behind is the changes in reward distribution with time. The probability of the availability of a given channel is likely to change with time. Here, we will observe the non-stationary environment cases where the reward distribution has abrupt changes with time. We will observe the rate optimality of two algorithms - **discounted UCB** and **sliding window UCB** and derive a lower bound for their regret. Regret is defined as the difference between the expected pay-off gained by an ideal policy which has now switching cost due to optimal stationary sniffer-channel assignment and that obtained by the given policy. It is used to measure the performance of a policy. In strategies concerning non-stationary environments, we pick the best arm at each step as opposed to stationary environments where we consider best arm on average. The number of arms represent the product of number of sniffers times number of channels in a sniffer-channel assignment problem. Going forward, we assume that the distributions of rewards remain constant during periods and change abruptly at some instants which we call breakpoints.

The discounted UCB policy calculates the average of the past rewards with a discount factor of γ, giving more weight to recent observations compared to the past. Let γ_T denote the number of breakpoints before time 'T' and $N_T(i)$ denote number of times arm 'i' was played when it was not the best arm during the T first rounds. Then, the minimum of the difference of expected reward of the best arm, $\mu t(*)$ and that of the sub-optimal arm i for all time period 1 to T, μt(i). $P\gamma$ and $E\gamma$ signify the probability distribution and expectation under policy D-UCB with discount factor γ. The expected reward estimate might be poor following a breakpoint due to which the D-UCB policy may play sub-optimal arms. This might be happen if $\mu t(i)$ is substantially over- estimated, if $\mu t(*)$ is substantially under-estimated, or if $\mu t(i)$ and $\mu t(*)$ are close from each other.

In sliding window UCB, at any time instant 't' instead of averaging the rewards over the complete past, we calculate the average over last τ plays with a with a discount factor of γ, same as above. The sliding window is of size τ. If the time period T and the growth rate of the number of breakpoints γ_T are known in advance, the window size τ can be chosen so as to minimize the regret. SW-UCB just needs to store the last τ actions and rewards at any time instant 'T' in order to calculate $N_T(i)$ and $X_T(i)$. These algorithms are rate optimal in a minimax sense.

5 Implementation and Evaluation

5.1 For Stationary User Environments

The components we have are wireless AP's, users and sniffers. The 2-D plane is divided into hexagon cells of radius 86 m. There is an AP at the centre of each cell.

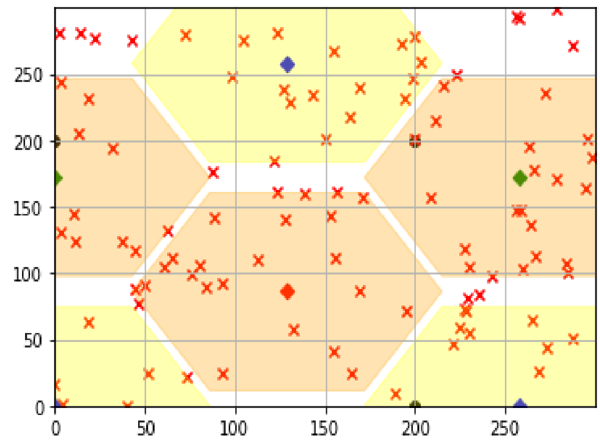

Figure 3: *Hexagonal layout with users (red '+'), sniffers (black solid dots), access points (APs) (at the center of each hexagonal cell), and channels of each cell (in different colors).*

Each cell is associated with a base station operating in a channel. There are methods to find if the given user is in the hexagon or not which tells us whether the user falls in the range of the given AP. Sniffers are present at a distance of 200 m from each other and have a range of 200 m. The sniffers are assigned Channel 0 and no user(User -1) by default. We can assign channels and users to the sniffer. We can also check if the given users lie in the sniffer's range or not. The users are placed randomly in the 2-D plane. They are assigned Channel 0, are inactive and have an activity probability of 0.3 by default. We can check the AP and sniffers in range for a user. We can also change its status from inactive to active.

