

Learning Mechanisms for Achieving Context Awareness and Intelligence in Cognitive Radio Networks

*A Technical Report submitted to the School of Engineering and Computer Science,
Victoria University of Wellington, New Zealand, February 2010
Technical Report: ECSTR10-03*

Kok-Lim Alvin Yau, Peter Komisarczuk, Paul D. Teal, and Winston K. G. Seah
Communications and Networking Research Group
School of Engineering and Computer Science, Victoria University of Wellington
P.O. Box 600, Wellington 6140, New Zealand
{kok-lim.yau, peter.komisarczuk, paul.teal}@ecs.vuw.ac.nz

Abstract—Given that the licensed or Primary Users (PUs) are oblivious to the presence of unlicensed or Secondary Users (SUs), Cognitive Radio (CR) is a new paradigm in wireless communication that allows the SUs to detect and use the underutilized licensed spectrum opportunistically and temporarily. Context awareness and intelligence are key characteristics of CR to enable the SU to sense for and use the underutilized licensed spectrum in an efficient manner. In this technical report, we investigate into various learning mechanisms for achieving context awareness and intelligence in CR networks. The learning mechanisms are Adaptation (Adapt), Window (Win), Adaptation-Window (AdaptWin), and Reinforcement Learning (RL). We investigate the learning mechanisms with respect to dynamic channel selection scheme that helps SU base station to select channel adaptively for data transmission to SU host in static and mobile centralized CR networks. The purpose is to enhance quality of service, particularly throughput and delay, and in terms of minimising number of channel switchings. Channel heterogeneity is considered in this paper. Our contribution in this paper is to investigate into learning mechanisms that are simple yet pragmatic for CR networks. Simulation results reveal that RL, AdaptWin and Win achieve approximately similar and the best possible network performance, followed by Adapt, and finally Random scheme, which does not apply any learning mechanism and serves as baseline.

I. INTRODUCTION

Cognitive Radio (CR) enables unlicensed spectrum users or Secondary Users (SU)s to change its operating frequency and use the underutilized spectrum of licensed users' or Primary Users (PU)s' (or white space) conditional on the interference to the PUs, which are unaware of the SUs, being below an acceptable level. Context awareness enables each SU to be aware of its operating environment; while intelligence enables each SU to make the right decision at the right time to achieve optimum performance. Generally speaking, context awareness and intelligence enable each SU to *sense*, *learn*, and *respond* in an efficient manner with respect to its operating environment without adhering to a *strict and static* self-defined policy.

In [1], a learning mechanism called Reinforcement Learning (RL) [2], or specifically, multi-armed bandit is investigated with respect to Dynamic Channel Selection (DCS). In [3], RL is applied to enable each SU to detect PU signal that may have deviated from its known signature. The investigation in [1] and [3] use machine learning performance metrics such as regret and fitness value, while this paper uses network performance metrics such as throughput and number of channel switchings. In [4], RL is applied in DCS in distributed CR networks in order to reduce call blocking and dropping probability, and the effects of RL parameters on network performance are investigated. In [5], RL is applied in DCS among the Base Stations (BS)s in order to enable each BS to cover a minimum percentage of service area with the highest SINR so as to reduce call blocking and dropping probability. In [6], RL is applied to identify channels with the most available white spaces at the BS in centralized networks. In [7], RL is applied in spectrum assignment strategy in OFDMA networks in order to improve the PU's performance metrics including spectral efficiency, users' Quality of Service (QoS) satisfaction, and the amount of licensed spectrum to be released to SUs. As a complement to [1], [3]-[7], this paper compares the RL results with other learning mechanisms.

This paper investigates the use of various simple and yet pragmatic learning mechanisms including Adaptation (Adapt), Window (Win), Adaptation-Window (AdaptWin), and RL to achieve context awareness and intelligence in CR networks with respect to DCS scheme. The DCS enables SU BS to select heterogeneous channel adaptively for data transmission to SU host in static and mobile centralized CR networks. Each channel has different levels of Primary User Utilization (PUL). Higher levels of PUL indicates higher levels of PU activity, and hence smaller amount of white spaces. Additionally, by considering heterogeneous channels, the channels may have different characteristics including Packet Error Rate (PER) and transmission range.

Higher levels of PER indicates higher levels of failed data packet transmission due to interference, path loss, time-varying channel conditions, neighbour interference, and other factors. Due to channel heterogeneity, a channel with low PUL does not imply a good channel if it has a high PER. The SU transmits using a fixed transmission power at different channels; hence the transmission range for each channel varies as shown in Figure 1. In general, lower frequency provides larger transmission range. Hence, an SU switches its operating channel when PUL or PER becomes high, as well as the transmission range becomes insufficient. A thorough investigation into the effects of multiple SU hosts is available in [8]. As similar trends are observed with single and multiple SU hosts, we assume that there are only two SUs, namely an SU BS and a static or mobile SU host to model a centralized network.

Our purpose is to enhance QoS, particularly throughput and delay, in terms of number of channel switchings. In [9], we have successfully applied the RL approach in DCS for centralized CR networks. The contribution of this paper is twofold. Firstly, we investigate into the effects of various parameters in Adapt, Win, AdaptWin, and RL on network performance. Secondly, using the optimal results from the preceding investigation, we compare the network performance of the learning mechanisms. The rest of the paper is organized as follows. Section II provides overview of learning mechanisms. Section III presents learning-based DCS scheme. Section IV provides details of the learning mechanisms. Section V shows simulation experiments, results and discussions. Section VI presents our conclusions.