We iterate over all users assigning them random coordinates. We find the AP in range of the given user and assign the channel of the given AP to the user. We change the status of the user from inactive to active based on the activity probability. Now, we separate the x and y coordinate of the sniffers, x and y coordinate of the APs based on their channels and x and y coordinate of the users and plot them to see a visualization. For each channel, we assign the channels to the sniffer and make a list of channel assignment and user assignment for the sniffers. Following this, we generate a restricted set of arms called spanner by generating phi and using the user assignment and sniffer channel assignment. We then define an upper confidence bound which takes into account the actions, their rewards and average rewards which help us in forming a learning policy (theta). It helps us in updating theta based on time, number of trials, arm actions(k) and their results. The learning policy, theta, the spanner arms and their actions help us in obtaining the regret as explained above. In each epoch, the sniffers are assigned to the user channel, the agent performs some action and regret is calculated. The UCB tries to minimize the regret to gain an optimal policy.

5.2 For Non-Stationary User Environments

In this case, we introduce the mobility of the users or time-variant channel conditions. This can be formulated as dynamic reward. Let there be *lower* and *amplitude or higher* bound for every reward.Every time the reward is normalized to avoid higher values shadowing lower reward values.

For the D-UCB and SW-UCB algorithm simulation, we consider 2 configurations. In the first configuration, we have K = 3 arms and the time horizon T = 10^4. The rewards of arm i from 1,2,....,K at time 't' are independent Bernoulli random variables with success probability denoted by $p_t(i)$.

For D-UCB α is 1 and γ is 0.99. For SW-UCB, Size of the sliding window τ is 1000 and α is 1.

6 Performance Comparision

6.1 For Stationary User Environments

Fig. 5 and 6 show the Switching regrets and Total regrets of the modified method and the naive UCB method over time. It can be also be seen in Fig 4 that in the two scenarios, the proposed method achieves similar sampling regrets even though the new method explores less often due to the use of epochs.When comparing the switching regrets and total regrets, the proposed method clearly outperforms the previous method. The differences in switching regrets are more pronounced when the number of sniffers increases from 3 to 6. Recalling that the number of assignments grows exponentially with the number of sniffers,we can explain this by noting that more switching is likely to occur when the number of possible assignments is larger, especially in the initial phase of learning. Furthermore, we observe the growth of the switching regrets in the proposed algorithm is much slower.E arlier it was log(t), now it is log(log(t)). In all scenarios, the two algorithms have comparable sampling regrets. This is expected because they use the same formula to find the best arm to play whenever the system needs to make a new decision. However, using epochs, the proposed method incurs much smaller switching costs, thus incurring lower total regret.

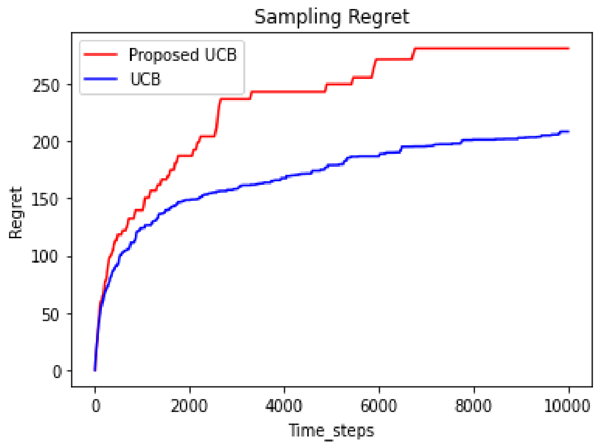

Figure 4: *sampling regret*

Figure 5: *switching regret*

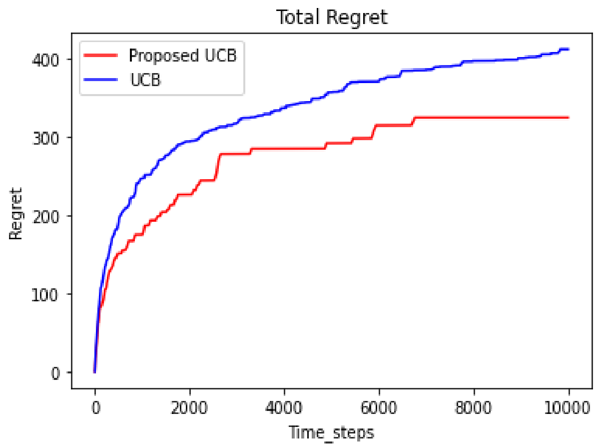

Figure 6: *Total regret*

6.2 For Non-Stationary User Environments

We can see from figure 7 to figure 10 that D-UCB and SW-UCB have a comparable performance working equally well in non-stationary user environments. They work significantly well compared to the EXP3.S and UCB-1 stationary environment algorithms by wasting less time and quickly figuring out the optimal arm