II. AN OVERVIEW OF LEARNING MECHANISMS

We model the SU BS, which selects a channel adaptively for data transmission to SU host, as a learning agent as shown in Figure 2. At a particular time instant, the agent observes only its operating environment. The agent can improve its local reward in the next time instant by carrying out a proper action. The learning engine provides knowledge on the operating environment by observing the consequences of its prior action [2]. As time progresses, the agent learns to carry out the proper action to maximize its reward. For instance, the learning engine is used to learn the channel conditions including PUL, PER, and transmission range. Based on the application schemes, the reward indicates distinctive performance metrics such as throughput and successful packet transmission rate. Thus, maximizing the reward provides network performance enhancement.

There are two types of actions. Exploitation chooses the best known or greedy action for performance enhancement; while exploration chooses random action once in a while in order to improve the knowledge so that better action may be

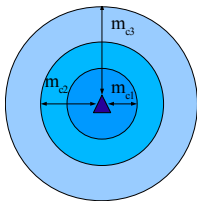


Figure 1. An SU and its transmission ranges using a fixed transmission power at different channels.
ECSTR10-03

discovered. This is necessary as choosing the best known action at all times does not cater for the actions that are never chosen. In CR networks, exploration is necessary as most application schemes require a CR host to keep track of its operating environment, i.e. Out-of-Band Measurement that requires the BS to keep a list of prioritized backup channels in IEEE 802.22 [10]. Hence, all the learning mechanisms in this paper perform exploration. This paper applies ϵ -greedy approach [2] where an agent chooses the greedy action as its next action with probability $1-\epsilon$, and random action with a small probability ϵ .

Figure 3 shows the flowchart of the learning mechanism model. For *action selection*, the agent observes the operating environment, choose an exploration or exploitation action, and execute the action. For *knowledge update*, the agent observes the operating environment, and update its knowledge.

Applying simple and yet pragmatic learning mechanisms in DCS provides several advantages. Firstly, it helps an SU to adapt to its dynamic and uncertain operating environment. Secondly, using a simple modeling approach, the complexity involved in modeling the operating environment and channel heterogeneity can be minimized. For instance, an SU that selects a channel for data transmission does not model the channel behavior, which is characterized by channel selective fading, path loss, PU interference and others that affect an SU's performance in a complex manner. Thirdly, rather than addressing a single factor at a time, a learning agent observes relevant factors in decision making and optimizes a general goal as a whole, such as throughput, with regard to the environment.

III. LEARNING-BASED DYNAMIC CHANNEL SELECTION SCHEME

The DCS scheme provides the strategy to select an available licensed channel for data transmission from an SU BS to an SU host given that the objective is to maximize overall throughput and minimize delay, in terms of number of channel switchings, in the presence of different levels of PUL and PER in the licensed channels with different transmission ranges, as well as nodal mobility.

The DCS scheme is modeled using various kinds of learning mechanisms. The learning mechanism is embedded in the SU BS, which is the decision maker that determines the channel to use for data transmission to the SU host. The probability of a successful data packet transmission is dependent on the channel PUL, PER and transmission range. Due to the limited sensing capability at each SU, there are K available channels. The action is to choose a

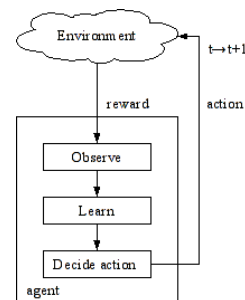


Figure 2. Abstract view of a learning agent in its environment.

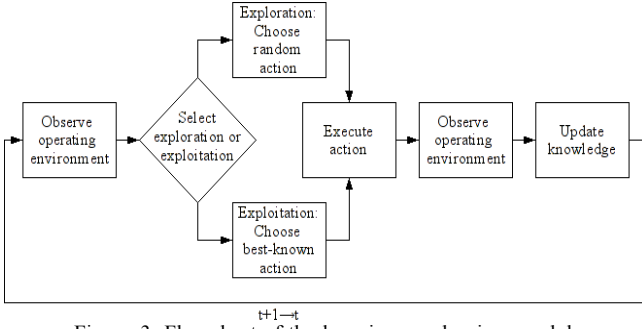


Figure 3. Flowchart of the learning mechanism model.

channel for data transmission from the available channels set $\{C=c_1, c_2, \dots, c_K\}$. Data packet transmission is successful when a link-layer acknowledgment is received for the data packet sent, else the transmission is unsuccessful. Additionally, if an SU senses PU signals immediately prior to data packet transmission, it is considered unsuccessful.

IV. DETAILS OF LEARNING MECHANISMS

In this paper, the learning mechanisms determine how an SU BS chooses its operating channel for data transmission in DCS. There are two major differences among the four kinds of learning mechanisms, namely Adapt, Win, AdaptWin, and RL. Firstly, during action selection, “How does the agent choose its best known action?” Secondly, during knowledge update, “How does the agent maintain and update its knowledge?” In the next few subsections, we present the learning mechanisms based on the two aforementioned features.

A. Adaptation (Adapt) Approach

During channel selection, the agent chooses its previous chosen channel. Two conditions that trigger channel switching are number of consecutive failed data packet transmissions, n^{Adapt} and exploration. After channel switching, the agent remains in the channel until either one of the two conditions are encountered. There is no knowledge update. The channel selection is random.

B. Window (Win) Approach

In Win approach, the SU BS keeps track of the probability of successful data packet transmission, P_{S,c_i}^{Win} for all available channels C in a Win-table with $|C|$ entries. Denote the number of most recent attempts of data packet transmissions or window size by n^{Win} , and the number of successful data packet transmissions within n^{Win} using channel c_i by n_{S,c_i}^{Win} . During channel selection, the agent computes the probability of successful data packet transmission using channel c_i , $P_{S,c_i}^{Win} = n_{S,c_i}^{Win} / n^{Win}$, and chooses the channel with the highest P_{S,c_i}^{Win} . Two conditions that trigger channel switching are a channel with higher P_{S,c_i}^{Win} is discovered, and exploration. During knowledge update, the agent computes P_{S,c_i}^{Win} and updates its Win-table.

C. Adaptation-Window (AdaptWin) Approach

AdaptWin incorporates both Adapt and Win approaches. During action selection, the agent chooses the channel with

the highest P_{S,c_i}^{Win} . Two conditions that trigger channel switching are number of consecutive failed data packet transmissions, n^{Adapt} and exploration. During knowledge update, the agent computes P_{S,c_i}^{Win} and updates its Win-table. Hence, during action selection and knowledge update, AdaptWin follows the Win approach; while the conditions that trigger channel switching follow the Adapt approach.

D. Reinforcement Learning (RL) Approach

In RL [2] approach, the SU BS keeps track of learned action value or Q-value, $Q_t(c_i)$ for all available channels C in a Q-table with $|C|$ entries. The Q-value $Q(c_i)$ indicates the appropriateness of choosing channel c_i in the operating environment. At time $t+1$, the Q-value of a chosen channel at time t is updated as follows:

$$Q_{t+1}(c_{i,t}) \leftarrow (1-\alpha)Q_t(c_{i,t}) + \alpha r_{t+1}(c_{i,t}) \quad (1)$$

where $0 \leq \alpha \leq 1$ is the learning rate and $r_{t+1}(c_{i,t})$ is the delayed reward, which is the reward received at time $t+1$ for the action taken at time t . For every successful data packet transmission, there is a reward with positive constant value $r_{t+1}(c_{i,t}) = +RW$, otherwise a cost with negative constant value $r_{t+1}(c_{i,t}) = -CT$ is incurred. In practice, the value of RW and CT are based on the amount of revenue and cost that a network operator earns or incurs for each successful or unsuccessful data packet transmission. During channel selection, the agent chooses the channel with the highest Q-value. Two conditions that trigger channel switching are a channel with higher Q-value is discovered, and exploration. During knowledge update, the agent computes and updates the Q-value for its chosen channel using (1) in Q-table.

V. SIMULATION EXPERIMENTS, RESULTS AND DISCUSSIONS

In this section, the simulation model, assumptions and its parameters are presented. Simulations were preformed using the INET framework for OMNeT++ [11]. We have implemented a CR-enabled environment in OMNeT++. Simulation parameters are shown in Table I.

A. Simulation Model, Assumptions and Parameters

We consider a centralized CR network with an SU BS, and an SU host in all scenarios. The SU BS is static, while the SU host could be static or mobile. The SU BS is always backlogged. Each channel has its own characteristics including maximum transmission range m_{c_i} , PUL $L_{c_i}^{PU}$, and PER, $P_{c_i}^E$ where $1 \leq i \leq K$. We assume that for each channel, the channel utilization pattern of PU follows independent and identically distributed (i.i.d.) stochastic model, and follows Poisson process. Each PU accesses one of the channels, while the SU can access any one of the K available channels for data transmission. There are K PUs, each PU uses one distinctive channel and broadcasts packets throughout the entire simulation area.

In mobile networks, the SU host moves in a random manner with its speed and direction following a normal

distribution with mean and standard deviation as shown in Table I. In static network, the SU BS could communicate with the SU host using all $K=3$ channels; while in mobile network, some of the $K=3$ channels may be out of range, however, the SU must be located within the maximum transmission range from the SU BS or m_{c_s} in Figure 1. From Table I, the RL parameters RW and CT were chosen empirically to optimize simulation performance.

TABLE I
NOTATIONS AND DEFAULT PARAMETER SETTINGS IN SIMULATION

Category	Symbol	Details	Values
Initialization	N	Number of SU	2
	K	Number of available channels	3
	F	Center frequencies of available channels	{400MHz, 800MHz, 5.7GHz}
	$P_{c_i}^E$	PER of each available channel	[0.1, 0.9] Default = 0.1
	T	Total simulation time	500s
	ϵ	Exploration probability	0.1
MAC Protocol	t_{ex}	Data packet expiration timer	5.798ms
	D	Data rate	2Mbps
Mobile Networks	μ_S	Mean of speed	20m/s
	σ_S	Standard deviation of speed	8m/s
Secondary user	$t_{DATA, SU}$	Data packet duration	5.44ms
	$T_{SU, w}$	Switching delay	100 μ s
Primary user	$t_{DATA, PU}$	Data packet duration	5.44ms
	$L_{c_i}^{PU}$	PUL of each PU at each available channel	[0.1, 0.9] Default = 0.1
Adapt	n^{Adapt}	Number of consecutive failed data packet transmissions	{1, 2, 4, 8, 16, 32}
Win	n^{Win}	Window size	{1, 2, 4, 8, 16, 32}
AdaptWin	n^{Adapt}		4
	n^{Win}		{1, 2, 4, 8, 16, 32}
RL		Initial Q-value	1
	α	Learning rate of Q-learning	{0.0125, 0.05}
	RW	Reward	15
	CT	Cost	5

An illustration of the cognitive MAC is shown in Figure 4. In the figure, switching delay may be ignored if channel switching is not necessary. For the cognitive MAC to operate, each SU has two transceivers, one is tuned to a common control channel, which is free of PU, at all times for control message exchange; while the other one is tuned to any other available channels for data transmissions. The common control channel has the largest transmission range compared to the available channels for data transmission. Since the last spectrum sensing indicates the PU occupancy in a particular channel, the channel, which was free, may become busy within a Short Inter-Frame Spacing (SIFS) interval right immediately prior to data transmission. In this

case, the SU BS restarts its data transmission cycle again with RTS-CTS handshaking, and may reassign its channel. The RTS has channel switching information determined by the SU BS according to its learning mechanism. Synchronization between PUs and SUs, as well as among the SUs, are not necessary. In Table I, t_{ex} is the data packet expiration timer, which is initiated after sending a data packet and is reset upon receiving its ACK packet. The expiration of the t_{ex} indicates failed data packet transmission.