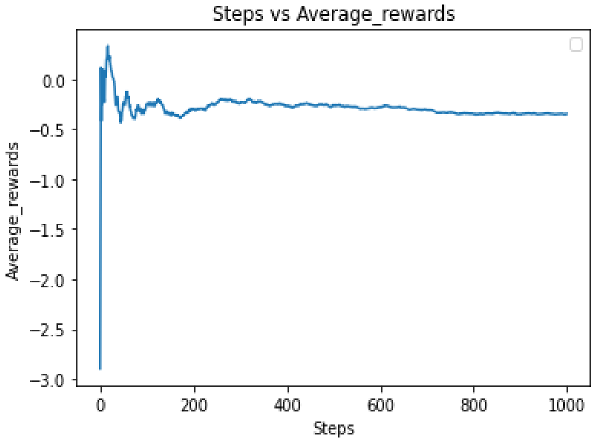

Figure 7: *Steps vs Average Rewards for D-UCB*

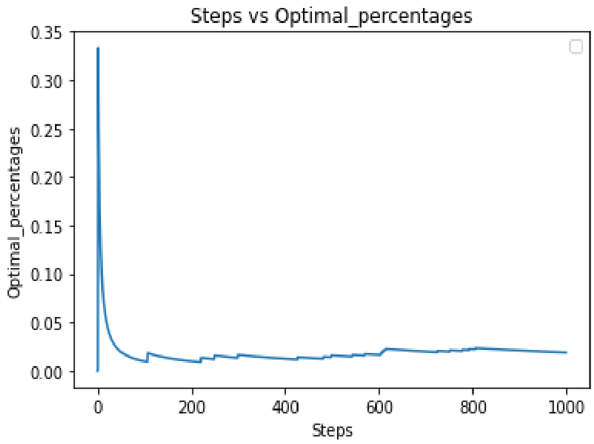

Figure 8: *Steps vs Optimal Percentages for D-UCB*

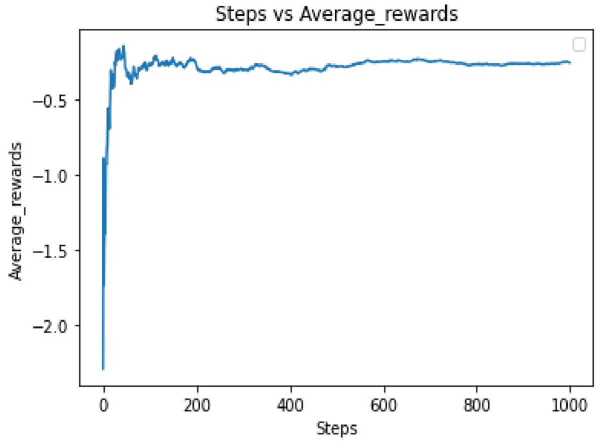

Figure 9: *Steps vs Average Rewards for SW-UCB*

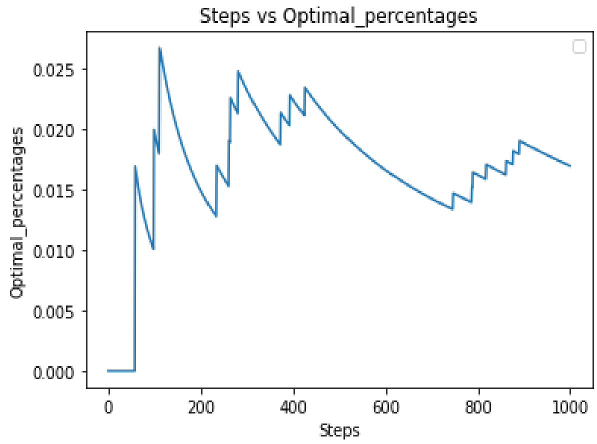

Figure 10: *Steps vs Optimal Percentages for SW-UCB*

and working on it. In the first configuration, we have K = 2 arms and the time horizon T = 10^4. The rewards of arm i from 1,2,....,K at time 't' are still independent Bernoulli random variables with success probability denoted by $p_t(i)$ but they are in continuous, persistent evolution. We observe that the performance of D-UCB and SW-UCB are almost equivalent while naive UCB accumulates larger regret. We can also see that in Figure 8 the optimal percentage shows a spike and then goes down with increasing steps (or time). This may be due to the fact that the expected reward estimate might be poor following a breakpoint due to which the D-UCB policy may play sub-optimal arms. In addition to this, we observe that the performance of these policies also depend on a few important parameters like the discount factor γ for D-UCB and the window size τ for SW-UCB. Figure 10 depicts that the algorithm works well in the chosen window and when there is a change, it suffers.

7 Conclusion

In Summary, timely estimates of network conditions and performance characteristics can yield to better performance in a number of applications. While the knowledge of network coverage can help wireless service providers and network administrators can help make better decisions about dimensioning and allocation of network resources, the cross-layer information and the operational network logs can help network administrators identify the root causes of problems like service outage, malicious behavior and intrusion. Last but not the least, information about individual devices and their operational parameters ,e.g., channels, subcarriers, hopping sequences, transmission power levels, etc can be retrieved. The optimal sniffer channel assignment helps to determine the activity of users both stationary and non-stationary , which is initially unknown to the sniffers. The sniffers learn it sequentially making better channel assignment decisions.

References

[1] P. Arora, C. Szepesvári and R. Zheng, "Sequential learning for optimal monitoring of multi-channel wireless networks," 2011 Proceedings IEEE INFOCOM, Shanghai, 2011, pp. 1152-1160.

[2] Aurélien Garivier (LTCI), Eric Moulines (LTCI), "On Upper-Confidence Bound Policies for Non-Stationary Bandit Problems,"

[3] T. Le, C. Szepesvári and R. Zheng, "Sequential Learning for Multi-Channel Wireless Network Monitoring With Channel Switching Costs," in IEEE Transactions on Signal Processing, vol. 62, no. 22, pp. 5919-5929, Nov.15, 2014.

[4] N. Cesa-Bianchi, G. Lugosi, and G. Stoltz, “Regret minimization under partial monitoring”, Mathematics of Operations Research, vol. 31, no. 3, pp. 562–580, Aug. 2006.

[5] L.Kocsis and C. Szepesvári. Discounted UCB. 2nd PASCAL Challenges Workshop, Venice, Italy, April 2006

[6] Xu, Jing, Qingsi Wang, Kai Zeng, Mingyan Liu and Wei Liu. “Sniffer Channel Assignment With Imperfect Monitoring for Cognitive Radio Networks.” IEEE Transactions on Wireless Communications 15 (2016): 1703-1715.

[7] Zhang, T.; Liu, S.; Xiang, W.; Xu, L.; Qin, K.; Yan, X. A Real-Time Channel Prediction Model Based on Neural Networks for Dedicated Short-Range Communications. Sensors 2019, 19, 3541.