B. Performance Metrics

Our goal is to maximize throughput and minimize number of channel switchings, which causes non-negligible delay for data transmission, over different heterogeneous channels with different transmission range, PUL and PER. The mean amount of throughput and number of channel switchings of the learning-based DCS, as well as Random DCS are investigated. The Random chooses an available channel for the next data packet transmission in a random manner. Graphs are presented with PUL and PER as ordinate respectively. When PUL is ordinate, each simulation result of mean throughput or mean number of channel switchings is for all possible combinations of PUL with the other parameter values remain constant as shown in Table I. As an example, a PUL of 0.8 for $K=3$ available channels may indicate [0.8,0.8,0.8], [0.8,0.7,0.9], and [0.9,0.9,0.6]. In the case of mobile networks, each set of PUL such as [0.8,0.7,0.9] is applied to various channels with different frequencies for each simulation run. The similar randomness applies to PER among the available channels.

When network performance is investigated with respect to mean PUL for all channels, the PER for all channels are set to 0.1. This investigation shows the effectiveness of the learning approaches in choosing a channel with a low level of PU activity for data transmission in the presence of low PER for all channels. When network performance is investigated with respect to mean PER for all channels, the PUL for all channels are set to 0.1. This investigation shows the effectiveness of the learning approaches in choosing a channel with a low level of PER for data transmission in the presence of low level of PU activity for all channels.

Section C, D, E and F investigate the effects of parameter n^{Adapt} in Adapt, n^{Win} in Win and AdaptWin, and α in RL, respectively. These sections give the parameters that provide the best possible network performance, and subsequently, Section G compares the network performance of the learning mechanisms.

C. Effects of Parameters in Adapt on Network Performance

The effects of n^{Adapt} on throughput with respect to PUL and PER in mobile network is shown in Figure 5 and 6

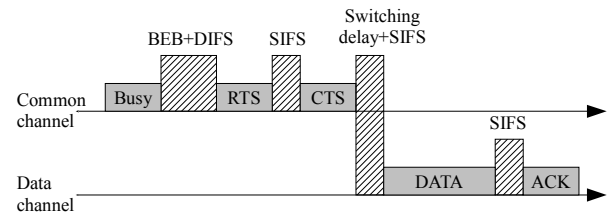


Figure 4. Illustration of cognitive MAC protocol.

respectively. Figure 5 shows that $n^{Adapt}=2$ provides the highest level of throughput for all levels of PUL. Figure 6 shows that $n^{Adapt}=1$ provides the highest level of throughput from 0.1 to 0.3 PER, followed by $n^{Adapt}=2$ from 0.4 to 0.6 PER, and followed by $n^{Adapt}=4$ from 0.7 to 0.9 PER. Hence, $n^{Adapt}=2$ provides higher throughput with respect to PUL and PER in mobile networks because it is more adaptive to the changes in the amount of white spaces in each channel. The effects of n^{Adapt} on throughput is not significant for various levels of PUL and PER in static network; so their graphs are not shown in this paper.

D. Effects of Parameters in Win on Network Performance

The effects of n^{Win} on throughput with respect to PUL in static and mobile network is shown in Figure 7 and 8 respectively. Figure 7 shows that $n^{Win}=8$ provides the highest level of throughput at 0.1 PUL, and followed by $n^{Win}=32$ from 0.2 to 0.9 PUL. Figure 8 shows that $n^{Win}=32$ provides the highest level of throughput for all levels of PUL.

The effects of n^{Win} on throughput with respect to PER in static and mobile network is shown in Figure 9 and 10 respectively. Figure 9 shows that $n^{Win}=8$ provides the

highest level of throughput at 0.1 PER, and followed by $n^{Win}=32$ from 0.2 to 0.9 PER. Figure 10 shows that $n^{Win}=32$ provides the highest level of throughput for all levels of PER.

Hence, window size $n^{Win}=32$ provides higher throughput in all cases because higher number of most recent attempts of data packet transmissions (or historical information) is applied to compute the probability of successful data packet transmission, $P_{S,c}^{Win}$.

E. Effects of Parameters in AdaptWin on Network Performance

We set $n^{Adapt}=2$ as it provides the best possible performance in the Adapt approach. The effects of n^{Win} on throughput with respect to PUL and PER in static network is shown in Figure 11 and 12 respectively. Figure 11 shows that $n^{Win}=8$ provides the highest level of throughput for 0.1 PUL, followed by $n^{Win}=32$ from 0.2 to 0.8 PUL, and followed by $n^{Win}=16$ for 0.9 PUL. Figure 12 shows that $n^{Win}=8$ provides the highest level of throughput for 0.1 PER, and followed by $n^{Win}=32$ from 0.2 to 0.9 PER, and followed by $n^{Win}=16$ for 0.9 PER. The effects of n^{Win} on throughput is not significant for various levels of PUL and PER in mobile network; so their graphs are not shown in

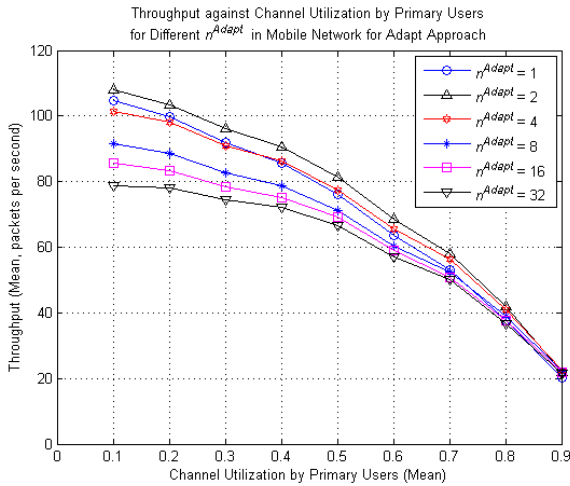


Figure 5. The mean throughput of an SU sender versus mean PUL for Adapt approach with different n^{Adapt} values in mobile networks. PER for all channels are set to 0.1.

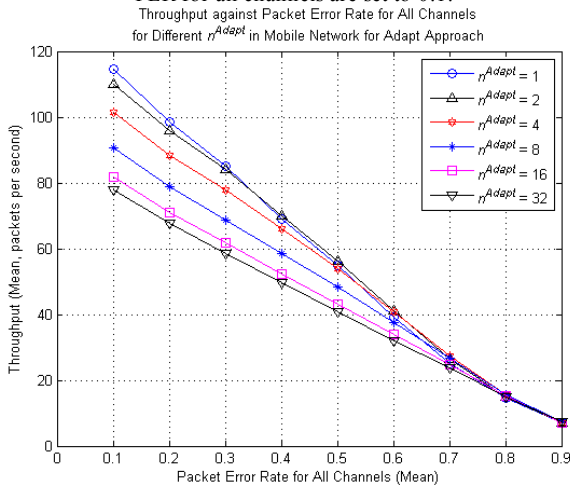


Figure 6. The mean throughput of an SU sender versus mean PER for Adapt approach with different n^{Adapt} values in mobile networks. PUL for all channels are set to 0.1.

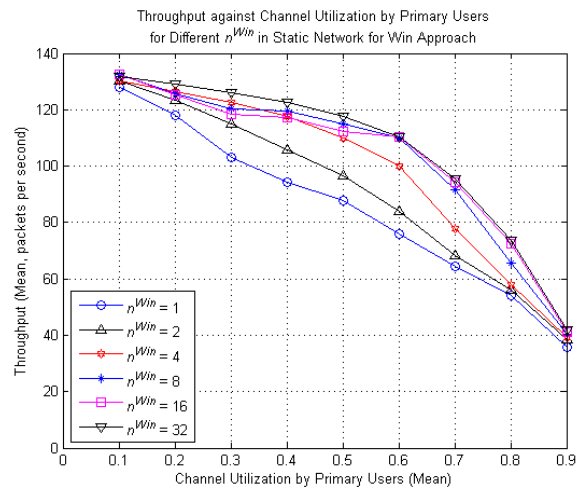


Figure 7. The mean throughput of an SU sender versus mean PUL for Win approach with different n^{Win} values in static networks. PER for all channels are set to 0.1.

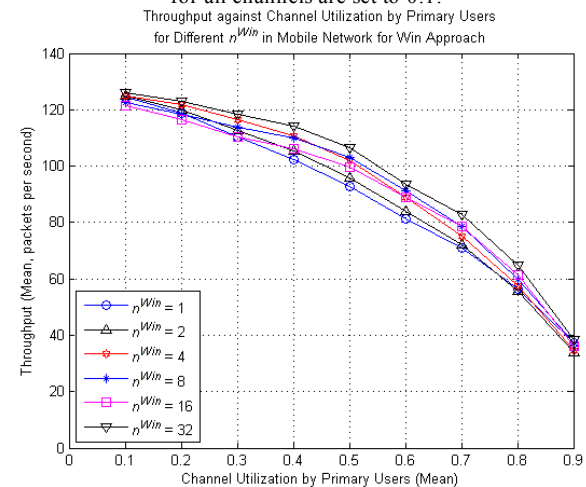


Figure 8. The mean throughput of an SU sender versus mean PUL for Win approach with different n^{Win} values in mobile networks. PER for all channels are set to 0.1.

this paper. Hence, window size $n^{Win}=32$ provides higher throughput in all cases.

F. Effects of Parameters in RL on Network Performance

The effects of α on throughput and number of channel switchings are investigated in [12]. It has been shown that RL with $\alpha=0.0125$ and $\alpha=0.05$ provide the best network performance in static and mobile network respectively.

G. Comparison of Learning Mechanisms

This section compares the network performances, namely throughput and number of channel switchings, among the learning mechanisms. From the preceding sections, the parameters that provide the best possible throughput performance are $n^{Adapt}=2$ for Adapt, $n^{Win}=32$ for Win, $n^{Adapt}=2$ and $n^{Win}=32$ for AdaptWin, $\alpha=0.0125$ for RL in static network and $\alpha=0.05$ for RL in mobile network.

Figure 13 and 14 show the throughput and number of channel switchings achieved by the learning mechanisms with respect to PUL in a static network. Figure 13 shows that, in comparison with Random, at 0.5 PUL, the RL, AdaptWin and Win approaches provide approximately 1.7

times, and Adapt provides 1.5 times throughput enhancement. Figure 14 shows that, in comparison with Random within 500s of simulation run, at 0.5 PUL, the RL, AdaptWin and Win approaches provide 4.9 times, while Adapt provides 4.6 times number of channel switchings reduction.

Figure 15 and 16 show the throughput and number of channel switchings achieved by the learning mechanisms with respect to PUL in a mobile network. Figure 15 shows that, in comparison with Random, at 0.5 PUL, the RL, AdaptWin and Win approaches provide approximately 2.7 times, and Adapt provides 2.1 times throughput enhancement. Figure 16 shows that, in comparison with Random within 500s of simulation run, at 0.5 PUL, the RL, AdaptWin and Win approaches provide 4.7 times while Adapt provides 3.5 times number of channel switchings reduction.

Figure 17 and 18 show the throughput and number of channel switchings achieved by the learning mechanisms with respect to PER in a static network. Similar trends are observed for network performance with respect to PUL in Figure 13 and 14 respectively.

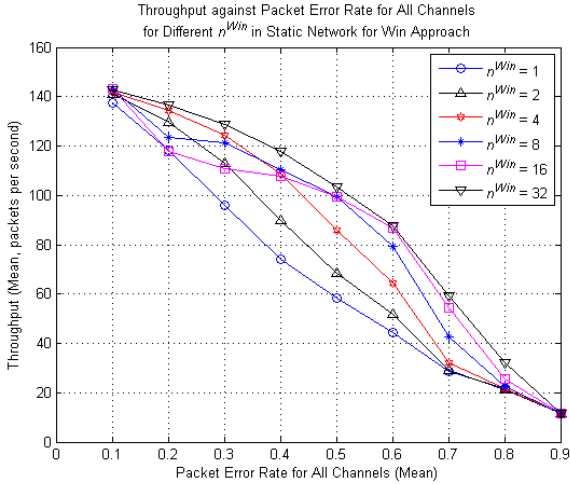


Figure 9. The mean throughput of an SU sender versus mean PER for Win approach with different n^{Win} values in static networks. PUL for all channels are set to 0.1.

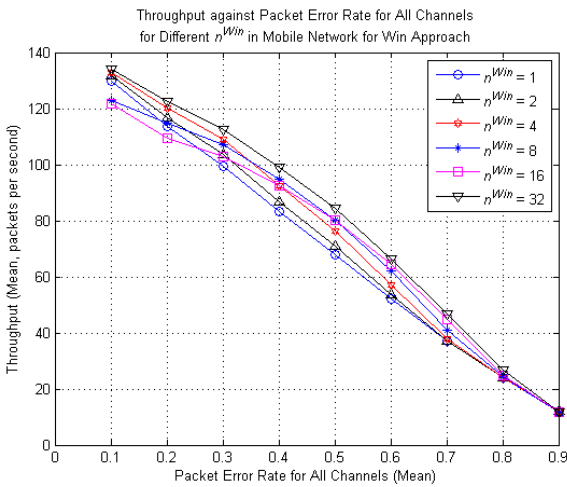


Figure 10. The mean throughput of an SU sender versus mean PER for Win approach with different n^{Win} values in mobile networks. PUL for all channels are set to 0.1.

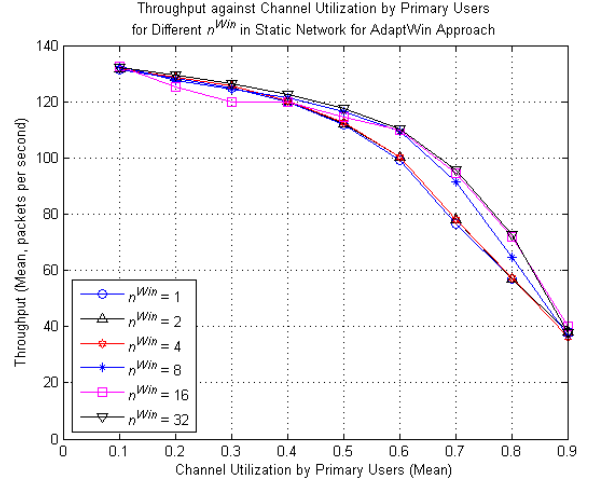


Figure 11. The mean throughput of an SU sender versus mean PUL for AdaptWin approach with different n^{Win} values in static networks. PER for all channels are set to 0.1. n^{Adapt} is set to 4.

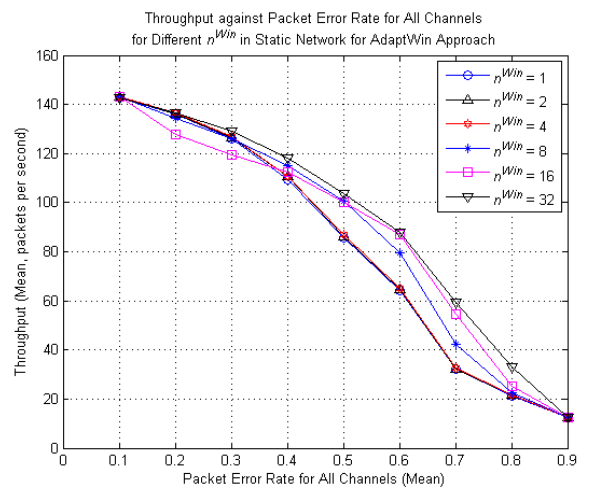


Figure 12. The mean throughput of an SU sender versus mean PER for AdaptWin approach with different n^{Win} values in static networks. PER for all channels are set to 0.1. n^{Adapt} is set to 4.

Figure 19 and 20 show the throughput and number of channel switchings achieved by the learning mechanisms with respect to PER in a mobile network. Similar trends are observed for network performance with respect to PUL in Figure 15 and 16 respectively.

The results in this section shows that, in most of the cases, the RL, AdaptWin and Win approaches achieve approximately similar network performance. In these schemes, an agent receives reward for successful data packet transmissions, and cost for unsuccessful ones. The reward is RW in RL, and the probability of $1/n^{Win}$ in AdaptWin and Win. The cost is CT in RL, and the probability of $1/n^{Win}$ in AdaptWin and Win.

The RL, AdaptWin and Win approaches achieve the best network performance, followed by Adapt, and finally Random. Hence, the learning mechanisms help the SU node to choose a channel with low PUL, PER and sufficient transmission range for data transmission. The learning mechanisms increase throughput and reduce number of channel switchings, which decreases switching delay. The RL chooses the channel with the highest Q-value $Q(c_i)$, while AdaptWin and Win choose the channel with the

highest probability of successful data packet transmission $P_{S,c}^{Win}$. In contrast, Adapt chooses channel in a random manner during channel switching, hence its network performance is lower than RL, AdaptWin and Win.

This paper also shows that simple and yet pragmatic learning mechanisms such as AdaptWin and Win achieve similar network performance provided by the RL in [1], [3]-[7].

CONCLUSIONS

In this paper, we applies four types of simple and yet pragmatic learning mechanisms to achieve context awareness and intelligence in static and mobile centralized Cognitive Radio (CR) networks, which is comprised of a Secondary User (SU) base station and an SU host, with respect to Dynamic Channel Selection (DCS). The learning mechanisms are Adaptation (Adapt), Window (Win), Adaptation-Window (AdaptWin), and Reinforcement Learning (RL). The learning mechanisms differ among themselves in action selection and knowledge update. In addition, the Random approach, which chooses an available channel for data transmission in a uniformly distributed

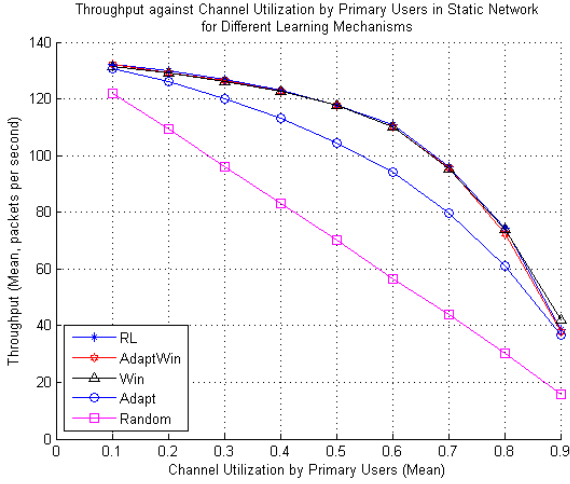


Figure 13. The mean throughput of an SU sender versus mean PUL for learning mechanisms in static networks. PER for all channels are set to 0.1.

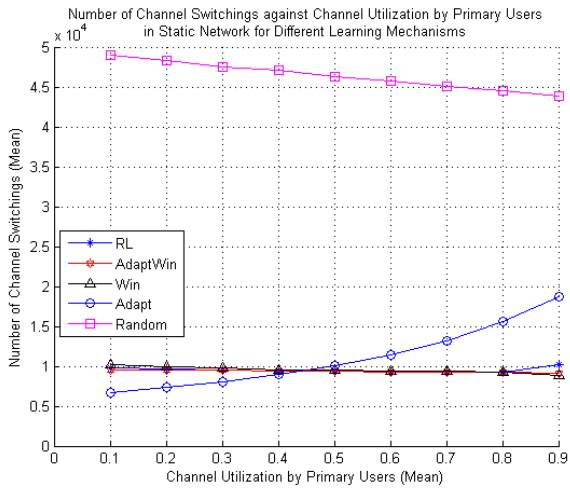


Figure 14. The mean number of channel switchings of an SU sender versus mean PUL for learning mechanisms in static networks. PER for all channels are set to 0.1.

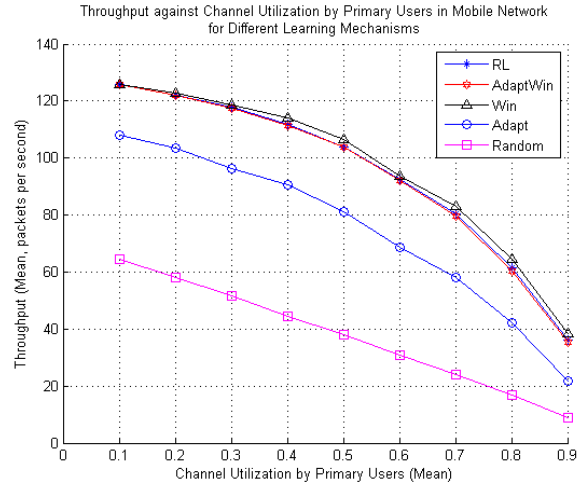


Figure 15. The mean throughput of an SU sender versus mean PUL for learning mechanisms in mobile networks. PER for all channels are set to 0.1.

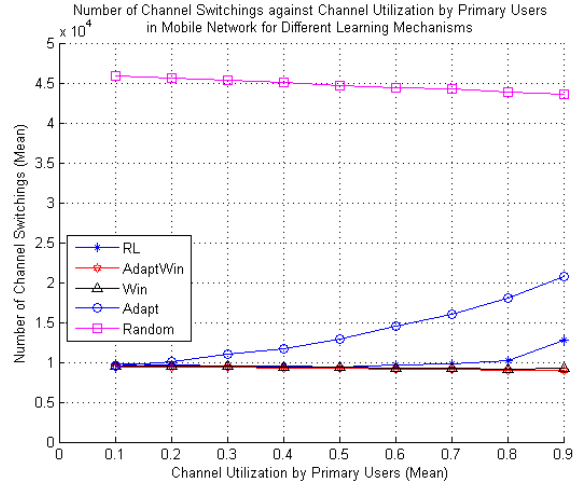


Figure 16. The mean number of channel switchings of an SU sender versus mean PUL for learning mechanisms in mobile networks. PER for all channels are set to 0.1.

random manner without learning, serves as a baseline. The learning mechanism parameters that provide the best possible network performance were investigated, and are applied to compare the learning mechanisms. The RL, AdaptWin, and Win provide approximately similar and highest network performance, followed by Adapt, and finally Random.

REFERENCES

[1] A. Alaya-Feki, E. Moulines, and A. LeCorneic, "Dynamic spectrum access with non-stationary multi-armed bandit", *Signal Processing Advances in Wls. Comm. (SPAWC) IEEE 9th Workshop on*, 2008.

[2] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*. Cambridge MA, MIT Press, 1998.

[3] Y. B. Reddy, "Detecting primary signals for efficient utilization of spectrum using Q-learning", *IT: N. Gen., 5th Intl Conf. (ITNG)*, 2008.

[4] T. Jiang, D. Grace, and Y. Liu, "Performance of cognitive radio reinforcement spectrum sharing using different weighting factors", *Comm. and Nwk. in China (ChinaCom) 3rd Intl. Conf. on*, 2008.

[5] M. Yang, and D. Grace, "Cognitive radio with reinforcement learning applied to heterogeneous multicast terrestrial communication systems," *Cog. Rad. Otd. Wls. Nwk. & Comm. (CROWNCOM) 4th Intl. Conf. on*, 2009.

[6] U. Berthold, F. Fu, M. v. d. Schaar, and F. K. Jondral, "Detection of spectral resources in cognitive radios using reinforcement learning", *N. Frontiers in Dy. Spec. Acc. Nwk. (DySPAN) 3rd Sym. on*, 2008.

[7] F. Bernado, J. Perez-Romero, O. Sallent, and R. Agusti, "A self-organized spectrum assignment strategy in next generation OFDMA networks providing secondary spectrum access", *Intl. Conf. Comm. (ICC)*, 2009.

[8] K.-L. A. Yau *et al.* "Achieving Context Awareness and Intelligence in Cognitive Radio Networks using Reinforcement Learning for Stateful Applications" *Technical Report ECSTR10-01, School of Engineering and Computer Science, Victoria University of Wellington, New Zealand*, January 2010.

[9] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal, "A context-aware and intelligent dynamic channel selection scheme for cognitive radio networks," *Cognitive Radio Oriented Wireless Networks and Communications (CrownCom'09)*, June 2009.

[10] <http://www.ieee802.org/22/>

[11] INET Framework for OMNet++/OMNEST release 2006-10-12. <http://www.omnetpp.org/doc/INET/>.

[12] K.-L. A. Yau, P. Komisarczuk, and P. D. Teal, "Performance analysis of reinforcement learning for achieving context-awareness and intelligence in cognitive radio networks", *9th IEEE Intl. Wsp. on Wls. Local Nwks. (WLN)*, October 2009.

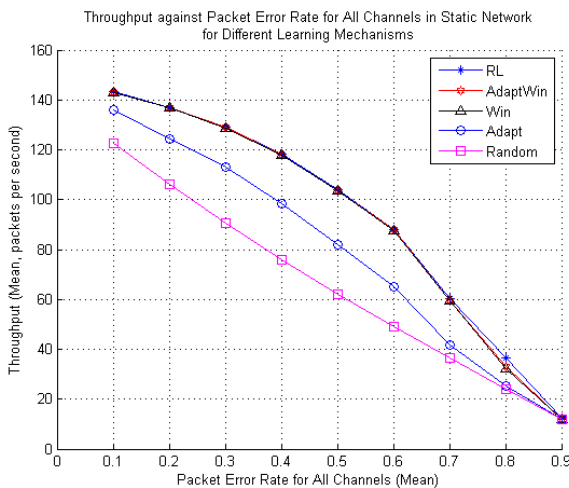


Figure 17. The mean throughput of an SU sender versus mean PER for learning mechanisms in static networks. PUL for all channels are set to 0.1.

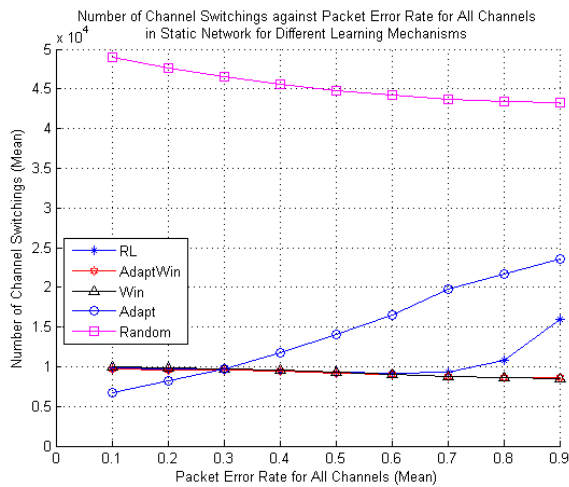


Figure 18. The mean number of channel switchings of an SU sender versus mean PER for learning mechanisms in static networks. PUL for all channels are set to 0.1.

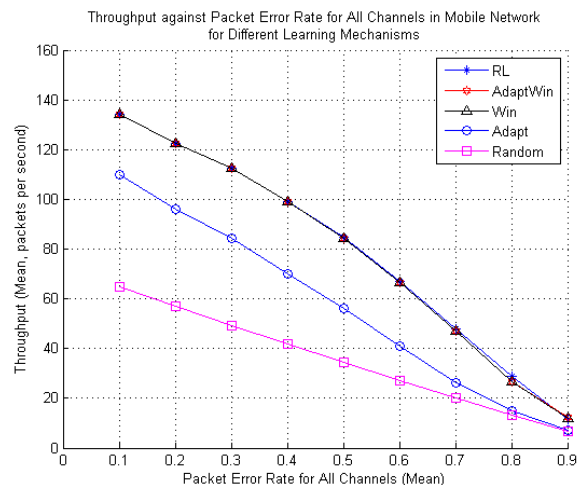


Figure 19. The mean throughput of an SU sender versus mean PER for learning mechanisms in mobile networks. PUL for all channels are set to 0.1.

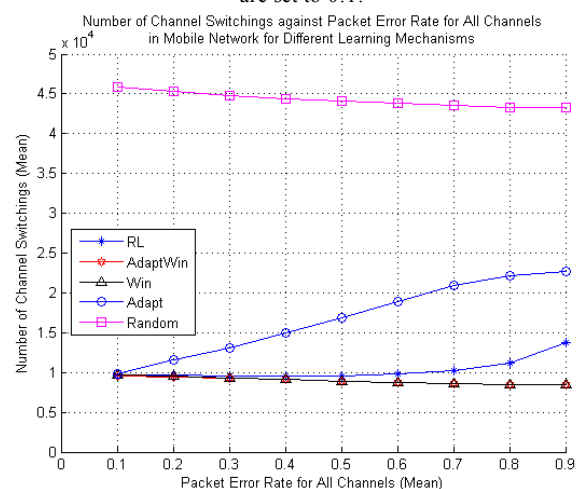


Figure 20. The mean number of channel switchings of an SU sender versus mean PER for learning mechanisms in mobile networks. PUL for all channels are set to 0.1.